

**НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ПРОБЛЕМ МОДЕЛЮВАННЯ В ЕНЕРГЕТИЦІ ім. Г.Є. ПУХОВА**

Кваліфікаційна наукова
праця на правах рукопису

КОВАЛЕНКО ОЛЕКСІЙ ЄПІФАНОВИЧ

УДК 004.891.2:004.82

ДИСЕРТАЦІЯ

**МОДЕЛІ І МЕТОДИ ПОБУДОВИ КОНВЕРГЕНТНИХ СИСТЕМ
СИТУАЦІЙНОГО УПРАВЛІННЯ**

Спеціальність 05.13.05 – комп'ютерні системи та компоненти

Галузь знань – інформаційні технології

Подається на здобуття наукового ступеня доктора технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело



О.Є. Коваленко

(підпис, ініціали та прізвище здобувача)

Науковий консультант – Верлань Анатолій Федорович

чл.-кор. НАПН України, доктор технічних наук, професор

Київ – 2021

АНОТАЦІЯ

Коваленко О.Є. Моделі і методи побудови конвергентних систем ситуаційного управління. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти. – Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, Київ, 2021.

Дисертація присвячена вирішенню проблеми розвитку теоретичних засад побудови конвергентних комп'ютерних систем ситуаційного управління шляхом створення нових моделей та розвитку методів забезпечення ефективності таких систем із урахуванням вимог і обмежень цільової області та використанням моделей знань. Розробка формальних моделей архітектури ситуаційних систем і обґрунтування їх ефективності є основою для створення конвергентних систем ситуаційного управління. Конвергенція заснована на використанні корисних спроможностей та принципів, властивих різним технологіям. Процес ситуаційного управління можна розглядати як ланцюжок збільшення цінності інформації, і рішення, по суті, є інформаційним продуктом. Використання відповідних інформаційних та комп'ютерних технологій надає додаткову цінність інформації на різних етапах та стадіях ситуаційного управління. Загалом процес ситуаційного управління можна охарактеризувати як процес трансформації та збільшення цінності інформації на основі використання конвергентних комп'ютерних систем.

Метою дослідження є створення нових моделей і розвиток існуючих методів проектування проблемно-орієнтованих комп'ютерних інформаційних систем із високими функціональними та структурними показниками для вирішення задач ситуаційного управління.

У дисертаційному дослідженні розвинуто концептуальні основи побудови моделей компонентних архітектур конвергентних систем ситуаційного управління

для забезпечення високих показників їх ефективності з урахуванням обмежень та вимог сфери застосування. При виконанні роботи отримані такі наукові результати.

Запропоновано визначення ситуації як усвідомленого знання суб'єкта про динаміку навколишнього середовища, представлене певними типами інформаційних повідомлень і є основою для побудови семантичної моделі взаємозв'язків між сутностями навколишнього середовища. Для формалізації знань у сфері ситуаційного управління запропоновані класифікація ситуацій на основі їх типів і модальностей, онтологічна модель представлення такої класифікації.

На основі аналізу класифікацій ситуаційних систем пропонується узагальнена класифікація ситуаційних систем, яка враховує конкретне призначення ситуаційної системи та служить основою для побудови спеціалізованих моделей знань предметних областей ситуаційної діяльності.

На основі аналізу інформаційних категорій та спрямованості їх практичного використання пропонується модель трансформації інформації в циклі ситуаційного управління. Модель базується на методі подання та контексті використання інформації, який поєднує різні категорії інформації та забезпечує її адекватне використання в різних моделях ситуаційного управління.

На основі аналізу моделей когнітивного циклу ситуаційного управління розроблено повну формальну модель циклу ситуаційного управління шляхом використання відповідної системи логічних міркувань та прийняття рішень на кожному етапі трансформації інформації ситуаційного управління. Абдуктивні міркування використовуються для побудови формальних моделей ситуаційних гіпотез. Індуктивне міркування – для побудови формально обґрунтованої теоретичної моделі ситуації на основі гіпотез. Прийняття рішень під час ситуаційного управління здійснюється шляхом дедуктивних міркувань на основі формальної моделі ситуації або шляхом застосування механізмів логічного умовиводу на прецедентах.

Запропоновано агентно-орієнтовану архітектуру систем ситуаційного управління, засновану на особливостях когнітивної моделі перцептивного циклу та на моделі перетворення інформації в ситуаційних системах. При описі поведінки

«розумних» агентів враховується їх обмежена раціональність, пов'язана з проблемою вибору, обмеженнями знань, наявними ресурсами тощо. Обмежена раціональність є основою математичного моделювання при прийнятті рішень, коли прийняття рішення розглядається як процес пошуку оптимального раціонального вибору з множини альтернатив на основі наявної інформації. Запропоновано модель багаторівневої архітектури для побудови мультиагентних ситуаційних систем, яка забезпечує координацію семантики, функціональності та супроводу архітектурних рішень, дозволяє реалізувати наскрізний процес проєктування таких систем. Запропонована формалізація процесу проєктування є уніфікованою і може підтримуватися будь-якими відомими технологіями моделювання систем. Крім того, ця модель адекватно представляє різні рівні абстракції опису інформаційних систем, зберігаючи при цьому семантичний зв'язок між цими рівнями і коректність переходу від одного рівня абстрактного подання до іншого без втрат семантичної єдності всіх рівнів.

Запропоновано та розроблено метод побудови композиційної моделі конвергенції компонентів систем ситуаційного управління з використанням моделей знань. Застосування заснованого на знаннях підходу дозволяє встановити відповідність між формальним описом предметної області та засобами інформаційних технологій при побудові систем ситуаційного управління. Запропонований підхід забезпечує формування сховища зразків та конвергенцію компонентів інформаційних технологій для вирішення проблем ситуаційного управління у цільовій області шляхом композиції етапів онтологічного аналізу вимог, функціональної декомпозиції, інтерпретації предметної області та фізичної реалізації. Використання запропонованої моделі процесу проєктування забезпечує формалізований синтез ситуаційної системи управління для цільової предметної області шляхом конвергенції необхідних компонентів на основі моделей знань.

Розвинуто методи оптимізації для побудови ефективної архітектури системи ситуаційного управління на основі узагальненого критерію ефективності та багатокритеріальної оптимізації за критерієм Парето, для яких розроблено прототип web-орієнтованого програмного застосунку. Запропоновані методи оцінки

альтернативних варіантів конфігурації систем ситуаційного управління та метод багатокритеріальної оптимізації таких систем дозволяють реалізувати цільовий механізм вибору оптимальних варіантів конфігурації в умовах часткової невизначеності або зміни системних вимог. Програмна реалізація запропонованої методології була використана як аналітичний компонент у системах підтримки прийняття рішень при обґрунтуванні вибору конфігурацій інформаційної системи і в системах управління проектами в процесі створення та обслуговування систем ситуаційного управління. Використання web-орієнтованої програмної системи для оцінки ефективності архітектури дозволяє ефективно керувати та супроводжувати проекти, отримувати актуальну інформацію про стан та конфігурацію проекту, обґрунтовано обирати архітектурні рішення. Інструменти опису метаданих внутрішнього проекту забезпечують гнучкість опису різних систем та адаптації конкретних проектів до вимог користувача.

Запропоновано метод та алгоритм оцінки динамічної стійкості ситуаційної системи управління як дисипативної системи на основі аналізу змін рівня інформаційної ентропії.

Запропоновано метод забезпечення технічної інтероперабельності, заснований на ефективному завантаженні та мінімізації часу обслуговування запитів у розподілених інформаційних системах шляхом розв'язування прямої та двоїстої задач математичного програмування. Запропоновані моделі підтримки технічної інтероперабельності та алгоритми розподілу задач у середовищі конвергентних систем ситуаційного управління забезпечують їх ефективне управління.

Розвинуто моделі аналізу подіє-орієнтованих систем, заснованих на теорії масового обслуговування. Моделювання проводилось для комбінованих режимів отримання та обслуговування заявок, що відповідають детермінованому та марківському процесам. Застосування математичного апарату систем масового обслуговування дозволяє проводити таке моделювання для різних дисциплін обслуговування з різною кількістю сервісних пристроїв. Для моделювання була обрана модель архітектури системи масового обслуговування з одним операційним

пристроєм та множиною пристроїв – генераторів заявок. Результати імітаційного моделювання порівнювались із теоретичними моделями.

Подальшого розвитку отримали математичні моделі для опису динаміки ситуаційних систем. Запропоновано модель дослідження стійкості динамічної стохастичної системи на основі принципу стійкості Ляпунова у вигляді системи звичайних диференціальних рівнянь. Запропоновано та досліджено агентно-орієнтований підхід включення моделі стійкості Ляпунова в процес ситуаційного управління динамічною стохастичною системою як цільової проєктної діяльності.

Розроблена онтологічна модель знань для використання в системі ситуаційного реагування на інциденти безпеки, що дозволяє забезпечити конвергенцію в єдиній базі знань архітектурних, орієнтованих на ризик та адаптивних до уразливостей аспектів управління безпекою комп'ютерних інформаційних систем. Розробка моделей знань для управління системами безпеки забезпечує підтримку безпечного функціонування цих систем відповідно до їх призначення. Запропонована метамодель інтеграції моделей безпеки на основі дескрипційної логіки описує аспекти діяльності з управління безпекою. Визначено ключові компоненти діяльності з управління безпекою для системи підтримки прийняття рішень та запропоновано методи їх реалізації. Запропонована модель конвергенції сервісів безпеки включає набір концепцій, що представляють поєднання компонентів управління безпекою: інтегрованої моделі ризиків; інтегрованої моделі загроз безпеки; інтегрованої моделі уразливостей системи. Співвідношення між концептами та категоріями бази знань визначає семантику управління безпекою, а функції інтерпретації визначають правила суперпозиції складових моделей.

Результати теоретичних та експериментальних досліджень, а також розроблені методи знайшли практичне застосування та впровадження в Інституті проблем математичних машин і систем НАН України (м. Київ), ДП "S&T UKRAINE" (м. Київ), Scientific Research Center for Disaster Risk Reduction of the Department of Information Technology and Communications of the University of National and World Economy (м. Софія, Болгарія). Також результати дисертації впроваджені в

навчальний процес Національного університету біоресурсів і природокористування України (м. Київ), Кам'янець-Подільського національного університету імені Івана Огієнка (м. Кам'янець-Подільський) для вдосконалення теоретико-методичного забезпечення та у виконанні кваліфікаційних студентів робіт.

Ключові слова: архітектура комп'ютерної системи, система ситуаційного управління, перцептивний цикл, конвергентна система, оптимізація архітектури системи, інтелектуальний агент.

ABSTRACT

Kovalenko O. E. Models and Methods for Designing Convergent Systems of Situational Control. – Qualification scientific work as a manuscript.

Thesis for the degree of Doctor of Engineering Sciences in specialty 05.13.05 – Computer Systems and Components. – Pukhov Institute for Modelling in Energy Engineering of National Academy of Sciences of Ukraine, Kyiv, 2021.

The thesis is devoted to solving the problem of development of theoretical principles for the design of convergent computer systems of situational control through the creation of new models and improvement of methods for ensuring efficiency of such systems, taking into account requirements and restrictions of the target domain and using the models of knowledge. The development of formal models of situational systems architecture and substantiation of their efficiency is the basis for the creation of convergent situational control systems. Convergence is based on the use of useful capabilities and principles inherent in different technologies. The process of situational control can be considered as a value chain of information, and the decision is essentially an information product. The use of appropriate information and computer technologies provides additional value to information at different phases and stages of situational control. In general, the process of situational control can be described as a process of transformation and increase the value of information based on the use of convergent computer systems.

The aim of the research is to create new models and develop existing methods for designing problem-oriented computer information systems with high functional and structural performance to solve problems of situational management.

In the dissertation research the conceptual bases of the design of component architectures models of convergent situational management systems for maintenance of high indicators of their efficiency, taking into account restrictions and requirements of

the problem domain, are developed. The following scientific results were obtained in the work.

It is proposed to define a situation as a conscious knowledge of the subject about the dynamics of the environment which is represented by certain types of information messages and serves as a basis for building a semantic model of relationships between the entities of the environment. To formalize the knowledge in the domain of situational control, the classification of situations based on their types and modalities, and an ontological model of representing such classification is proposed.

Based on the analysis of the classifications of situational systems, a generalized classification of situational systems is proposed, which takes into account the specific purpose of the situational system and serves as a basis for building specialized models of knowledge of situational activities subject domains.

Based on the analysis of information categories and the focus of their practical use, a model of information transformation in the situational control cycle is proposed. The model is based on the method of presentation and context of information use which combines different categories of information and ensures its adequate use in different models of situational control.

Based on the analysis of the models of the situational control cognitive cycle, a complete formal model of the situational control cycle through the use of an appropriate system of logical reasoning and decision-making at each stage of transformation of situational control information was developed. Abductive reasoning is used to build formal models of situation hypotheses. Inductive reasoning is used to build a formally sound theoretical model of a situation based on hypotheses. Decision-making during situational control is carried out by deductive reasoning based on the formal model of the situation or by applying mechanisms of case-based reasoning on previous experience.

An agent-oriented architecture of situational control systems based on the features of the cognitive model of the perceptual cycle and on the model of information transformation in situational systems is proposed. When describing the behavior of «smart» agents, their limited rationality related to the problem of choice, knowledge limitations, available resources, etc. are taken into account. Limited rationality is the basis

of mathematical modeling in decision-making when it is seen as a rational process of finding an optimal choice from a variety of alternatives based on available information. A model of multilevel architecture is proposed for the construction of multi-agent situational systems which ensures the coordination of semantics, functionality and supporting architectural solutions, and allows implementing the end-to-end process of designing such systems. The proposed formalization of the design process is unified and can be supported by any known systems modeling technologies. In addition, this model adequately represents different levels of abstraction of the formation systems description, maintaining a meaningful connection between these levels which ensures the correctness of the transition from one level of abstract representation to another without losing the semantic unity of all levels.

A method for constructing a compositional model of convergence of situational control systems components using knowledge models is proposed and developed. The application of the knowledge-based approach allows establishing the correspondence between the formal description of the subject area and the means of information technology in the construction of situational management systems. The proposed approach provides the formation of a repository of samples and the convergence of information technology components to solve problems of situational management in the target area by composing stages of ontological analysis of requirements, functional decomposition, interpretation of the subject area and physical implementation. The use of the proposed model of the design process provides a formalized synthesis of the situational control system for the target subject area through the convergence of the necessary components based on knowledge models.

Optimization methods for building an effective architecture of a situational system based on a generalized efficiency criterion and a multi-criteria optimization according to the Pareto criterion, for which a prototype of a web-oriented software application has been implemented, have been developed. The proposed methods for evaluating alternative configuration options for situational management systems and the method of multi-criteria optimization of such systems allow implementing the target mechanism for selecting optimal configuration options in the conditions of partial uncertainty or changing system

requirements. The software implementation of the proposed methodology was used as an analytical component in decision support systems in justifying the choice of information system configurations and in project management systems in the process of creating and maintaining situational management systems. The use of a web-based software system to assess the effectiveness of the architecture allows you to effectively manage and maintain projects, obtain up-to-date information about the status and configuration of the project, reasonably choose architectural solutions. Internal project metadata description tools provide flexibility in describing different systems and adapting specific projects to user requirements.

A method and an algorithm for estimating the dynamic stability of the situational control system as a dissipative system based on the analysis of changes in the level of information entropy are proposed.

A method for ensuring technical interoperability based on efficient loading and minimization of query service time in distributed information systems by solving primal and dual problems of mathematical programming is proposed. The offered models of support of technical interoperability and algorithms of tasks distribution in the environment of convergent systems of situational management provide their effective management.

The models of analysis of event-oriented systems based on the theory of queuing have been developed. The simulation was performed for the combined modes of receiving and servicing jobs that correspond to deterministic and Markov processes. The use of the mathematical apparatus of queuing systems allows carrying out such modeling for different disciplines of service with different number of service devices. The model of queuing system architecture with one operating device and many devices – job generators – was chosen for the simulation. The results of the simulation were compared with theoretical models.

Mathematical models for describing the dynamics of situational systems were further developed. A model for studying the stability of a dynamic stochastic system based on the Lyapunov stability principle in the form of a system of ordinary differential equations is proposed. An agent-oriented approach to the inclusion of the Lyapunov

stability model in the process of situational control of a dynamic stochastic system as a target project activity is proposed and investigated.

An ontological model of knowledge for use in the system of situational response to security incidents has been developed which allows providing convergence in a single knowledge base of architectural, risk-oriented and adaptive to vulnerabilities aspects of security management of computer information systems. The development of knowledge models for the management of security systems provides support for the safe operation of these systems in accordance with their purpose. The proposed metamodel for the integration of security models based on descriptive logic describes some aspects of security management activities. The key components of security management activities for the decision support system are identified and methods for their implementation are proposed. The proposed model of convergence of security services includes a set of concepts representing a combination of components of security management: an integrated risk model, an integrated model of security barriers, an integrated model of system vulnerabilities. The relationship between the concepts and categories of the knowledge base determines the semantics of security management, and the functions of interpretation determine the rules of superposition of constituent models.

The results of theoretical and experimental research, as well as the developed methods, have found practical use and implementation in the Institute of Mathematical Machines and Systems Problems of NAS of Ukraine (Kyiv), SC «S&T UKRAINE» (Kyiv), Scientific Research Center for Disaster Risk Reduction of the Department of Information Technology and Communications of the University of National and World Economy (Sofia, Bulgaria). In addition, the results of the dissertation are implemented into the educational process of the National University of Life and Environmental Sciences of Ukraine (Kyiv), and Kamianets-Podilskyi National University named after Ivan Ohienko (Kamianets-Podilskyi) to improve theoretical and methodological support and in the performance of students' qualifying works.

Keywords: computer system architecture, situational control system, perceptual cycle, convergent system, system architecture optimization, intelligent agent.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Морозов А.О., Баран Л.Б., Вишневський В.В., Коваленко О.Є., Малишев О.В. Розділ 6. Ситуаційний центр як засіб колективної роботи законотворців. *Електронний парламент України: досвід створення. Наукове видання* / за заг. ред. С.О. Довгого. Київ : Логос, 2015. С. 395-441. ISBN: 978-617-7361-07-6.

2. Kovalenko O., Velevev D. Ad-Hoc Architecture of Systems for Disaster Risk Management. *Information Technology in Disaster Risk Reduction. IFIP Advances in Information and Communication Technology* / Y. Murayama, D. Velevev, P. Zlateva (eds.). 2020. Vol. 575. Springer, Cham. P. 135–145. DOI: 10.1007/978-3-030-48939-7_12. (Scopus Q3, DBLP). ISSN 1868-4238, ISSN 1868-422X (electronic), ISBN 978-3-030-48938-0, ISBN 978-3-030-48939-7 (eBook).

3. Kovalenko O., Visnevsky V., Kosolapov V. Models of information processing optimization for technical interoperability in a network of distributed situational centers. *CEUR Workshop Proc.* 2020. Vol. 2608. P. 426–435. (Scopus, DBLP). ISSN 1613-0073.

4. Коваленко О. Є., Косолапов В.Л. Моделі оптимізації завантаження мережі розподілених ситуаційних центрів. *Математичне та комп'ютерне моделювання. Технічні науки*. 2020. Вип. 21. С. 102–113. DOI: 10.32626/2308-5916.2020-21.102-113. (BASE, PKP Index, ICI, SIS, Academic Resource Index (ResearchBib), Google Scholar). ISSN 2308-5916.

5. Коваленко О. Є., Косолапов В.Л. Модель стійкості агентної системи ситуаційного управління. *Математичні машини і системи*. 2020. № 3. С. 93–104. DOI: 10.34121/1028-9763-2020-3-93-104. (Google Scholar). ISSN 1028-9763.

6. Коваленко О. Є. Онтологія та модель трансформації інформації в ситуаційних агентних системах. *Електронне моделювання*. 2020. Т. 42, № 5. С. 5–23. DOI: 10.15407/emodel.42.05.005. (Index Copernicus International, CrossRef, Ulrich's Periodicals Directory, Google Scholar, Cambridge Scientific Abstracts (CSA), Computer

and Information Systems Abstracts, INIS Collection, e-LIBRARY, BIHITI PAH). ISSN 0204-3572 (print), ISSN 2616-9525 (online).

7. Коваленко О. Є. Принципи інженерії ситуаційних систем. *Математичні машини і системи*. 2019. № 4. С. 65–78. DOI: 10.34121/1028-9763-2019-4-65-78. (Google Scholar). ISSN 1028-9763.

8. Kovalenko O.E. Knowledge Based Design of Convergent Systems of Situational Management. *Математичні машини і системи*. 2019. № 3. С. 67–74. DOI: 10.34121/1028-9763-2019-3-67-74. (PIHЦ, Google Scholar). ISSN 1028-9763.

9. Коваленко О. Є. Моделювання навантаження мережної інфраструктури систем ситуаційного управління. *Математичні машини і системи*. 2019. № 2. С. 101–110. (PIHЦ, Google Scholar). ISSN 1028-9763.

10. Власова Т.М., Коваленко О.Є., Косолапов В.Л. Організаційно-інформаційний сервіс для технології прийняття рішень у ситуаційному центрі шляхом проведення нарад. *Математичні машини і системи*. 2019. № 1. С. 68–79. (Google Scholar). ISSN 1028-9763.

11. Коваленко О. Є. Композиційна конвергенція інформаційних технологій у системах ситуаційного управління на основі моделі знань предметної області. *Математичне моделювання в економіці*. 2019. Т. 16, № 2. С. 39–44. DOI: 10.35350/2409-8876-2019-15-2-40-45. (Index Copernicus, Academic Resource Index (ResearchBib), Google Scholar). ISSN 2409-8876 (print), ISSN 2663-9068 (online).

12. Коваленко О.Є. Модель оцінки ефективності архітектури конвергентної системи ситуаційного управління. *Моделювання та інформаційні технології: зб. наук. праць ІПМЕ ім. Г.Є. Пухова НАН України*. 2019. Вип. 87. С. 11–14. DOI: 10.5281/zenodo.3612232. ISSN 2309-7647. (Google Scholar). ISSN 2309-7647.

13. Коваленко О.Є. Системна інженерія та життєвий цикл систем. *Електронне моделювання*. 2018. Т. 40, № 6. С. 61-82. DOI: 10.15407/emodel.40.06.061. (Index Copernicus International, CrossRef, Ulrich's Periodicals Directory, Google Scholar, Cambridge Scientific Abstracts (CSA), Computer and Information Systems Abstracts, INIS Collection). ISSN 0204-3572 (print), ISSN 2616-9525 (online).

14. Коваленко О.Є. Моделі агентно-орієнтованих систем ситуаційного управління. *Математичні машини і системи*. 2018. № 2. С. 96–102. (*Google Scholar*). ISSN 1028-9763.

15. Kovalenko O. E., Kovalenko T. Knowledge Model for Decision Support System Security Management. *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2016)*: proc. (Sofia, 19 February 2019). Sofia, 2019. P. 165–170. (*EBSCOHost, ProQuest, Google Scholar*). ISSN 2367-7635 (print), ISSN 2367-7643 (online).

16. Kovalenko O. E. The Formalization of Organizational Support Creation for Systems of Situational Management. *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE-2015)*: proc. (Sofia, 15 August 2016). Sofia, 2016. P. 292–301. (*EBSCOHost, ProQuest, Google Scholar*). ISSN 2367-7635 (print), ISSN 2367-7643 (online).

17. Коваленко О.Є. Побудова багаторівневих мультиагентних інформаційних систем, заснованих на знаннях. *Information Technology and Security*. 2016. Т. 4, № 2 (7). С. 146–154. (*Index Copernicus, Google Scholar*). ISSN 2411-1031 (print), ISSN 2518-1033 (online).

18. Коваленко О.Є. Застосування хмарних технологій при побудові систем ситуаційного управління. *Information Technology and Security*. 2015. Т. 3, № 1. С. 11–18. (*Index Copernicus, Google Scholar*). ISSN 2411-1031 (print), ISSN 2518-1033 (online).

19. Kovalenko Oleksii E. Knowledge Models for Organizational Maintenance of Situation Centers. *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE)*: proc. (Sofia, 15 September 2015). Sofia, 2015. P. 241–248. (*EBSCOHost, ProQuest, Google Scholar*). ISSN 2367-7635 (print), ISSN 2367-7643 (online).

20. Коваленко О. Є. Застосування модальної логіки при прийнятті рішень на моделях знань. *Математичне та комп'ютерне моделювання. Технічні науки*: зб.

наук. праць. Кам'янець-Подільський, 2012. Вип. 6. С. 106–112. (BASE, PKP Index, NSD, Google Scholar). ISSN 2308-5916.

21. Коваленко О.Є. Оптимізація архітектур модульних систем електронного навчання. *Математичне та комп'ютерне моделювання. Технічні науки*: зб. наук. праць. Кам'янець-Подільський, 2008. Вип. 1. С. 95–99. (BASE, PKP Index, NSD, Google Scholar). ISSN 2308-5916.

22. Kovalenko A.E., Velevev D.G. Compositional Model of Design Process in the Problem-Oriented CAD of Applied Software. *Engineering Simulation*. 1998. Vol. 16, Issue 1. P. 129–133. Gordon & Breach Science Publ Inc, Newark, NJ, United States. ISSN 10631100. (Scopus).

23. Kovalenko, A.E., Velevev, D.G. Computer-aided programming system for intelligent measurement systems in electronics. *Engineering Simulation*. 1995. Vol. 12, Issue 4. P. 642–649. Gordon & Breach Science Publ. Inc., Newark, NJ, United States. ISSN 10631100. (Scopus).

Наукові праці, які засвідчують апробацію матеріалів дисертації:

24. Kovalenko O., Vishnevsky V., Kosolapov V. Towards Creating the Network of Situational Governance Centers and Decision Making Technologies in Distributed Environments. *15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET-2020)*: proc. (07 May 2020). 2020. Vol. May. P. 540–545. DOI: 10.1109/TCSET49122.2020.235491. (Scopus). ISBN 978-1-7281-5566-1 (electronic).

25. Kovalenko O. Information Taxonomy and Ontology for Situational Management. *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT-2018)*: proc. (7 November 2018). 2018. Vol. 2, Article number 8526723. P. 94–97. DOI: 10.1109/STC-CSIT.2018.8526723. (Scopus, Web of Science Core Collection). ISBN 978-1-5386-6464-3 (electronic).

26. Kovalenko O., Kovalenko T. Knowledge Model and Ontology for Security Services. *2018 IEEE 1st International Conference on System Analysis and Intelligent*

Computing (SAIC-2018): proc. (31 October 2018). 2018. Article number 8516875. DOI: 10.1109/SAIC.2018.8516875. (*Scopus, Web of Science Core Collection*). ISBN 978-1-5386-7196-2 (electronic).

27. Kosolapov, V., Kovalenko, O. Agent Based Modelling Support for Systems of Situational Management. *2018 IEEE 1st International Conference on System Analysis and Intelligent Computing (SAIC-2018)*: proc. (31 October 2018). 2018. Article number 8516818. DOI: 10.1109/SAIC.2018.8516818. (*Scopus, Web of Science Core Collection*). ISBN 978-1-5386-7196-2 (electronic).

28. Vlasova T., Kovalenko O. and Kosolapov V. Organizational-Information Technology for Providing and Decisions Making in Situational Management. *14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET-2018)*: proc. (10 April 2018). 2018. Vol. April. P. 152–157. DOI: 10.1109/TCSET.2018.8336176. (*Scopus*). ISBN 978-1-5386-2556-9 (electronic).

29. Kovalenko O., Kosolapov V. Optimization of Loading of a Network of Distributed Situational Centers. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. 9-ї Міжнар. наук. конф. (м. Кам'янець-Подільський, 14–15 травня 2020 р.). Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2020. С. 89–91.

30. Коваленко О. Є. Архітектурний підхід до управління життєвим циклом систем управління інформаційною безпекою на основі моделей знань. *Кібербезпека енергетики* : зб. пр. конф. Київ : Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, 2019. С. 5–7.

31. Коваленко О.Є. Елементи архітектури систем кібер-енергетики. *Безпека енергетики в епоху цифрової трансформації* : програма та матеріали наук.-практ. конф. Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 20 грудня 2019 р.). Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2019. С. 69–71.

32. Коваленко О.Є. Онтологія та трансформація інформації в системах ситуаційного управління. *Сучасні проблеми інформатики в економіці, управлінні,*

освіті та подоланні наслідків Чорнобильської катастрофи : матеріали XVII Міжнар. наук. семінару (м. Київ – оз. Світязь, 2–6 липня 2018 р.). Київ : Національна академія управління, 2018. С. 12–16.

33. Kovalenko O. E. On Consolidation of Information in Systems of Situational Management. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. 8-ї Міжнар. наук. конф., присвяченої 100-річчю Національної академії наук України та 100-річчю Кам'янець-Подільського національного університету імені Івана Огієнка. Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2018. С. 118–119.

34. Коваленко О. Є., Коваленко Т. О. Конвергенція та конфігурування інформаційних технологій в системах ситуаційного управління. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. 8-ї Міжнар. наук. конф., присвяченої 100-річчю Національної академії наук України та 100-річчю Кам'янець-Подільського національного університету імені Івана Огієнка. Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2018. С. 122–123.

35. Коваленко О.Є. Інформаційна технологія консолідації даних в системах ситуаційного управління. *Сучасні проблеми інформатики в економіці, управлінні, освіті та подоланні наслідків Чорнобильської катастрофи* : матеріали XVI Міжнар. наук. семінару (м. Київ – оз. Світязь, 3–7 липня 2017 р.). Київ : Національна академія управління, 2017. С. 13–17.

36. Коваленко О.Є. Ситуаційне управління як проектна діяльність. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. одинадцятої дистанційної наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2017. С. 28–31.

37. Velev D.G., Kovalenko O.E. Research on the Applicability of Virtual Reality in Disaster Risk Reduction. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. одинадцятої дистанційної наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2017. С. 32–35.

38. Коваленко Т.О., Коваленко О.Є. Модель знань для управління безпекою інформаційної системи. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. одинадцятої дистанційної наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2017. С. 113–116.

39. Коваленко О.Є. Обробка модальностей при прийнятті рішень у системах ситуаційного управління. *Теорія прийняття рішень* : праці VIII міжнар. школи-семінару (м. Ужгород, 26 вересня – 1 жовтня 2016 р.). Ужгород : УжНУ, 2016. С. 139–140.

40. Коваленко О.Є. Застосування онтологій в системах ситуаційного управління. *Сучасні проблеми інформатики в управлінні, економіці, освіті та подоланні наслідків Чорнобильської катастрофи* : матеріали XV Міжнародного наукового семінару, (м. Київ – оз. Світязь, 4–8 липня 2016 року). Київ : Національна академія управління, 2016. С. 84–89.

41. Коваленко О.Є. Технологічні моделі ситуаційного управління. *Проблеми виведення з експлуатації об'єктів ядерної енергетики та відновлення оточуючого середовища INUDECО'16* : зб. матеріалів першої міжнар. конф. (м. Славутич, 25–27 квітня 2016 р.). Славутич : Славутиська філія НТУУ «КПІ», 2016. С. 166–168.

42. Коваленко О. Є., Коваленко Т. О. Елементи модельно-орієнтованого підходу технології розробки систем ситуаційного управління. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. VII міжнар. наук. конф. (м. Кам'янець-Подільський, 20–21 квітня 2016 р.). Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2016. С. 97–98.

43. Коваленко О.Є. Формалізація діяльності систем ситуаційного управління. *Сучасні проблеми інформатики в управлінні, економіці та освіті* : матеріали XIV-го міжнар. наук. семінару (м. Київ – с. Світязь, 29 червня – 3 липня 2015 р.). Київ : СІК ГРУП Україна, 2015. С. 119–123.

44. Коваленко О.Є. Стандартизація формального опису системної архітектури ситуаційних центрів. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доповідей десятої дистанційної наук.-практ. конф. з міжнар. участю.

Київ: Інститут проблем математичних машин і систем НАН України, 2015. С. 111–114.

45. Коваленко О.Є. SWOT-аналіз технологій ситуаційного управління. *Обчислювальний інтелект (результати, проблеми, перспективи)* : праці Міжнар. наук.-практ. конф. (м. Київ – Черкаси, 12–15 травня 2015 р.). Черкаси : видавець Чебаненко Ю., 2015. С. 354.

46. Коваленко О.Є. Моделі організаційного забезпечення інформаційних технологій ситуаційних центрів. *Сучасні проблеми інформатики в управлінні, економіці та освіті* : матеріали XIII-го міжнар. наук. семінару (м. Київ – с. Світязь, 30 червня – 4 липня 2014 р.). Київ: Національна академія управління, 2014. С. 105–110.

47. Kovalenko O. E. The Architecture of Agent-Oriented Situation Management System. *Proceedings of 3rd International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2013)* (Sofia, Bulgaria, December 6 – 7th 2013). Sofia, Bulgaria : University of National and World Economy (UNWE), 24 July 2014. P. 621–627. (EBSCOHost, ProQuest).

48. Коваленко О.Є. Інтеграція знань в агентно-орієнтовані системи підтримки прийняття рішень. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. VI Міжнар. наук. конф. (м. Кам'янець-Подільський, 4–5 квітня 2014 р.). Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2014. С. 77.

49. Коваленко О.Є. Модель організації віртуальних інформаційних систем для розподіленої колективної обробки інформації. *Обчислювальний інтелект (результати, проблеми, перспективи)* : матеріали 2-ї Міжнар. наук.-техн. конф. (м. Черкаси, 14–18 травня 2013 р.). Черкаси: Маклаут, 2013. С. 367.

50. Коваленко О.Є. Архітектура сервісів організаційного забезпечення ситуаційного центру. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. дев'ятої дистанційної наук.-практ. конф. з між нар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2013. С. 60–63.

51. Kovalenko O. Agents Knowledge Models for Situation Management Systems. *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE-2012)* (Sofia, Bulgaria , 5–6 October 2012). Sofia, Bulgaria : University of National and World Economy (UNWE), 11 January 2013. P. 125–131. ISBN 978-954-92247-4-0.

52. Kovalenko O. E., Velev D. G. Integration of Information Services in Agent-Oriented Situation Management System. *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE-2012)* (Sofia, Bulgaria, 5–6 October 2012). Sofia, Bulgaria : University of National and World Economy (UNWE), 11 January 2013. P. 132–135. ISBN 978-954-92247-4-0.

53. Коваленко О.Є. Організація агентно-орієнтованих систем ситуаційного управління. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ: Інститут проблем математичних машин і систем НАН України, 2012. С. 63–65.

54. Kovalenko O. E. Functional Approach in the Simulation Modeling of the Problem-Oriented Information Systems. *Комунікації, інформаційні технології и статистика. Актуални проблеми на теорията и практиката*: колективна монографія. София : Университетско издателство «Стопанство», 2011. С. 278–281.

55. Коваленко А.Е. Построение систем поддержки принятия решений на принципах облачных вычислений. *Proceedings of the International Conference on Application of Information and Communication Technology in Economy and Education (ICAICTEE-2011)* (Sofia, Bulgaria, 2–3 December, 2011). Sofia, Bulgaria : University of National and World Economy, 2011. P. 475–482.

56. Коваленко О. Є. Елементи моделі цілеспрямованих систем орієнтованих на знання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2011. С. 172–174.

57. Коваленко О.Є., Коваленко А.О. Аспекти функціонування інтелектуальних систем, заснованих на знаннях. *Обчислювальний інтелект*

(результати, проблеми, перспективи) : матеріали 1-ї Міжнар. наук.-техн. конф. (м. Черкаси, 10–13 травня 2011 р.). Черкаси : Маклаут, 2011. С. 95–96.

58. Коваленко О.Є. Організація комп'ютерного навчання з використанням моделей середовища навчання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2010. С. 160–163.

59. Коваленко О.Є. Оптимізація процесів електронного навчання на основі імітаційних моделей. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : збірник наук. праць за матеріалами четвертої міжнар. наук. конф. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2010. С. 88–91.

60. Коваленко О.Є. Моделювання процедур експертного оцінювання у навчальних системах *Комп'ютерна математика в науці, інженерії та освіті (CMSEE-2010)* : матеріали Четвертої міжнар. наук.-техн. конф. (м. Полтава, 1–31 жовтня 2010 р.). Полтава: ПолтНТУ, 2010. С. 19.

61. Коваленко О.Є. Інтегральна модель засвоєння знань у навчальних системах. *Интегральные уравнения – 2009 = Integral equations – 2009* : сб. тезисов конф. (г. Киев, 26–29 января 2009 г.). Киев : ИПМЭ им. Г.Е. Пухова НАН Украины, 2009. С. 91–92.

62. Коваленко А., Велев Д. Интеграция приложений электронного обучения в среде RIA/SOA. *Приложна информатика и статистика – съвременни подходи и методи* : доклади от Междунар. науч. конф. (г. Равда, България, 25–26 септември 2009 г.). София : Университетско издателство «Стопанство», 2009. С. 44–45.

63. Коваленко О.Є. Онтологічна модель системи прийняття рішень методом голосування. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2009. С. 51–54.

64. Коваленко О.Є. Семантично-функціональний підхід до побудови багаторівневих систем електронного навчання. *Системи підтримки прийняття*

рішень. *Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2008. С. 166–169.

65. Коваленко О.Є. Стандартизація навчальних курсів для глобального освітнянського простору. *Проблеми безперервного професійного розвитку лікарів і провізорів* : зб. пр. наук.-метод. конф. з міжнар. участю. Київ : Національна медична академія післядипломної освіти імені П.Л. Шупика, 2007. С. 512–514.

66. Коваленко О.Є., Вознюк П.Є. Керування конфігурацією модульних систем електронного навчання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2007. С. 137–140.

67. Kovalenko O. Multilayer model of e-learning systems. *Бизнес Информатика* : сб. доклади от междунар. науч. конф. по повод 40-та годишнина на специалност „Информатика“ (г. София, 11–12 Октомври 2007 г.). София : Университет за национално и световно стопанство, 2007. С. 116–120.

68. Коваленко О.Є. Логічна модель стандартизованого навчального курсу для модельно-керованого процесу розробки. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : зб. наук. пр. (за матеріалами міжнар. наук.-метод. конф.). Кам'янець-Подільський : Кам'янець-Подільський державний університет, 2006. С. 161–169.

69. Коваленко О.Є. Особливості розробки стандартизованих навчальних курсів для багаторазового використання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2006. С. 182–185.

70. Kovalenko O. Evaluation of e-learning deployment scale. *Управленски, информационни и маркетингови аспекти на икономическото развитие на балканските страни* : сб. доклади от междунар. науч. конф. (г. София, 3 ноември 2006 г.). София : Университет за национално и световно стопанство, 2006. С. 427–429.

71. Kovalenko O. Efficiency of Corporate Electronic Learning. *Управленски, информационни и маркетингови аспекти на икономическото развитие на*

балканските страни : сб. доклади от междунар. науч. конф. (г. София, 4 ноември 2005 г.). София : Университет за национално и световно стопанство, 2005. С. 197–208.

72. Коваленко О.Є. UML-подання моделі екстремального програмування для проектування систем безпеки. *Проблеми впровадження інформаційних технологій в економіці* : матеріали V міжнар. наук.-практ. конф. (м. Ірпінь, 15–17 травня 2004 р.). Ірпінь : Академія ДПС України, 2004. С. 499–502.

73. Коваленко О. Є. Моделювання технологій адаптивного проектування інформаційних систем. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : зб. наук. пр. (за матеріалами Всеукр. наук.-метод. конф.) (м. Кам'янець-Подільський, 23–24 вересня 2004 р.). Кам'янець-Подільський: Кам'янець-Подільський державний університет, 2004. С. 111–115.

74. Коваленко О.Є., Антоненко В.М. Технологія програмування та захист інформаційних систем. *Проблеми впровадження інформаційних технологій в економіці* : матеріали IV міжнар. наук.-практ. конф. (м. Ірпінь, 15–17 травня 2003). Ірпінь : Академія ДПС України, 2003. С. 625–629.

Наукові праці, які додатково відображають наукові результати дисертації:

75. Коваленко О.Є. Комп'ютерний практикум з навчальної дисципліни «Системне інтегрування та адміністрування комп'ютерних мереж»: навч.-метод. посіб. Київ: Вид-во ІСЗЗІ НТУУ «КПІ», 2012. 188 с.

76. Коваленко О.Є. Захист інформації та програмних продуктів. Лабораторний практикум: навч. посіб. Київ : КСУ, 2007. 252 с.

77. Коваленко О.Є. Дослідження операцій : навч. посіб. для спеціальностей «Менеджмент організацій», «Фінанси», «Облік і аудит», «Маркетинг». Київ : КСУ-МГІ, 2005. 55 с.

78. Kovalenko O. E. Models and Means for Service Agents Orchestration in Situation Management Systems. *Actual Problems of Economics*. 2014. N 4 (154). P. 462–

467. (*Scopus, Index Copernicus, EBSCOhost, Ulrich's Periodicals Directory, EconLit, ABI/Inform (by ProQuest), Cabell's Directories*).

79. Коваленко О. Є. Координація взаємодії програмних агентів у системах ситуаційного управління. *Актуальні проблеми економіки*. 2012. № 12. С. 183–189. (*Scopus, Web of Science Core Collection, Index Copernicus, EBSCOhost, Ulrich's Periodicals Directory*).

80. Коваленко О. Є. Мультиагентна модель колективного прийняття рішень. *Актуальні проблеми економіки*. 2011. № 11. С. 279–285. (*Scopus, Web of Science Core Collection, Index Copernicus, EBSCOhost, Ulrich's Periodicals Directory*).

81. Коваленко О.Є. Стійкість результатів обробки експертних оцінок у системах підтримки прийняття рішень. *Актуальні проблеми економіки*. 2010. № 10. С. 177–181. (*Scopus, Web of Science Core Collection, Index Copernicus*).

82. Коваленко О.Є. Організація процедур прийняття рішень у ситуаційних центрах на основі голосування. *Актуальні проблеми економіки*. 2009. № 6. С. 289–293. (*Scopus, Web of Science Core Collection, Index Copernicus*).

83. Коваленко О.Є. Архітектура модульних систем дистанційного навчання. *Актуальні проблеми економіки*. 2007. № 12. С. 172–176.

84. Коваленко О.Є., Вишневський В.В. Організація системи дистанційного навчання на основі стандартних компонентів. *Актуальні питання фармацевтичної і медичної науки та практики*. Запоріжжя: Запорізький державний медичний університет, 2007. Вип. XVIII. С. 38–44.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	4
ВСТУП.....	6
РОЗДІЛ 1 МЕТОДОЛОГІЧНІ ОСНОВИ СТВОРЕННЯ ТА ЗАСТОСУВАННЯ КОНВЕРГЕНТНИХ СИСТЕМ СИТУАЦІЙНОГО УПРАВЛІННЯ.....	17
1.1 Системні засади побудови систем ситуаційного управління	17
1.2 Інженерія системи систем	39
1.3 Ситуаційне управління як продукт трансформації інформації засобами інформаційних технологій.....	45
1.4 Моделі ситуаційного управління.....	51
1.5 Система ситуаційного управління як кіберматична система	65
1.5 Конвергентний підхід при побудові комп'ютерних систем	71
1.5.1 Конвергенція знань і технологій.....	71
1.5.2 Конвергентні інфраструктури.....	72
1.5.3 Гіперконвергентні інфраструктури	75
1.5.4 Порівняння конвергентних і гіперконвергентних інфраструктур	76
1.6 Висновки до розділу 1.....	77
РОЗДІЛ 2 АСПЕКТИ ПОБУДОВИ КОНВЕРГЕНТНИХ СИСТЕМ СИТУАЦІЙНОГО УПРАВЛІННЯ	79
2.1 Динамічні моделі ситуаційного управління	79
2.2 Трансформація інформації в системах ситуаційного управління.....	83
2.3 Архітектурний аспект побудови систем ситуаційного управління	107
2.4 Технологічний аспект побудови систем ситуаційного управління	124
2.6 Висновки до розділу 2.....	127
РОЗДІЛ 3 ФОРМАЛІЗАЦІЯ АРХІТЕКТУРИ КОНВЕРГЕНТНОЇ ССУ	129
3.1 Формальна модель багаторівневої ССУ	129
3.2 Модель ситуаційного агента	132
3.3 Трансформація знань у багаторівневій ССУ	145

3.4 Модель інтероперабельності багаторівневої ССУ	158
3.5 Формальна модель конвергенції технологій ССУ	162
3.6 Ситуаційне управління інформаційною безпекою на основі конвергенції моделей знань.....	170
3.8 Висновки до розділу 3.....	176
РОЗДІЛ 4 МЕТОДИ СТРУКТУРНОЇ ОПТИМІЗАЦІЇ КОНВЕРГЕНТНИХ АРХІТЕКТУР ССУ	179
4.1 Загальна постановка задачі оптимізації конвергентної ССУ	179
4.2 Однокритерійна оптимізація конвергентної ССУ	180
4.3 Багатокритерійна оптимізація конвергентної ССУ	183
4.4 Приклад застосування методу багатокритерійної оптимізації.....	186
4.5 Висновки до розділу 4.....	193
РОЗДІЛ 5 МОДЕЛІ ДИНАМІЧНОЇ ОПТИМІЗАЦІЇ КОНВЕРГЕНТНИХ ССУ	196
5.1 Модель діяльності конвергентної ССУ.....	196
5.2 Метод аналізу динамічної стійкості конвергентної ССУ	198
5.3 Метод забезпечення технічної інтероперабельності конвергентної ССУ	204
5.4 Висновки до розділу 5.....	213
РОЗДІЛ 6 ЗАСОБИ ПІДТРИМКИ ПОБУДОВИ КОНВЕРГЕНТНИХ СИСТЕМ.....	215
6.1 Засоби оптимізації конфігурацій конвергентних систем.....	215
6.2 Архітектурні та інформаційні моделі ситуаційної діяльності в ССУ.....	223
6.2.1. Моделі організації роботи груп у ССУ	228
6.2.2 Методи та моделі колективного прийняття рішень в агентних ССУ	232
6.3 Аналіз завантаження інфраструктури ССУ	251
6.4 Висновки до розділу 6.....	264
ВИСНОВКИ.....	266
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	269
ДОДАТКИ.....	302
Додаток А Акти впровадження.....	303

Додаток Б Список публікацій здобувача за темою дисертації	310
Додаток В Модулі системи управління проєктами конвергенції	323
Додаток Д Інформаційне забезпечення системи управління проєктами конвергенції	329

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

АОССУ	– агентно-орієнтовані середовища ситуаційного управління.
ЖЦ	– життєвий цикл.
ІКС	– інформаційно-комунікаційні системи.
ІКТ	– інформаційно-комунікаційна технологія.
КЗТС	– конвергенція знань і технологій на благо суспільства.
КЛС	– кіберлюдська система.
МАС	– мультиагентні системи.
МОСІ	– модельно-орієнтована системна інженерія.
ОПР	– особа, яка приймає рішення.
ПЗ	– програмне забезпечення.
СУ	– ситуаційне управління.
ССУ	– система ситуаційного управління.
СЦ	– ситуаційний центр.
ЦОД	– центр обробки даних.
ЦС	– цільова система.
ADL	– Architecture Description Language.
ARCADIA	– ARChitecture Analysis & Design Integrated Approach.
BKCASE	– Body of Knowledge and Curriculum to Advance Systems Engineering.
CHS	– Cyber-Human Systems.
CKTS	– Convergence of Knowledge and Technology for Society.
DIKW	– Data, Information, Knowledge, Wisdom.
DoDAF	– Department of Defense Architectural Framework.
EIRA	– European Interoperability Reference Architecture.
FFBD	– Functional Flow Block Diagram.
I-SDKW	– Information as Signal-Data-Knowledge-Wisdom.
IDEF0	– Integration Definition for Functional Modeling.
INCOSE	– International Council on Systems Engineering.

MBSE	– Model-Based Systems Engineering.
MDA	– Model Driven Architecture.
MODAF	– Ministry of Defence Architecture Framework.
OMG	– object management group.
OWL	– Ontology Web Language.
OPM	– Object-Process Methodology.
PMBoK	– Project Management Body of Knowledge.
SEBoK	– System Engineering Body of Knowledge.
SoI	– System on Interest.
SoS	– System of Systems.
SysML	– Systems Modeling Language.
UAF	– Unified Architectural Framework.
UML	– Unified Modelling Language.
UPDM	– Unified Profile for DoDAF/MODAF.

ВСТУП

Актуальність теми. Сучасні інформаційні системи управління є невід’ємною частиною засобів підтримки ефективної діяльності у різних предметних сферах. Непередбачуваність, випадковість та мінливість середовища у різних сферах діяльності потребує усвідомлення ситуаційного характеру процесів і застосування адекватних підходів та принципів у процесі здійснення такої діяльності. Таким підходом є ситуаційний підхід для організації діяльності в рамках відповідних ситуаційних систем. Ситуаційно-орієнтована діяльність реалізується в рамках перцептивного циклу і включає етапи емпіричного усвідомлення стану середовища (цільової предметної області), побудови його моделі та застосування цієї моделі при формуванні раціональної поведінки у середовищі на основі періодичного оновлення усвідомлення поточного стану цільової предметної області. Оскільки ситуаційне управління загалом стосується непередбачуваних випадкових подій, то цільова ситуаційна система повинна бути адаптована до вирішення специфічних задач конкретної проблемної ситуації. Така адаптація може бути реалізована на основі конвергенції інформаційних технологій у ситуаційних системах. Методологія ситуаційного управління (СУ) передбачає розробку і впровадження універсальних методів обробки інформації при формуванні та впровадженні ситуаційної діяльності на основі формалізованих описів цільової предметної області з використанням моделей знань.

Дослідження проблем ситуаційного управління та ситуаційної діяльності у світі та Україні мають вже більш ніж півсторічну історію. Існуючі підходи і теоретичні засади створення комп’ютеризованих систем ситуаційного управління та їх компонентів стосуються або окремих задач ситуаційного управління, або окремих рівнів та процесів їх функціонування і є специфічними для кожної реалізації і можуть бути обмежено застосовані при масштабуванні та тиражуванні таких систем. Аналіз публікацій за тематикою побудови систем ситуаційного управління

показує фрагментарність досліджень і пропонованих технічних рішень, що обмежує їх застосування в різних задачах ситуаційного управління.

Однією з важливих проблем створення сучасних систем ситуаційного управління є забезпечення оперативного й ефективного поєднання спроможностей систем у складі цільової системи або системи систем на основі конвергенції комп'ютеризованих засобів підтримки ситуаційної діяльності в різних проблемних областях. Конвергенція здійснюється через глибоку інтеграцію знань, засобів та усіх раціональних видів людської діяльності для досягнення спільної мети, можливостей отримання відповідей на нові питання стосовно зміни відповідної технічної або соціальної системи. Такі зміни у відповідній системі відкривають нові перспективи, шляхи та можливості дивергенції у процесі ситуаційної діяльності.

Ефективне вирішення проблеми створення конвергентних систем ситуаційного управління може бути досягнуто шляхом уніфікації математичних моделей для ситуаційних систем різного призначення з можливостями врахування особливостей окремих ситуаційних задач, що розв'язуються з використанням таких систем. Врахування особливостей системи ситуаційного управління цільового призначення може бути забезпечено на основі використання формальної предметної бази знань. Для розв'язання вказаної проблеми доцільним є застосування модельно-орієнтованого підходу на основі конвергенції онтологічних, об'єктних, процесних, архітектурних, функціональних та агентних моделей.

Суттєвий внесок у вирішення проблем розробки теоретичних засад, методів і засобів створення універсальних та спеціалізованих комп'ютерних систем різного призначення, комп'ютерних систем з адаптивною архітектурою, технологій системної інтеграції внесли Глушков В.М., Морозов А.О., Поспелов Д.А., Палагін О.В., Згуровський М.З., Верлань А.Ф., Додонов О.Г., Петренко М.Г., Петренко А.І., Голуб С.В., Опанасенко В.М., Кривий С.Л., Стасюк О.І., Самойлов В.Д., Субботін С.О., Лисецький Ю.М., Городецький В.І., Тимченко А.А., Павлов О.А., Томашевський В.М., Теленик С.Ф., Ролік О.І., Анохин П.К., Mihajlo D. Mesarović, Yasuhiko Takahara, Gabriel Jakobson, John Buford, Lundy Lewis, Keith Devlin, Michael

Wooldridge, Mark Burgess, Svyatoslav Kotusev, Scott A. Bernard, Vivek Singh, Kurt Sandkuhl, Hasan Koç, Janis Stirna, William S. Bainbridge, Mihail C. Roco та ін.

Таким чином, у зв'язку з розширенням сфери застосування, ускладненням функцій і архітектур ситуаційних систем актуальною є науково-технічна проблема розвитку теоретичних засад побудови конвергентних комп'ютерних систем ситуаційного управління шляхом розробки нових моделей і розвитку методів для забезпечення ефективності таких систем з урахуванням вимог і обмежень сфери застосування та використання моделей знань.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження з теми дисертаційної роботи проведені в Інституті проблем математичних машин і систем НАН України та Інституті проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України відповідно до плану науково-дослідних робіт НАН України при виконанні фундаментальних і прикладних держбюджетних тем: НДР «Методологічні засади створення мережі ситуаційних центрів органів державної влади України та інформаційних технологій підтримки прийняття рішень у розподіленому середовищі», («Дельта»), 2015–2018 рр., державний реєстраційний № 0115U003014 (відповідальний виконавець); НДР «Комплексні, масштабовані програмно-технічні платформи ситуаційних центрів вищого державного рівня», («СЦ-Інтеграція»), 2014–2015 рр., державний реєстраційний № 0114U000024; НДР «Розробка теоретичних основ побудови нових моделей, алгоритмів і розподілених інформаційних технологій підтримки прийняття рішень у системах електронного урядування», («Е-Система»), 2011–2014 рр., державний реєстраційний № 0111U002509; НДР «Розробка нових архітектурних рішень для програмно-технічних комплексів систем підтримки прийняття рішень на вищому державному рівні», («СЦ-Рада»), 2011–2013 рр., державний реєстраційний № 0111U003107; НДР «Комплексна система підтримки прийняття рішень у розподілених середовищах», («Регламент»), 2008–2010 рр., державний реєстраційний № 0107U008885; НДР «Створення інтелектуальних інформаційних технологій, моделей та мовних засобів для СППР», (шифр «Експерт»), 2000–2006 рр., державний реєстраційний № 0102U002700; НДР «Розробка програмно-технічних засобів супроводу роботи

комітетів Верховної Ради України у складі інтегрованої інформаційно-аналітичної системи «Електронний Парламент», («Рада-КС»), 2013 р., державний реєстраційний № 0113U007157с; ГД «Розробка організаційного забезпечення та діючого макетного зразка робочого місця народного депутата України для Ситуаційного центру як зали засідань профільних комітетів Верховної Ради України у складі інтегрованої інформаційно-аналітичної системи «Електронний Парламент», 2012 р., державний реєстраційний № 0112U005475; ГД «Розробка концепції, методології та архітектури електронних робочих місць народного депутата України в сесійній залі, залі засідань комітету, на погоджувальній раді, у службовому офісі» («Рада-4К»), 2011 р., державний реєстраційний № 0111U005970; договір № 48 від 25.09.2003 «Автоматизована система підготовки персоналу ЦОРКС і складу оперативних груп ЦОРКС» (автор – відповідальний виконавець).

Тематика дисертаційного дослідження підпадає під пріоритетний тематичний напрям фундаментальних досліджень і науково-технічних розробок на період до 2020 р. «Технології та інструментальні засоби електронного урядування. Інформаційно-аналітичні системи, системи підтримки прийняття рішень. Ситуаційні центри» затверджених постановою Кабінету Міністрів України № 556 від 23.08.2016.

Мета і завдання дослідження. Метою дисертаційної роботи є створення нових та вдосконалення існуючих моделей і методів побудови конвергентних проблемно-орієнтованих комп'ютерних інформаційних систем із високими функціональними та структурними показниками для розв'язування задач ситуаційного управління.

Для досягнення поставленої мети в роботі вирішувались такі завдання:

1. Провести аналіз та узагальнення класифікацій ситуаційних систем для визначення особливостей їх функціонування.

2. Провести аналіз моделей циклів ситуаційної діяльності для визначення множини технологій її проведення.

3. Провести аналіз архітектурних моделей побудови комп'ютерних інформаційних систем для визначення повноти представлення аспектів моделі ситуаційних систем.

4. Провести аналіз моделей трансформації інформації в системах ситуаційного управління для визначення особливостей представлення та використання інформації в циклі ситуаційного управління.

5. Провести аналіз моделей агентних систем для визначення особливостей застосування агентного підходу при побудові систем ситуаційного управління.

6. Розвинути конвергентний підхід для побудови систем ситуаційного управління.

7. Розробити онтологічні моделі представлення знань ситуаційного управління для використання при побудові систем ситуаційного управління.

8. Розробити модель ситуаційного агента як уніфікованого компонента конвергентних систем ситуаційного управління.

9. Розробити керовану знаннями модель конвергенції компонентів у складі системи ситуаційного управління.

10. Розвинути оптимізаційні методи конвергенції архітектурних компонентів у системах ситуаційного управління.

11. Розвинути методи аналізу динамічної стійкості конвергентних ситуаційних систем під час функціонування.

12. Розробити фрагменти конвергентних систем ситуаційного управління для аналізу та практичного використання.

Об'єктом дослідження є архітектура систем і процесів функціонування комп'ютерних систем для вирішення проблем ситуаційного управління.

Предметом дослідження є моделі, методи та засоби побудови ефективних конвергентних комп'ютерних систем для вирішення задач ситуаційного управління з використанням моделей знань.

Методи дослідження. У дисертаційній роботі було використано методи:

– загальної теорії систем – для визначення принципів побудови ситуаційних систем;

- теорії ієрархічних багаторівневих систем – для побудови концептуальної моделі та архітектурних моделей комп'ютеризованих ситуаційних систем;
- математичного та імітаційного моделювання динамічних систем – для дослідження моделей ситуаційних систем та аналізу динаміки їх функціонування;
- логічних теорій першого порядку, зокрема, дескрипційні та модальні логіки – для формалізації знань предметної області та процесів ситуаційного управління в контексті заданих архітектур;
- ситуаційної теорії – для опису контексту побудови ситуаційних систем;
- інженерії знань та онтології – для створення прикладних онтологічних моделей знань ситуаційних систем;
- теорії агентних систем – для створення моделей агентно-орієнтованих ситуаційних систем;
- теорії операцій – для створення моделей оптимізації архітектури конвергентних систем.

Наукова новизна отриманих результатів полягає в розвитку загальної теорії комп'ютерних систем агентного типу шляхом розробки нових моделей та на їх основі розвитку методів побудови конвергентних комп'ютерних систем ситуаційного управління, що забезпечує структурну ефективність і динамічну стійкість таких систем.

У результаті проведених досліджень було отримано такі наукові результати.

Вперше:

- запропоновано узагальнену класифікацію ситуаційних систем, яка враховує особливості цільового призначення ситуаційної системи і є основою для побудови спеціалізованих моделей знань предметних областей ситуаційної діяльності;
- розроблено модель трансформації інформації в когнітивному циклі ситуаційної діяльності, яка забезпечує безрозривне поєднання перетворень між різними категоріями інформації й забезпечує її адекватне використання в різних видах моделей на кожному етапі циклу ситуаційної діяльності;
- розроблено цілісну модель ситуаційної обізнаності на основі співставлення етапів циклу ситуаційного управління з методами логічного умовиводу, що

відповідають цільовому призначенню кожного етапу і дозволяють формувати формалізовані знання про ситуацію;

- розроблено модель ситуаційного агента для реалізації процесу ситуаційного управління шляхом трансформації категорій інформації, що узагальнює моделі інтелектуальних агентів і дає змогу будувати конвергентні системи ситуаційного управління як композицію уніфікованих ситуаційних агентів;

- розроблено метод композиційної конвергенції компонентів системи ситуаційного управління з використанням моделей знань, що дає формалізоване представлення процесу побудови цільових ситуаційних систем на основі вимог до них з урахуванням особливостей ситуаційного управління;

- розроблено технологічні моделі для побудови компонентів конвергентних систем ситуаційного управління для окремих етапів ситуаційного управління, що є основою для типізації таких систем;

- розроблено конвергентну онтологічну модель знань для застосування у системі ситуаційного управління інформаційною безпекою, що поєднує онтологічні моделі стандартів, загроз, уразливостей та архітектур безпеки.

Удосконалено:

- оптимізаційні методи оцінки структурної ефективності архітектури конвергентної системи ситуаційного управління, що, на відміну від евристичної моделі розробки, дає змогу формально обґрунтувати архітектурні рішення для цільових систем;

- оптимізаційні методи оцінки динамічної стійкості конвергентних систем ситуаційного управління на основі узагальнених показників, що дозволяє прогнозувати їх спроможності у процесі функціонування;

- моделі аналізу подіє-орієнтованих систем на основі теорії масового обслуговування, що дозволяє прогнозувати поведінку таких систем у процесі функціонування.

Отримала подальшого розвитку теорія модельно-орієнтованого проектування складних комп'ютеризованих систем у частині узагальнення та формалізації моделей і методів побудови конвергентних систем ситуаційного управління та

оцінки їх ефективності з використанням формалізованих моделей знань предметної сфери їх застосування.

Практичне значення отриманих результатів полягає в тому, що застосування розроблених моделей і методів дозволяє підвищити ефективність побудови та функціонування конвергентних систем ситуаційного управління з урахуванням особливостей сфери їх застосування на основі моделей знань.

Теоретичні та практичні результати дисертаційної роботи впроваджені:

- у Інституті проблем математичних машин і систем НАН України (м. Київ) для підвищення ефективності проведення науково-технічних розробок при створенні ситуаційних центрів органів державної влади України та електронної парламентської системи «Рада-4» шляхом розробки та використання баз знань із предметних областей реалізації компонентів таких систем;

- у ДП «ЕС ЕНД ТІ Україна» (м. Київ) для підвищення ефективності створення систем управління операціями на основі використання моделей знань та оптимізації архітектурних рішень таких систем;

- у Science Research Center for Disaster Risk Reduction Department of Information Technology and Communications University of National and World Economy (м. Софія, Болгарія) для підвищення ефективності побудови архітектури та методів обробки ситуаційної інформації при створенні інтегрованих інформаційних систем оцінки ризиків природних катастроф;

- у Національному університеті біоресурсів і природокористування України (м. Київ) для аналізу ефективності технічних рішень при виконанні навчальних проектних завдань із реалізації комп'ютеризованих систем управління;

- у Кам'янець-Подільському національному університеті імені Івана Огієнка (м. Кам'янець-Подільський) для вдосконалення теоретико-методологічного забезпечення шляхом створення та використання моделей знань при побудові конвергентних комп'ютеризованих систем у процесі вивчення профільних дисциплін та при виконанні кваліфікаційних робіт студентів.

Впровадження підтверджуються відповідними документами.

Особистий внесок здобувача. Всі результати дисертаційної роботи, винесені на захист, отримано автором самостійно. Роботи [6] - [9], [11] - [14], [16] - [21], [25], [30] - [33], [35], [36], [39] - [41], [43] - [51], [53] - [56], [58] - [61], [63] - [65], [67] - [73], [75] - [83] написані самостійно. У роботах, опублікованих у співавторстві, особистий внесок здобувача полягає в наступному: [1] – проведено порівняльний аналіз архітектурних моделей комп'ютеризованих організаційних систем та запропоновано типову організаційну структуру ситуаційного центру; [2] – концепція та принципи побудови ситуативних архітектур систем ситуаційного управління; [3], [4] – ідея конвергентного поєднання спроможностей ситуаційних систем у рамках мережі розподілених ситуаційних центрів та математична постановка задачі оптимізації завантаження мережі; [5] – концепція оцінки рівня інтелекту систем ситуаційного управління на основі зменшення загальної ентропії системи; [10] – запропоновано модель прецедентів для опису процесу ситуаційного управління та квазіформальну модель структури процесу ситуаційного управління; [15] – загальна концепція та структура онтологічної моделі ситуаційного управління системою інформаційної безпеки; [22] – запропоновано композиційна модель процесу проектування проблемно-орієнтованих систем; [23] – запропоновано загальні принципи організації та архітектуру програмних засобів конфігураційного управління адаптивними інформаційними системами; [84] – запропоновано компонентну модель побудови навчальних систем на основі ситуаційних задач.

Апробація результатів дисертації. Основні положення і результати дисертаційної роботи доповідалися та обговорювалися на міжнародних конференціях: Third International Workshop on Computer Modeling and Intelligent Systems (CMIS–2020), м. Запоріжжя; 4th IFIP Conference on Information Technology in Disaster Risk Reduction (ITDRR–2019), м. Київ, Київський національний університет культури і мистецтв; IEEE First International Conference on System Analysis & Intelligent Computing (SAIC–2018), м. Київ, НТУУ «КПІ»; IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT–2018), м. Львів, Національний університет «Львівська політехніка»; 14th та 15th International Conferences on Advanced Trends in

Radioelectronics, Telecommunications and Computer Engineering (TCSET–2018 та TCSET–2020), м. Львів – смт. Славське, Національний університет «Львівська політехніка»; 2nd IFIP Conference on Information Technology in Disaster Risk Reduction (ITDRR–2017), UNWE, Sofia, Bulgaria; Сучасні проблеми інформатики в економіці, управлінні, освіті та подоланні наслідків Чорнобильської катастрофи: Міжнародні наукові семінари, ВНЗ «Національна академія управління», м. Київ – смт. Світязь, 2007–2019 рр.; Сучасні проблеми математичного моделювання, прогнозування та оптимізації: Міжнародні наукові конференції, м. Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2008, 2010, 2012, 2014, 2016, 2018, 2020 рр.; Системи підтримки прийняття рішень. Теорія і практика: дистанційні науково-практичні конференції з міжнародною участю, м. Київ, Інститут проблем математичних машин і систем НАН України, 2007–2013, 2015, 2017 рр.

Публікації. За результатами виконаних теоретичних і експериментальних досліджень опубліковано 84 наукові публікації, з них: розділ у колективній монографії, 15 статей у наукових виданнях, включених до Переліку наукових фахових видань України, затвердженого МОН України, 7 публікацій у наукових періодичних виданнях інших держав у напрямі, з якого підготовлено дисертацію, з них 4 статті у виданнях, проіндексованих у базі даних Scopus, у тому числі 1 стаття у виданні, віднесеному до третього квартиля (Q3), 14 статей підготовлено одноосібно; 5 статей у закордонних книжкових наукових виданнях із напрямку дисертації проіндексованих у базах даних Scopus та Web of Science; 46 публікацій апробаційного характеру у збірниках праць і тез наукових конференцій. Додатково наукові результати дисертації відображені у 3-х навчальних посібниках, затверджених вченими радами ВНЗ; 6 наукових статей у науковому періодичному виданні з економічного напрямку, з яких 5 статей у випусках, проіндексованих у базах даних Scopus та Web of Science, та 1 стаття у науковому періодичному виданні з медичного напрямку. 62 публікації підготовлено одноосібно. 24 публікації видано англійською мовою.

Структура та обсяг дисертації. Дисертація складається з анотації, вступу, шести розділів, висновків, списку використаних джерел та 4-х додатків. Загальний обсяг роботи становить 339 сторінок, із них основного тексту дисертації – 268 сторінок, 71 рисунок, 18 таблиць, список використаних джерел включає 312 найменувань та займає 34 сторінки, обсяг додатків складає 37 сторінок.

РОЗДІЛ 1

МЕТОДОЛОГІЧНІ ОСНОВИ СТВОРЕННЯ ТА ЗАСТОСУВАННЯ КОНВЕРГЕНТНИХ СИСТЕМ СИТУАЦІЙНОГО УПРАВЛІННЯ

1.1 Системні засади побудови систем ситуаційного управління

У широкому розумінні термін «система» означає сукупність технічних, природних чи соціальних елементів або їх комбінацію. Як і в інших спеціальних дисциплінах, у системній інженерії використовуються терміни, які можуть бути незрозумілими за межами предметної області. Наприклад, у теорії систем, а отже, і в системній інженерії, «відкрита» означає, що система здатна взаємодіяти зі своїм оточенням на відміну від «закритої» для оточення системи. Але в більш широкому інженерному середовищі «відкрити» розуміють як «неприватну», тобто «публічно доступну». У таких випадках у системі знань намагаються уникати неправильного тлумачення, використовуючи альтернативи: наприклад, «управління системою» або «управління проектуванням систем».

Будь-яка система розглядається в рамках її життєвого циклу, коли вона виконує своє призначення. Життєвий цикл системи – проміжок часу від моменту ініціації (концепції) створення системи до припинення її існування (виведення з експлуатації) через фази створення (розробки і побудови), функціонування (застосування) та удосконалення (супроводу).

Ефективність функціонування системи залежить від якості планування й обґрунтованості прийнятих у ній (для неї) організаційних та управлінських рішень.

Управління – це будь-який вплив на об'єкт управління для досягнення визначених цілей.

Ціль – бажаний стан об'єкта управління.

Планування – визначення цілей управління і засобів досягнення цих цілей.

Необхідність вирішення проблем побудови найскладніших технічних систем обумовила потребу у виробленні більш систематичних і цілісних підходів до їх вирішення. Для систематизації та узагальнення знань із розробки та використання

складних людино-машинних систем під егідою Міжнародної ради з системної інженерії (International Council on Systems Engineering (INCOSE)), Комп'ютерної спілки Інституту інженерів з електротехніки та електроніки (Institute of Electrical and Electronics Engineers Computer Society (IEEE-CS)) та Дослідного центру з системної інженерії (Systems Engineering Research Center (SERC)) було створено і підтримується проєкт «Ядро знань та навчальна програма з розвитку системної інженерії» (Body of Knowledge and Curriculum to Advance Systems Engineering (BKCASE)), одним із напрямів якого є узагальнення та підтримка в актуальному стані ядра (головної частини) знань із системної інженерії у вигляді керівництва до ядра знань із системної інженерії (Guide to the Systems Engineering Body of Knowledge (SEBoK)) [85], а також забезпечення його використання як безкоштовного і відкритого ресурсу з вільним доступом. Принципи та практики, надані у SEBoK, сприяють еволюції системної інженерії як дисципліни.

Ядро знань із системної інженерії SEBoK [85] позиціонує системну інженерію у ширшому розумінні, тобто система розглядається як фундаментальна частина. Для цього, не намагаючись перевизначити загальну системну термінологію, SEBoK запроваджує два пов'язані визначення, специфічні для системної інженерії:

- розроблювана система – це технічна або соціально-технічна система, яка є предметом життєвого циклу (ЖЦ) інженерії систем;
- розроблювана система, крім внутрішніх зв'язків між елементами, включає також зв'язки з інженерними, соціальними або природними системами в одному або кількох визначених середовищах.

Проблеми людини як елемента інженерної системи, зростання функціональності та ролі програмного забезпечення були усвідомлені в 1960-х роках [86], [87] і активно досліджувались у період з 1970-х [88] до 1990-х років [89], [90], [91], [92]. Результатом усвідомлення людського елемента в технічних системах була переорієнтація із традиційної системної інженерії на «програмні» підходи у системній інженерії. Для традиційної апаратно-орієнтованої системної інженерії характерними є послідовні процеси, заздалегідь задані вимоги, функціонально-ієрархічні архітектури, рішення на основі математичних обґрунтувань та розробка

системи за один етап. Підхід до системної інженерії на основі програмних систем зумовлений вимогами, які розвиваються і змінюються, одночасним визначенням вимог та рішень, поєднанням багаторівневої сервісно-орієнтованої та функціонально-ієрархічної архітектур, рішеннями на основі евристик та еволюційною розробкою системи. Зростання ролі програмного забезпечення як критичного елемента систем зумовило виникнення програмної інженерії (Software Engineering) як дисципліни, тісно пов'язаної з системною інженерією.

За визначенням [85], [93], системна інженерія є міждисциплінарним підходом, що надає засоби для реалізації успішних систем. Вона сфокусована на цілісному та одночасному розумінні потреб зацікавлених сторін, вивченні можливостей, документуванні вимог, а також синтезі, верифікації, валідації та еволюціонуванні рішень при розгляді проблеми в цілому, починаючи від розробки концепції системи до її передачі.

Успішні системи повинні задовольняти потреби своїх споживачів, користувачів та інших зацікавлених сторін. Інженерія систем включає такі ключові елементи (рис. 1.1) [85].

Принципи та поняття, що характеризують систему як комбінацію елементів, які взаємодіють між собою для досягнення визначених цілей. Система взаємодіє зі своїм середовищем, яке може включати інші системи, користувачів та природне середовище [94], [95]. Елементи, що складають систему, можуть включати апаратне та програмне забезпечення, мікрокод (прошивку), людей, інформацію, методи, засоби, служби та інші підтримуючі елементи.

Системний інженер – це людина або роль, яка підтримує міждисциплінарний підхід. Системний інженер виявляє і переводить потреби клієнта у специфікації, які можуть бути реалізовані командою розробників системи.

Системний інженер, допомагаючи в реалізації успішних систем, підтримує множину (набір) процесів ЖЦ, починаючи з ранніх етапів концептуального проектування і далі на протязі всього життєвого циклу системи через її створення, розгортання, використання та утилізацію. Він повинен аналізувати, специфікувати, розробляти та перевіряти систему, щоб забезпечити збалансованість її

функціональних, інтерфейсних, робочих, фізичних та інших показників якості й вартості для задоволення потреб зацікавлених сторін системи.

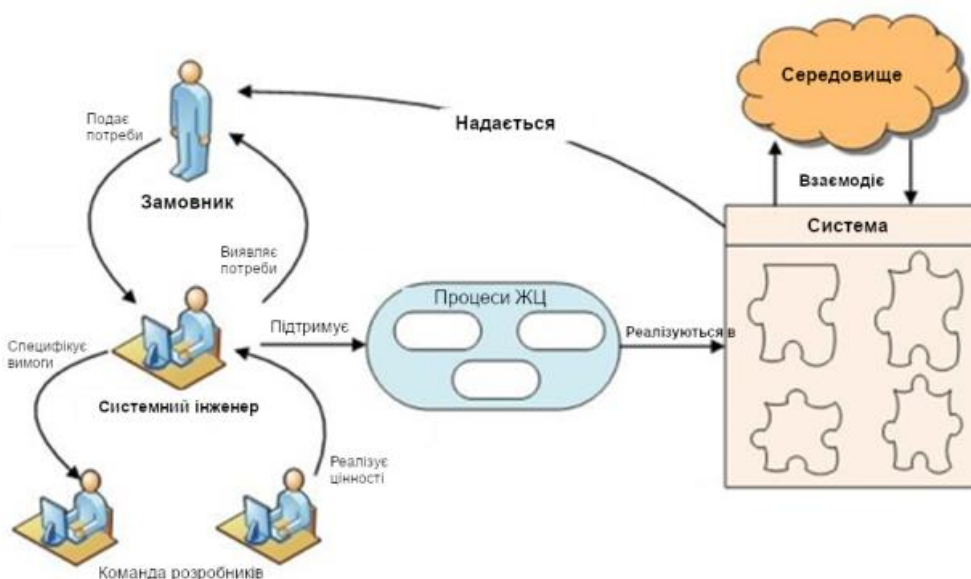


Рисунок 1.1 – Ключові елементи системної інженерії

Системний інженер допомагає забезпечити сумісність елементів системи для досягнення цілей і, зрештою, задовольнити потреби користувачів та інших зацікавлених сторін, які придбають і будуть використовувати систему.

Системна інженерія сфокусована на забезпеченні спільної роботи частин системи для досягнення загального призначення і базована на поєднанні практик, що застосовується у ряді суміжних галузей (зокрема, аерокосмічній та оборонній), які були використані як основа для стандартизованого підходу до управління ЖЦ будь-якої складної системи. Отже, практики системної інженерії на даний час значною мірою ґрунтуються на евристичних методах. Поточні дослідження спрямовані на розвиток теоретичної основи для системної інженерії з урахуванням основних знань із різних джерел. Системна інженерія розвивається у відповідності з постійним підвищенням складності систем. Значною мірою ця еволюція пов'язана з моделями та інструментами, орієнтованими на конкретні аспекти системної інженерії, такі як розуміння вимог зацікавлених сторін, подання системних архітектур або моделювання конкретних системних властивостей. Інтеграція між

різними дисциплінами, етапами розробки та проектами залишається ключовою проблемою системної інженерії.

Системна інженерія отримала визнання у різних галузях промисловості, науки та державного управління. Однак практика системної інженерії відрізняється залежно від галузей, організацій та типів систем. Взаємне збагачення практик системної інженерії повільно, але неухильно відбувається у різних галузях діяльності, однак глобальна потреба у спроможностях систем перевершила прогрес у системній інженерії.

Сфера застосування системної інженерії – це не все, що стосується інженерії та управління розробленою системою. Діяльність може бути частиною середовища системної інженерії, за винятком управління специфічною функцією, яка не вважається її частиною. Не зважаючи на те, що системна інженерія забезпечує реалізацію успішної системи, у випадку, коли певний вид діяльності не входить у сферу охоплення системної інженерії, (наприклад, погано кероване та погано організоване виробництво), системна інженерія не може забезпечити успішної реалізації.

Зручним способом визначення охоплюваної сфери діяльності системної інженерії є використання діаграми Венна. На рис. 1.2 [85] показано взаємозв'язок між системною інженерією, реалізацією системи та управлінням проектами (системами). Такі види діяльності, як аналіз альтернативних методів розробки, тестування та використання, є частиною функцій планування та аналізу системної інженерії. Такі види діяльності, як замовлення та монтаж виробничих ліній, а також їх використання у виробництві, хоча і є важливими для навколишнього середовища, зрештою залишаються поза межами системної інженерії. Слід зауважити, що інженерна реалізація системи включає також аспекти, пов'язані з виробництвом програмного забезпечення системи, тому програмна інженерія не розглядається як підмножина системної інженерії.



Рисунок 1.2 – Перетин областей знань інженерії систем, їх реалізації та управління проєктами (системами)

У документі INCOSE Systems Engineering Vision 2025 [96] зроблено висновок, що системна інженерія готова відігравати важливу роль у деяких глобальних викликах XXI-го століття і вже почала змінюватися, щоб відповідати цим викликам, але вона повинна пройти більш значущі трансформації, щоб повністю подолати ці виклики. Ключові пункти, взяті з підсумкового розділу [96], визначають атрибути трансформованої дисципліни системної інженерії в майбутньому.

Відповідність широкому колу областей застосування далеко за межами традиційного застосування в аерокосмічній та оборонній сферах, для задоволення зростаючих пошуків системних рішень, що забезпечують основні потреби у глобальному конкурентному середовищі.

Більш широке застосовування соціально-фізичних систем при проведенні оцінки для підтримки політичних рішень та різних форм відновлення.

Всебічна комплексна інтеграція вимог із боку різних ринкових, соціальних та екологічних зацікавлених сторін з урахуванням повного ЖЦ та довгострокових ризиків.

Ключова роль інтеграції для підтримки співпраці, що охоплює різноманітні організаційні та регіональні межі, а також широкий спектр дисциплін.

Підтримка більш ґрунтовним теоретичним базисом та досконаліми, оснований на моделях, методами та інструментами, що дозволяють краще зрозуміти найбільш складні системи й рішення в умовах невизначеності.

Покращені освітня інфраструктура, орієнтована на системне мислення, та системний аналіз на всіх етапах навчання.

Підготовка професіональних кадрів, які мають не тільки технічну освіту в предметній області, але й володіють передовими інструментами і методами, необхідними для вирішення поточних системних та інтеграційних завдань.

Деякі з цих напрямів системної інженерії розглядаються в SEBoK. Інші необхідно запровадити та повністю інтегрувати в області знань про системну інженерію у процесі їх розвитку. Ці області знань будуть використовуватися для надання огляду цих трансформуючих аспектів системної інженерії й будуть інтегровані в усі аспекти SEBoK у процесі удосконалення.

Діяльність щодо визначення системи забезпечує створення та детальний опис цільової системи (ЦС, system-of-interest, SoI) для задоволення виявленої потреби. Ця діяльність описується як основоположні процеси, що включають визначення системних вимог, системної архітектури, задуму системи та системного аналізу. Визначення архітектури системи може включати розробку моделей логічної та фізичної архітектури. Під час або в кінці будь-якої ітерації виконується аналіз пробілів для забезпечення відповідності всіх системних вимог архітектурі та задуму. Діяльність щодо визначення системи спирається на артефакти та рішення, отримані на етапі визначення концепції, в першу чергу, при чіткому формулюванні

призначення ЦС, потреб та вимог зацікавлених сторін і попередніх робочих концепцій.

Організація ЖЦ систем. Поняття ЖЦ систем почали використовувати при дослідженні соціально-економічних систем у 1950-х роках [95], [96]. Це поняття логічно, структуровано та методично описує процеси систем (продуктів) на кожному етапі їх існування. У роботі [97] зазначено, що ЖЦ розробки систем був запроваджений у 1960-х роках для створення великомасштабних функціональних бізнес-систем. Кількість етапів (фаз) ЖЦ системи та їх назви залежали від галузі застосування та призначення системи.

Термін «життєвий цикл» запозичений із природничих наук і використовується для опису змін, які зазнає окремий організм протягом життя, і взаємодії багатьох організмів із метою розвитку своєї популяції. В системній інженерії ЖЦ використовується для опису повного циклу існування екземпляра ЦС (system of interest, SoI), а також для опису керованої комбінації кількох таких екземплярів, які підтримують спроможності або забезпечують потреби зацікавлених сторін.

Модель ЖЦ визначає основні етапи, через які проходить специфічна ЦС, починаючи з моменту її започаткування до виводу з експлуатації. Моделі ЖЦ, як правило, реалізуються у проєктах розробки та строго узгоджуються із плануванням управління та прийняттям рішень.

Системна інженерія (рис. 1.3) [85] визначає загальні цілі зусиль із розробки системи, а саме: розуміння цінностей зацікавлених сторін; вибір конкретної потреби, яку необхідно вирішити; перетворення цієї потреби у систему (продукт або послугу), що забезпечує цю потребу; використання даного продукту або послуги для забезпечення цінностей зацікавлених сторін. Ця парадигма відповідає принципам системного підходу і використовується як базова для області знань SEBoK.

На рис. 1.3 *а* показано структурну декомпозицію системи на ЦС (SoI) у процесі її формування. ЦС1 розділено на основні елементи, які в даному випадку також є системами – ЦС2 і ЦС3, що складаються з системних елементів $E1, \dots, E4$. На рис. 1.3 *б* кожна ЦС має відповідну модель ЖЦ, яка складається з етапів, наповнених процесами. Функція цих процесів полягає у визначенні роботи, яка

повинна виконуватися, і пов'язаних з нею артефактів, які будуть створені. У модельно-орієнтованому підході ці артефакти фіксуються в моделі системи, яка представляє собою ЦС. Зауважимо, що деякі вимоги, визначені для задоволення потреб, розподіляються на ранніх етапах ЖЦ для ЦС1, тоді як інші позначаються для ЖЦ для ЦС2 або ЦС3. Декомпозиція системи свідчить про фундаментальне поняття рекурсії, як визначено у стандарті ISO/IEC/IEEE 15288:2015 [100], тобто положення стандарту будуть повторно застосовані для кожної ЦС. Важливо зазначити, що вимоги можуть бути розподілені по різних елементах системи, інтегрованих на різних етапах ЖЦ будь-якої із трьох ЦС, однак разом вони утворюють цілісну систему. Наприклад, ЦС1 може бути простою машиною, яка складається з шасі, двигуна та елементів керування, ЦС2 – вбудованою апаратною системою, а ЦС3 – основним програмним інтерфейсом та системою керування.

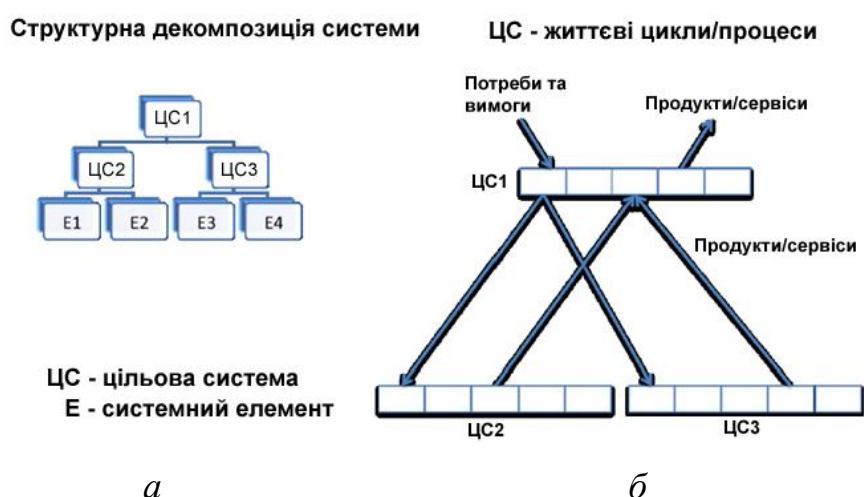


Рисунок 1.3 – Загальна парадигма системної інженерії: *а* – структурна декомпозиція системи; *б* – життєві цикли (процеси)

Під час виконання процесів ЖЦ системної інженерії потрібно послідовно проводити ітерації між етапами (наприклад, при уточненні визначення системи або при оновленні чи модернізації існуючої системи). Робота, яка виконується у процесах та етапах, може бути виконана паралельно в ЖЦ будь-якої з ЦС, а також у різних ЖЦ.

Цей підхід є фундаментальною основою для розуміння системної інженерії в цілому, а також для застосування системної інженерії до різних типів систем.

Узагальнена модель ЖЦ систем. Кожна ЦС (SoI) має пов'язану з нею модель ЖЦ. Системна інженерія зазвичай синхронізована з різними варіантами таких моделей ЖЦ, які могли б повністю задовольнити потреби зацікавлених сторін. Не існує єдиної моделі ЖЦ системи, яка підходить до всього і може забезпечити конкретні вказівки для всіх можливих ситуацій проєкту. Стандарт ISO/IEC 15288:2015 визначає узагальнену структуру процесів ЖЦ (рис. 1.4) [85], яка є початковою точкою для найпоширеніших версій заздалегідь визначених, послідовних, еволюційних, ситуативних та паралельних моделей ЖЦ.

Модель визначається як сукупність етапів, в рамках яких виконуються технічні та управлінські заходи. Ці етапи розділені точками прийняття рішень, де основні зацікавлені сторони вирішують, слід перейти до наступного етапу чи залишатись на поточному етапі, припинити чи перемасштабувати пов'язані з ними деталі проєктів.

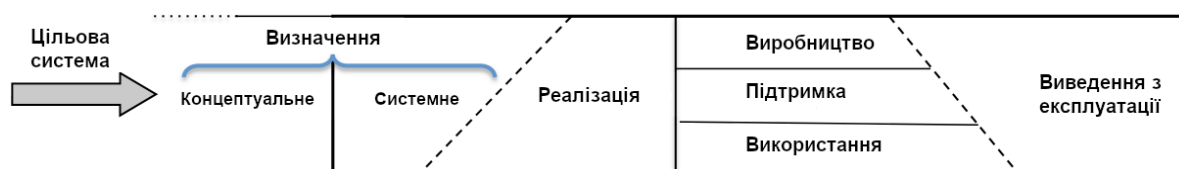


Рисунок 1.4 – Узагальнена структура процесів ЖЦ

Концептуальне визначення. Цей етап починається з рішення ініціатора (особи або організації) інвестувати ресурси в нову або вдосконалену інженерну систему. Ряд зацікавлених сторін погоджуються з необхідністю зміни контексту розроблюваної системи та вивчають, чи можлива розробка нових або модифікованих систем, переваги ЖЦ яких варті вкладання коштів у витрати. Діяльність включає розробку концепції оперування та множин сценаріїв (бізнес-кейсів); визначення ключових зацікавлених сторін та бажаних спроможностей; обговорення вимог між ключовими зацікавленими сторонами та вибір елементів системи, що не будуть розроблятися.

Системне визначення починається тоді, коли основні зацікавлені сторони вирішують, що потреби діяльності та вимоги зацікавлених сторін є достатньо чітко визначеними для обґрунтування необхідних варіантів рішення та вартості встановленого ЖЦ. Діяльність включає розробку системних архітектур, визначення та узгодження рівнів системних вимог, розробку планів ЖЦ системних рівнів та проведення системного аналізу, щоб оцінити сумісність та доцільність отриманого визначення системи. Перехід на стадію реалізації системи зумовлює одноразову або багаторазову розробку.

Слід зазначити, що дані визначення системи описують дії, які виконуються системними інженерами при вирішенні проблем системної інженерії. Існує дуже високий рівень паралелізму між формулюванням проблемної ситуації чи спроможності та описом одного або декількох можливих системних рішень, як це відбувається при системному підході. Інші пов'язані з визначеннями функції включають створення прототипів або фактичну розробку елементів високого ризику для демонстрування можливості реалізації системи; співпрацю з бізнес-аналітиками або проведення аналізу ефективності місії для забезпечення реалізації діяльності; модифікацію реалізованих систем для поліпшення їх виробництва, підтримки або використання. Ці заходи зазвичай відбуваються на основі ЖЦ для управління еволюцією системи, особливо при багатопотоковій розробці.

Реалізації системи. Цей етап починається тоді, коли основні зацікавлені сторони вирішують, що архітектура ЦС (SoI) і аргументи техніко-економічного обґрунтування мають достатньо низькі ризики. Це є обґрунтуванням необхідності визначених ресурсів для розробки та підтримки первинної операційної спроможності або розробки завершеної (повної) операційної спроможності. Діяльність включає побудову елементів розробки; інтеграцію цих елементів між собою та з нерозроблюваними елементами; верифікацію і валідацію елементів та їх інтеграцію, як це передбачено; підготовку до одночасної діяльності з виробництва, підтримки та утилізації.

Створення, підтримка та використання системи. Ці етапи починаються, коли основні зацікавлені сторони вирішують, що доцільність і безпека ЖЦ SOI

мають досить низькі рівні ризику, і це виправдовує використання ресурсів, необхідних для створення, підтримки та утилізації системи протягом очікуваного терміну служби. Тривалість виробництва, підтримки та використання, ймовірно, буде різною. Після завершення виробництва і підтримки, користувачі часто продовжують використовувати непідтримувані системи.

Створення системи включає виготовлення екземплярів або версій ЦС та потрібних для підтримки запасних частин, а також моніторинг і поліпшення якості продукції; діяльність із приймання продукції або послуг; постійне вдосконалення виробничого процесу. Крім того, створення системи може включати низькорейтингову початкову продукцію для удосконалення виробничого процесу або сприяння збереженню потенціалу виробництва і подальших збільшень попиту.

Підтримка системи включає різні класи технічного обслуговування: коригувальні (для дефектів), адаптивні (для взаємодії з взаємозалежними системами) і досконалі (для підвищення продуктивності, зручності використання). Вона також включає гарячі лінії та відповіді на запити користувачів або надзвичайну підтримку й надання необхідних витратних матеріалів (газу, води, електроенергії тощо). Підтримка системи має кілька сірих меж, а саме межа між невеликими удосконаленнями системи та розробкою більших нових доповнень, а також межа між переробкою (підтримкою) раніше проведених інкрементів в інкрементній або еволюційній розробці. Зазвичай підтримка продовжується після завершення створення системи.

Використання системи включає застосування ЦС операторами, адміністраторами, широким загалом або системами вищих рівнів в ієрархії ЦС. Зазвичай використання проводиться після припинення підтримки системи.

Виведення системи з експлуатації. Цей етап часто виконується покроково, оскільки версії системи або її елементів стають застарілими або нерентабельними для підтримки. Тому вони піддаються вилученню або переробці їх вмісту. Зростання доступних компенсацій робить переорієнтацію системи привабливою альтернативою.

Процеси системної інженерії (інженерії систем (SE)) застосовуються для забезпечення коректного прогресування від одного рівня розвитку до наступного деталізованого рівня з використанням контрольованих базових позицій. Ці процеси використовуються для системи, підсистем та системних компонентів, а також для підтримуючих або допоміжних систем, що використовуються при виробництві, експлуатації, навчанні, підтримці та утилізації ЦС. Виконання процесів та заходів технічного менеджменту, таких як дослідження ринку або управління ризиками, може виявити специфічні вимоги, інтерфейси або проєктні рішення, які не є оптимальними, і запропонувати зміни для підвищення загальної продуктивності системи, забезпечення економії коштів або дотримання термінів виконання планів. Процес ЖЦ систем являє собою спеціально виділений, функціонально злагоджений процес, який використовується для досягнення цілей і результатів стадій ЖЦ.

Процеси системної інженерії – це сукупність дій, яка цілеспрямовано використовується для вирішення задач системної інженерії, в якій використовується холістський підхід, тобто, будь-яка система подається як цілісний ансамбль, що поєднує збалансовані аспекти такої діяльності: організаційно-господарської, адміністративно-аналітичної, технічної (техноло-гічної).

Організаційно-господарська діяльність поділяється на зовнішню (придбання і постачання) і внутрішню (забезпечення діяльності). Основний зміст функціонування системи полягає у здійсненні процесів технічної діяльності (процесів розробки та використання системи) під управлінням процесів адміністративно-аналітичної діяльності (процесів управління життєвим циклом системи) для реалізації свого цільового призначення. Отже, найважливішими типами процесів системної інженерії є такі:

- технічні процеси систем;
- процеси управління ЖЦ систем (технічного управління).

Технічні процеси систем пов'язані зі створенням, впровадженням, використанням та припиненням діяльності систем і визначаються обраною для системи моделлю ЖЦ її розвитку (еволюціонування).

Технічні процеси системи забезпечують:

- перетворення визначених потреб і побажань зацікавлених осіб в опис збалансованої в рамках ЖЦ сукупності рішень щодо людей, продуктів і процесів;
- створення інформації для прийняття рішень;
- надання інформації для здійснення подальших технічних зусиль.

Процеси управління життєвим циклом систем здійснюють контроль над технічними процесами через планування, оцінювання, інформаційне забезпечення, прийняття рішень [101], [102], [103], [104]. Процеси системної інженерії повинні бути здатні до адаптації, під час якої слід визначати та документувати обставини, що впливають на адаптацію. Ці обставини включають (але не обмежують) таке:

- стабільність і різноманітність середовища функціонування;
- комерційні і (або) експлуатаційні ризики, що стосуються зацікавлених сторін;
- новизну, розміри і складність створюваної системи;
- дату початку та заплановану тривалість застосування;
- питання цілісності, такі як безпека, захист, секретність, зручність застосування, доступність;
- нові (що виникають) технологічні можливості;
- бюджетний профіль і доступні організаційні ресурси;
- готовність систем забезпечення до надання послуг.

Узагальнену модель відношень процесів розробки систем наведено на рис. 1.5.

Розрізняють такі етапи ЖЦ системи: дослідження, концептуалізація, розробка, реалізація, використання (підтримка), виведення з експлуатації.

У результаті успішного управління ЖЦ визначається:

- модель ЖЦ і сукупність процесів ЖЦ, які будуть використані організацією;
- політика і процедури застосування моделі і процесів ЖЦ;
- політика адаптації типових процесів ЖЦ до потреб окремих проєктів;
- показники, які дозволяють контролювати характеристики виконання процесів ЖЦ.



Рисунок 1.5 – Модель розробки системи

Системна інженерія на основі модельно-орієнтованого підходу. Системні інженери завжди використовували багато типів моделей, включаючи функціональні моделі для підтримки розвитку вимог, моделі для аналізу поведінки систем та інші аналітичні моделі для аналізу надійності, безпеки, масової властивості, споживання енергії і вартості. Проте системна інженерія суттєво залежить від артефактів на основі документів для фіксації та інформації про розробку, наприклад, вимог, документації по інтерфейсу керування та опису архітектури системи. Цю інформацію часто подано у великій кількості різних документів, включаючи текстові, неформальні рисунки та електронні таблиці. Такий документований підхід до системної інженерії має значні недоліки, а саме недостатню точність, невідповідність одного артефакту іншому, труднощі щодо збереження та повторного використання інформації.

Модельно-орієнтована системна інженерія ((MOCI), model-based systems engineering (MBSE)) – це формалізоване застосування моделювання для підтримки діяльності стосовно системних вимог, проектування, аналізу, верифікації та

валідації, починаючи з фази концептуального проєктування і продовжуючи у процесі розробки та на наступних етапах ЖЦ [105], [106], [107], [108]. Особливість підходу МОСІ полягає у тому, що модель є основним артефактом процесу проєктування системи. Основна увага при розробці, управлінні та налаштуванні моделі системи є перехід від традиційного підходу, базованого на документах, до системної інженерії, де акцент зроблено на документації про виробництво та налаштування системи. Використовуючи модель системи як основний артефакт, МОСІ дає потенціал для підвищення якості продукції, розширення повторного використання артефактів моделювання системи та покращення зв'язку між командами розробників. Це, у свою чергу, дозволяє зменшити час і витрати на інтеграцію та тестування системи, а також суттєво знизити вартість, ризики та спростити планування при створенні системи.

МОСІ включає різноманітний набір описових та аналітичних моделей, які можуть бути застосовані протягом усього ЖЦ, а також моделей системи систем (SoS) та моделей компонентів. Типові моделі можуть включати описові моделі архітектури системи, які використовуються для визначення та розробки системи, а також аналітичні моделі для аналізу продуктивності системи, фізичних та інших характеристик якості, таких як надійність, ремонтпридатність, безпека та вартість.

Термін MBSE (model-based systems engineering), тобто МОСІ, використано Уейном Ваймором у [109], де було наведено формальну систему на основі станів для аналізу систем стосовно їх вхідних (вихідних) характеристик та функцій цінності для оцінки корисності технологічно незалежних та технологічно залежних систем. Моделювання було широко використане у промисловості для забезпечення високоякісного аналізу ефективності складних систем. Стандарт визначення інтеграції для функціонального моделювання [110] було введено в 1990-х роках для підтримки базового функціонального моделювання. Модельний формалізм, який називається блок-схема розширеного функціонального потоку, використовувався для моделювання різних типів систем. Група об'єктного управління (OMG) представила концепцію модельно-керованої архітектури (МКА), Model Driven Architecture (MDA) [111], у якій використано стандартний підхід до моделювання.

Мову моделювання систем (OMG SysML) [112] було прийнято у 2006 р. як мову моделювання систем загального призначення. Крім того, у 2008 р. OMG прийняла уніфікований профіль для DoDAF та MODAF (UPDM) [113] для підтримки моделювання організацій.

Модельно-орієнтований підхід використовується при побудові різних типів складних людино-машинних систем [28], зокрема, для систем ситуаційного управління. Моделі для систем ситуаційного управління, крім технічних та організаційних підсистем, описують також сценарії, процедури узгодження, проєктну діяльність, шаблони тощо [114], [115], [116].

Системна інженерія та системна архітектура. Системна інженерія передбачає глибоку інтеграцію та узгоджену взаємодію (конвергенцію) технічної, соціально-економічної складових та середовища використання складної системи. Поєднання цих складових забезпечується побудовою та підтримкою архітектури складної системи. Системи або системна архітектура є концептуальною моделлю, яка визначає структурну, поведінкову та інші точки зору на систему. Опис архітектури – це формальний опис та подання системи, організовані так, щоб підтримувати обґрунтування рішень щодо структури та поведінки системи. Стандарт ISO/IEC 15288 визначає архітектуру системи як концептуальну її організацію, втілену в елементах, їх відношеннях між собою і з середовищем, а також визначає принципи, які є основою для прийняття рішень щодо її проєктування і еволюціонування.

Архітектура об'єкта, як визначено стандартом ISO/IEC/IEEE 42010 [117], є фундаментальною концепцією або властивостями об'єкта в його середовищі, втіленими в його елементах, відношеннях та принципах його проєктування та еволюції. Архітектура відображає:

- основні характеристики проблеми та області її вирішення з можливими альтернативами. Повне вирішення включає цільовий об'єкт і сутності, що його підтримують;
- координаційні дані для процесів, що підтримують життєвий цикл рішення, пов'язаного з архітектурою;

- інтереси зацікавлених сторін стосовно об'єкта розробки архітектури у формальних видах;

- припущення стосовно середовища кожного системного рішення для охоплення ЖЦ рішення – операційні процеси; природні, людські та технічні фактори, що взаємодіють із кожною системою; застосовані до них функціональні та нефункціональні обмеження, наприклад як у DLOD (United Kingdom Ministry of Defence Lines of Development), PESTEL (Political, Economic, Social, Technical, Environmental, Legal (Business Evaluation)), DOTPMLFI (Doctrine, Organization, Training, Materiel, Leadership & Education, Personnel, Facilities and Interoperability/Information) тощо.

Архітектура на рівні діяльності визначає, як група людей або організацій можуть співпрацювати в роботі із множиною архітектур зі спільним корпоративним баченням і як можна виконувати заходи щодо мотивації діяльності та способів управління проектами. Це забезпечує розуміння ландшафту архітектури разом з робочим середовищем, бібліотеками та сховищами в організації.

Архітектура на рівні спроможностей програми (проекту) охоплює бібліотеки, сховища, портфелі та дії, що використовуються у програмі або проекті спроможностей, який асоціюється з будь-якою архітектурою в середині організації.

Архітектурний підхід створює передумови та орієнтири для забезпечення життєздатності організацій і пов'язаних з ними заходів як в організації, так і в окремих проектах.

Складові архітектурного підходу. Різні види архітектур можна розглядати відповідно до їх цілей, областей застосування та ролей в ЖЦ організації та архітектури [118], [119], [120], [121], [122]. Наприклад, організація може визначити такі види архітектур:

- узагальнена (концептуальна) архітектура, що детально описує майбутню ситуацію (цей опис зазвичай охоплює кілька програм);

- архітектура домену (предметної області), яка буде використовуватися як еталонна за спроможностями (програмами) або для архітектури в межах домену (області) застосування;

– системна архітектура, що обмежується рамками окремої системи (проєкту) і стосується впровадження рішень.

Узагальнена (концептуальна) архітектура включає в себе:

- архітектурні принципи;
- спроможності, засоби, навички та компетенції (інструменти, дисципліни та спеціальності);
- зразки архітектури та побудови архітектур;
- активи – готові для поставки будівельні блоки;
- мотиваційні дані для побудови архітектури – політика та права, контракти, засоби узгодження, готовність та моделі зрілості.

Архітектура домену включає в себе:

- домен або предметну (стратегічну) область;
- мотиваційні дані;
- довідкові бібліотеки;
- репозиторії корпоративної архітектури;
- міграційний план для трансформації;
- портфелі активів;
- політику архітектури;
- архітектурну діяльність.

Системна архітектура включає в себе:

- масштаб системи (проєкту);
- мотиваційні дані;
- довідкові бібліотеки;
- репозиторії архітектури;
- міграційний план;
- портфелі активів;
- план управління архітектурою;
- проєктну архітектурну діяльність.

Управління спроможностями здійснюється в масштабі всієї організації, а також на рівні проєкту [123], [124], [125], [126]. Опис артефакту спрямовується

організаціями та проектами. Архітектурні описи зазвичай відображають набір пов'язаних проблем і розробляються для конкретних зацікавлених сторін.

Опис архітектури включає в себе один або декілька архітектурних видів (architecture views). Архітектурний вид (або просто вид) стосується одного або кількох інтересів зацікавлених сторін щодо цільової системи.

Види (проекції) – це ідеальний механізм для цілеспрямованої передачі інформації про певні проблеми. Вид — це точка зору (viewpoint), яка визначає поняття, моделі, методи аналізу та візуалізацію. Вид відображає архітектуру ЦС відповідно до архітектурної точки зору (Architecture viewpoint).

Точка зору охоплює одну або декілька проблем. Зацікавленість може бути сформована більш ніж однією точкою зору. Точка зору встановлює домовленості про визначення та оцінки видів для спрямування проблем, які охоплено цією точкою зору. Домовленості про точку зору можуть включати мови, нотації (позначки), типи моделі, правила розробки, методи моделювання та інші операції у видах.

Архітектура системи складається з компонентів системи та вже розроблених систем, які будуть функціонувати разом при реалізації загальної системи. Розроблено різні формальні мови для опису архітектури системи, які називаються мовами опису архітектури (ADL), зокрема, такі: Functional Flow Block Diagram (FFBD) [127]; Integration Definition for Functional Modeling (IDEF0) [110]; Object-Process Methodology (OPM) [128]; Systems Modeling Language (SysML) [112]; Unified Profile for United States Department of Defense Architecture Framework (DoDAF) and United Kingdom Ministry of Defense Architecture Framework (MODAF) [113]; OWL 2 Web Ontology Language (OWL 2) [129].

Життєвий цикл архітектури. Архітектура – це динамічна сутність, яка орієнтує процеси ЖЦ, описані у стандарті ISO/IEC/IEEE 15288, для системи з визначеною архітектурою. Архітектура має власний ЖЦ (побудова, використання та завершення використання, коли ця архітектура більше не застосовується або не задовольняє вимогам). Усвідомлення динамічної сутності архітектури обумовило початок розробки нових архітектурних стандартів для розширення контексту застосування архітектурного підходу у системній інженерії. Зокрема, разом з

роботою по оновленню стандарту опису архітектури систем ISO/IEC/IEEE 42010 [117] ведуться та плануються роботи по створенню нових стандартів ISO [130]: 42020 – архітектурні процеси (випуск очікується у 2018 році); 42030 – модель оцінки архітектури (випуск очікується у 2019 р.); 42040 – архітектурні методи (плануються); 42050 – архітектурні засоби (плануються).

Процеси (або види діяльності) підтримують цільовий об'єкт протягом його ЖЦ, тобто підтримують будь-яку діяльність, необхідну для того, щоб зробити цей об'єкт життєздатним протягом його ЖЦ. При безпосередній асоціації з системою ЖЦ архітектури відображає весь ЖЦ системи від концепції до його розпорядження. Архітектура забезпечує технічні домовленості власникам та розробникам систем на основі архітектурного плану, формування варіантів системи і підсистем цільового призначення та пов'язаних з ними допоміжних систем. Ці домовленості включають критичний шлях від початку розробки до реалізації численних інкрементів, керованих відповідними версіями планів інженерного управління системою. Іноді архітектура може відображати різні очікування, безпосередньо не пов'язані з окремою системою або її ЖЦ.

Приклади таких очікувань:

- Архітектура, створена для ідентифікації системи (систем), описує проблемний простір, щоб забезпечити вирішення проблем, що стосуються зацікавлених сторін. У такому випадку розробляються лише точки зору на діяльність (операції) та точки зору на спроможності. Вони використовуються для оновлення доктрини, робочих процесів або для придбання та управління системами або послугами. ЖЦ архітектури починається, коли починається аналіз проблеми, і завершується при найбільшому числі перетинів між просторами проблем та їх рішень.

- Архітектура, створена для охоплення кількох проєктів, що виконуються одночасно протягом певного періоду часу. Її можна назвати загальною, а набір проєктів розглядати як програму. ЖЦ такої архітектури починається з початком програми і закінчується останнім проєктом.

- Архітектура, створена для охоплення декількох систем (продуктів), які працюють одночасно протягом певного періоду часу. До цього варіанта належать виробничі лінії, сімейства систем та системи систем. Архітектура забезпечує загальне визначення, яке зазвичай вдосконалюється окремими архітектурами систем (продуктів).

- Архітектура, створена для покриття декількох проєктів, які виконуються послідовно, коли це можливо, протягом певного періоду часу. У цьому випадку архітектура надає план дій щодо трансформації, включаючи розробку та (або) заміну систем (продуктів), для виконання архітектурних завдань на визначеному проміжку часу.

Ці приклади підкреслюють необхідність налаштування архітектурних середовищ, дій та результатів на те, щоб вони були придатними для використання за призначенням. Таке налаштування залежить від архітектури організації та складності проблем, які в ній вирішуються, і може бути застосовано до різних планів та діяльності на архітектурному об'єкті.

Розробка сучасних складних людино-машинних (кіберлюдських, кіберсоціальних систем) систем різного призначення вимагає індустріального підходу для забезпечення контрольованості процесів проєктування, випуску, застосування та виведення з експлуатації таких систем. Проведений аналіз підходів до побудови і управління складними організаційно-технічними системами [1], [13] показав, що таким методологічним базисом при цьому є архітектурний підхід та використання концепції ЖЦ при розгляді складних систем.

Одним з сучасних методів моделювання є **ARCADIA (ARChitecture Analysis & Design Integrated Approach)**, заснований на архітектурному підході та керованій моделями діяльності із проєктування систем [131].

1.2 Інженерія системи систем

Інженерія системи систем виникла як підхід для розв'язання задачі побудови складних систем, до складу яких входять інші системи, зберігаючи при цьому свою ідентичність. Існує кілька визначень системи систем, деякі з яких залежать від особливостей області застосування. Зокрема, М. Майєр [132] постулював п'ять основних характеристик системи систем: операційна незалежність складових систем, управлінська незалежність складових систем, географічний розподіл, емерджентна (виникла) поведінка та процеси еволюційного розвитку. Визначив операційну незалежність та незалежність управління як дві основні відмінні характеристики для застосування терміна «система систем» (СС, SoS). Система, яка не має цих двох характеристик, не вважається системою систем незалежно від складності чи географічного розподілу її компонентів.

У характеристиці Майєра емерджентність відзначається як загальна характеристика СС, особливо в СС, що складається з декількох великих існуючих систем, заснованих на вимогах (часових та ресурсних) підпорядкування всіх можливих логічних потоків у множині функцій, спроможностей та даних системи в СС. Емерджентність може бути пов'язана з ризиками несподіваної або ненавмисної поведінки, що виникають у результаті поєднання систем, які мають власну індивідуально притаманну складну поведінку. Ці ризики можуть стати суттєвими у випадках, коли безпеці системи загрожує, наприклад, ненавмисна взаємодія між функціями, що надаються декількома складовими системами в СС. ISO/IEC/IEEE 21839 [133] дає таке визначення СС та її конститuentної (складової) системи.

Система систем – сукупність систем або системних елементів, які взаємодіють між собою, щоб забезпечити таку унікальну спроможність, яку жодна з її конститuentних (складових) систем не може досягти самотійно. Примітка: елементи системи можуть бути необхідними для полегшення взаємодії складових систем у системі систем.

Конституентні системи можуть бути частиною однієї або декількох СС. Примітка. Кожна складова є корисною системою сама по собі, має власний розвиток, цілі управління та ресурси, але взаємодіє в межах СС для забезпечення унікальних можливостей СС.

Слід зазначити, що формування СС – це не обов'язкове постійне явище, а скоріше питання необхідності узгодженої інтеграції та поєднання у мережу систем для конкретних цілей, таких як надійність, вартість, ефективність, функціональність тощо.

Враховуючи незалежність складових систем, процеси інженерії системи в більшості випадків реалізуються також і для інженерії системи систем. Отже, їх необхідно адаптувати для підтримки характеристик СС. Процеси, що реалізуються як на системному, так і на рівні СС, представлені в табл. 1.1 При цьому інженерія системи систем часто обмежується лише системами.

Таблиця 1.1 – Особливості реалізації процесів інженерії систем в інженерії СС

Процес інженерії систем	Реалізація стосовно інженерії системи систем
Процеси узгодження	Оскільки часто відсутні повноваження найвищого рівня для СС, ефективні угоди між системами у складі СС є ключовими для успішної інженерії СС
Організаційні процеси забезпечення проєкту	У рамках інженерії СС розробляються та підтримуються ті процеси, які є критичними для СС у межах обмежень процесів на системному рівні

Продовження таблиці 1.1

Процеси технічного управління	У рамках інженерії СС впроваджуються процеси технічного управління, застосовані для конкретних положень інженерії СС – планування, аналізу, організації та інтеграції спроможностей суміші існуючих і нових систем у спроможності системи систем у той час, як системи продовжують відповідати за технічне управління своїх систем
Технічні процеси	Технічні процеси інженерії СС визначають наскрізну спроможність СС, виходячи з аналізу діяльності/місії на рівні СС, а також визначення потреб та вимог зацікавлених сторін. Архітектура та дизайн СС поєднують планування, організацію та інтеграцію складових систем, обмежених системними архітектурами. Розробка, інтеграція, перевірка, перехід та валідація реалізуються системами за допомогою моніторингу та огляду інженерії СС. Інтеграція, верифікація, передача та перевірка в інженерії СС застосовуються, коли складові системи інтегруються в СС та перевірена і підтверджена її продуктивність

Стандарт ISO/IEC/IEEE 21841 [134] представляє таксономію СС, що базується на ступені незалежності складових і пропонує основу для розуміння СС на основі походження цілей СС та взаємозв'язків між зацікавленими сторонами як для СС, так і для її складових систем. За цією таксономією СС може бути віднесена до одного з чотирьох типів:

1. Підтверджені (визнані) СС (acknowledged SoS) – СС із підтвердженими цілями, призначеним керівником та виділеними для СС ресурсами.

Примітка. Складові системи зберігають свою незалежність, цілі, фінансування та підходи до розробки та супроводу. Зміни в системах ґрунтуються на угодах про співпрацю між СС та системою.

2. Співпрацюючі (колаборативні) СС (collaborative SoS) – СС, в яких компонентні системи взаємодіють більш-менш добровільно для виконання узгоджених найважливіших цілей.

Примітка. Складові системи спільно вирішують, як надавати або відмовляти у наданні сервісів, забезпечуючи тим самим засоби підкріплення та підтримання узгодженості.

3. Керовані СС (directed SoS) – СС, створені і керовані для виконання конкретного призначення, а складові системи підпорядковуються СС.

Примітка. Компонентні системи підтримують можливість самостійної роботи; однак їх нормальний режим роботи підпорядковується головній керованій меті.

4. Віртуальні СС (virtual SoS) – СС, в яких обмежені повноваження центрального управління та централізованої узгодженої мети для СС.

Примітка 1. Виникає поведінка великого масштабу, і це може бути бажано, але цей тип SoS покладається на відносно невидимі механізми його підтримки.

Примітка 2. Віртуальні СС, як правило, є самоорганізованими.

У більшості фактичних випадків СС відображатиме комбінацію типів СС, яка може змінюватися з часом. Інші таксономії можуть зосереджуватися на природі/типі компонентів, їх неоднорідності тощо [135].

Відмінності між інженерією систем та інженерією системи систем. Особливості реалізації процесів інженерії систем в інженерії системи систем (табл. 1.1) є орієнтовними, і в кожному конкретному випадку ступінь різниці залежить від практики. Реальні системи, як правило, дуже взаємопов'язані і лише умовно можуть бути віднесені до однієї і двох категорій у залежності від конкретного практичного випадку. Такі умовні відмінності застосування принципів інженерії систем в інженерії системи систем наведені в табл. 1.2 [85].

Таблиця 1.2 – Відмінності між основними рисами систем та системи систем

Риси систем	Риси системи систем
Чітко визначений перелік зацікавлених сторін	Декілька рівнів зацікавлених сторін зі змішаними і, можливо, конкуруючими інтересами
Чітко визначені цілі та призначення	Декілька і, можливо, суперечливих цілей та призначень
Чітко визначені оперативні пріоритети з розширеннями для визначення пріоритетів	Декілька та іноді різних операційних пріоритетів без чітких шляхів розширення
Єдиний життєвий цикл	Життєвий цикл включає декілька життєвих циклів конститuentних систем із елементами, реалізованими асинхронно
Чітко визначене володіння з можливістю переміщення ресурсів між елементами	Декілька власників приймають незалежні рішення щодо розподілу ресурсів

Ключовими принципами побудови СС є інтероперабельність та координація між її конститuentними системами. Інтероперабельність загалом визначається як здатність двох або більшої кількості систем, продуктів чи компонентів обмінюватися інформацією та використовувати інформацію, отриману в результаті обміну. Інтероперабельність у контексті СС означає, що кожна система може спілкуватися та взаємодіяти з будь-якою іншою системою незалежно від їх апаратних та програмних характеристик або природи. Це означає, що кожен конститuentний елемент (та потенційні нові елементи) СС повинен мати можливість спілкуватися з іншими без проблем сумісності в операційних системах, апаратних засобах, зв'язку тощо. Для цього СС потребує спільної мови, якою можуть користуватися системи СС. Прикладом такої мови є XML (eXtensible Markup Language).

Однією із ключових проблем побудови архітектури СС є те, що складові системи СС можуть розроблятися без урахування їх ролі СС, та обмежень параметрів архітектури СС. Архітектурний рівень, який накладає ці обмеження на складові системи і підтримує результуючі можливості СС, може базуватися на відкритих стандартах. Відкритість архітектури є основою для подальшого еволюціонування СС.

Найважливішим завданням переходу від концепції СС до інженерії СС є суттєві технологічні, людські та організаційні відмінності в системі розгляду системних підходів та управління [136]. Потенційним підходом до розробки СС може бути підхід на основі принципів, відкритих системам [137]:

- принцип відкритого інтерфейсу (open interface) – відкриті системи мають проникні межі, що дозволяють їм обмінюватися масою, енергією та інформацією з іншими системами;

- принцип синергізму (synergism) – поняття, яке означає, що кооперативна взаємодія між складовими системами має більший ефект за рахунок спільних зусиль, ніж сумарний ефект дій окремих частин; по суті, це те, що породжує емерджентність;

- принцип самокерованості (self-government) означає, що СС підтримує і розвиває свій внутрішній порядок без втручання зовнішніх джерел; він може бути реалізований на основі кібернетичного управління, гомеостазу чи самоорганізації;

- принцип емерджентності (emergence) – у цьому випадку мається на увазі виникнення нових та зв'язних структур, закономірностей та властивостей під час самоорганізації СС;

- принцип збереження (conservation) – встановлює, що енергія та маса (матеріал) зберігаються в межах СС;

- принцип реконфігурації (reconfiguration) – стосується переналаштування та адаптації СС, щоб підтримувати себе в умовах змін у своєму оточенні;

- принцип симбіозу (symbiosis) – системи в середині СС мають симбіотичний зв'язок одна з одною; або простіше, успішна розробка та підтримка СС залежить від

симбіотичної співпраці між зацікавленими сторонами систем, з яких вона складається;

– принцип модульності (modularity) – означає, що кожен рівень і кожна система певною мірою незалежні від інших. При створенні СС розробка незалежних модульних систем, які взаємодіють між собою за допомогою стандартизованих інтерфейсів, дає змогу підвищити гнучкість, що сприяє кращому еволюціонуванню СС.

1.3 Ситуаційне управління як продукт трансформації інформації засобами інформаційних технологій

Забезпечення сталого розвитку суспільства безпосередньо пов'язане зі створенням умов, засобів і технологій, що дозволяють враховувати природні, політичні, соціальні і техногенні впливи та мінімізувати (попереджувати, запобігати та усувати) їх шкідливі і катастрофічні наслідки, і здійснюється через цілеспрямовану діяльність, направлену на вирішення цих проблем.

Цілеспрямована діяльність описується траєкторією, що включає події, ситуації, мотивацію, цілі, ресурси, обмеження. Сукупність можливих траєкторій цілеспрямованої системи описується графом, вершини якого представляють стани системи, а дуги – події, що викликають перехід системи з одного стану в інший. Оскільки граф описує послідовний у часі зміну станів, він є ациклічним і орієнтованим. Поєднання понять «подія-умови-стан-ставлення-ситуація» визначають ситуацію як контекст (ставлення, семантику) стану, що визначається подіями за певних умов.

Одним із шляхів підтримки цілеспрямованої діяльності для вирішення проблеми сталого розвитку суспільства є створення та використання систем ситуаційного управління для різних сфер діяльності. Високі результати при створенні засобів і технологій ситуаційного управління дає конвергенція різних

предметних областей (доменів), що забезпечує синергетичний ефект при їх застосуванні.

Ситуаційними будемо називати системи, зміст стадій життєвого циклу яких визначається ситуаційними факторами [6], [7]. Формування та функціонування ситуаційних систем ґрунтуються на понятті ситуації та цільовому призначенні таких систем. Таксономія ситуаційних систем може бути побудована на основі ознак походження, застосування, реалізації тощо.

Ситуація розглядається як усвідомлене знання суб'єкта про динаміку навколишнього середовища, представленого певними типами інформаційних повідомлень, що є основою для побудови обґрунтованої інтерпретації послідовності зміни станів (динаміки) світу (предметної області) з певної точки зору. Семантика (зміст) ситуації визначається її модальністю, яка виражає зв'язок між ситуацією та реальністю (об'єктивною модальністю), суб'єктом і ситуацією (суб'єктивною модальністю).

Ситуаційні фактори (також відомі як зовнішні чинники) – це впливи, які виникають ззовні системи – середовища існування (функціонування) та інших оточуючих систем. Диспозиційні фактори визначають внутрішні (індивідуальні) характеристики системи, які впливають на поведінку та функціонування системи, такі як функціональні характеристики, показники якості та архітектура. Диспозиційні фактори визначають успішність функціонування (застосування) систем («Система виконала свої завдання завдяки достатньому рівню спроможностей», «Використання системи відповідає її призначенню»). Ситуаційні чинники впливають на неуспішність функціонування системи («Поставлені задачі не відповідають спроможностям», «Використання системи не відповідає призначенню»).

Ситуаційне управління – це метод управління, заснований на використанні сімейства концепцій, моделей та наявних технологій для розпізнавання, пояснення, впливу та прогнозування ситуацій, які виникли або можуть виникнути в динамічній комплексній системі за визначений час роботи. Ситуаційне управління являє собою цілеспрямовану індивідуальну або колективну діяльність, пов'язану з

розпізнаванням, поясненням і прогнозуванням ситуацій та впливом на них із використанням відповідних концепцій, моделей і технологій. Така діяльність здійснюється у системах ситуаційного управління – організаційно-технічних інформаційних комплексах підтримки управлінських рішень на основі комплексного моніторингу факторів впливу на процеси, що відбуваються в середовищі керування, з використанням сучасних інформаційних технологій [138].

Ситуаційне управління здійснюється на основі когнітивного моделювання як методології прийняття рішень у ситуаціях невизначеності або неповної визначеності. Методологія когнітивного моделювання орієнтована на формування управлінських рішень із обґрунтованою ефективністю, знаходження сценаріїв розвитку подій на основі виділення понять (концептів, чинників), що кількісно та якісно характеризують ситуацію, а також оцінки взаємного впливу цих чинників.

Таксономія ситуаційних систем може бути побудована на основі ознак походження, застосування, реалізації тощо. Приклад такої таксономії ситуаційних систем представлено на рис. 1.6 [7].



Рисунок 1.6 – Таксономія ситуаційних систем

Життєвий цикл ситуаційного управління як цілеспрямованої діяльності реалізується через послідовність таких процесів: усвідомлення ситуації → оцінка ситуації → визначення цілей СУ → розробка та відбір плану для досягнення цілей СУ → організація процедур прийняття рішень → виконання плану СУ → моніторинг виконання плану СУ → оцінка та збереження результатів виконання СУ.

Прийняття рішень при реалізації ситуаційного управління можна описати моделями, які базуються на проєктному підході та процесах його реалізації. Важливим елементом цього підходу є управління життєвим циклом проєкту.

Ситуаційне управління є методом управління (у складних, розподілених, динамічних системах), що ґрунтується на понятті ситуації, класифікації ситуацій та їх перетвореннях [139]. Процес ситуаційного управління являє собою цілеспрямовану індивідуальну або колективну діяльність на основі використання множини концепцій, моделей, доступних технологій, пов'язану з розпізнаванням, поясненням і прогнозуванням ситуацій, які виникли або можуть виникнути в динамічних системах протягом визначеного часу роботи [140], [141], та впливом на них з використанням відповідних наборів концепцій, моделей і технологій.

Теорія СУ орієнтована на вирішення проблем, які не можуть бути вирішені в рамках традиційної теорії автоматичного управління для об'єктів управління, що характеризуються переліком специфічних особливостей, а саме [139]:

- унікальністю;
- відсутністю формалізованої мети існування;
- відсутністю формальної моделі оптимізації;
- динамічністю (еволюційністю);
- неповнотою опису;
- наявністю свободи волі.

Оскільки СУ ґрунтується на класифікації ситуацій та їх перетвореннях, то необхідно визначити зміст поняття «ситуація» та критерії класифікації ситуацій відповідно до типів їх перетворень.

У словнику Merriam-Webster слово «ситуація» визначається як [142] всі факти, умови і події, які впливають на когось або щось у певний час і в певному місці. За визначенням Йона Барвіуса та Джона Перрі, ситуації є чітко усвідомленими (хоча й не точно індивідуалізованими) в загальному і лінгвістичному розумінні «частинами світу», що складаються з пов'язаних між собою об'єктів, які мають певні властивості. Події та епізоди є ситуаціями в часі, сцени є візуально розрізненими ситуаціями, зміни є послідовностями ситуацій, а факти є ситуаціями, збагаченими

(або спотвореними) засобами мови [143], [144]. Важливим у цьому визначенні є словосполучення «частини світу». Тобто, ситуації є лише частинами світу і інформація, яку суб'єкт (агент) має про дану ситуацію в будь-який момент часу, буде лише частиною всієї інформації, яка може бути теоретично доступна. Акцент на частковості відділяє ситуаційну семантику від того, що багато хто вважав її головним конкурентом теорії семантики, семантики можливих світів.

Сам термін «ситуація» визначається обсягом, сферою та контекстом його використання. Семантика (зміст) ситуації визначається його модальністю, яка виражає зв'язок між ситуацією та реальністю (об'єктивною модальністю), суб'єктом і ситуацією (суб'єктивною модальністю). Таким чином, ситуація повинна розглядатися як усвідомлене знання суб'єкта про динаміку навколишнього середовища, представленого певними типами інформаційних повідомлень, що є основою для побудови обґрунтованої інтерпретації послідовності зміни станів (динаміки) світу (предметної області) з певної точки зору [25], [40].

Ситуації за видами їх інтерпретації та перетворень можуть бути класифіковані за різними ознаками [145], зокрема, за масштабом охоплення, оцінкою стану об'єкта управління, ступенем структурування, причинами виникнення, складністю, тривалістю, змістом (предметною сферою). Також ситуації можуть бути класифіковані за притаманною змінюваністю як динамічні та статичні [146].

Діяльність щодо здійснення ситуаційного управління проводиться в рамках систем ситуаційного управління (ССУ). Один із сучасних підходів до ситуаційного (стратегічного, оперативного, превентивного, кризового, аварійного) управління погано структурованими процесами у складних системах заснований на використанні експертних методів прийняття рішень. Технологічне та інформаційне забезпечення фахівців здійснюється з використанням інструментів різного призначення, інтегрованих у ССУ. ССУ призначена для колективного прийняття рішень стосовно різних завдань (проблемних ситуацій) управління з використанням автоматизованих інструментів і всебічного аналізу багатоцільових сценаріїв та прогнозування ситуацій у різних предметних областях.

Система ситуаційного управління (ССУ) є важливою складовою забезпечення адекватного оперативного управління складними системами різного масштабу і призначення з використанням ситуаційних центрів (СЦ). СЦ являє собою організаційно-технічний комплекс підтримки процесів ситуаційного управління, основу якого складають сучасні інформаційні технології підтримки прийняття, впровадження та контролю виконання управлінських рішень шляхом комплексного моніторингу розвитку процесів, факторів впливу на них і аналізу ситуацій, що відбуваються в середовищі управління [147]. Основними складовими СЦ є підсистема моніторингу стану об'єкта управління, підсистема аналізу ситуацій та формування керуючих впливів на об'єкт управління, підсистема передачі керуючих впливів на об'єкт управління.

Діяльність (функціонування) в ССУ – це робочий процес трансформації (переробки) інформації з метою вироблення, доведення до виконавців та контролю виконання управлінських рішень, заснований на сучасних інформаційних технологіях, які забезпечують використання формальної і неформальної баз знань для вибору керуючих дій на об'єкти предметної області протягом визначеного періоду часу для створення необхідних умов досягнення цільового стану проблемної області [147], [148].

Загальний вигляд схеми прийняття рішень в ССУ представлений на рис. 1.7.

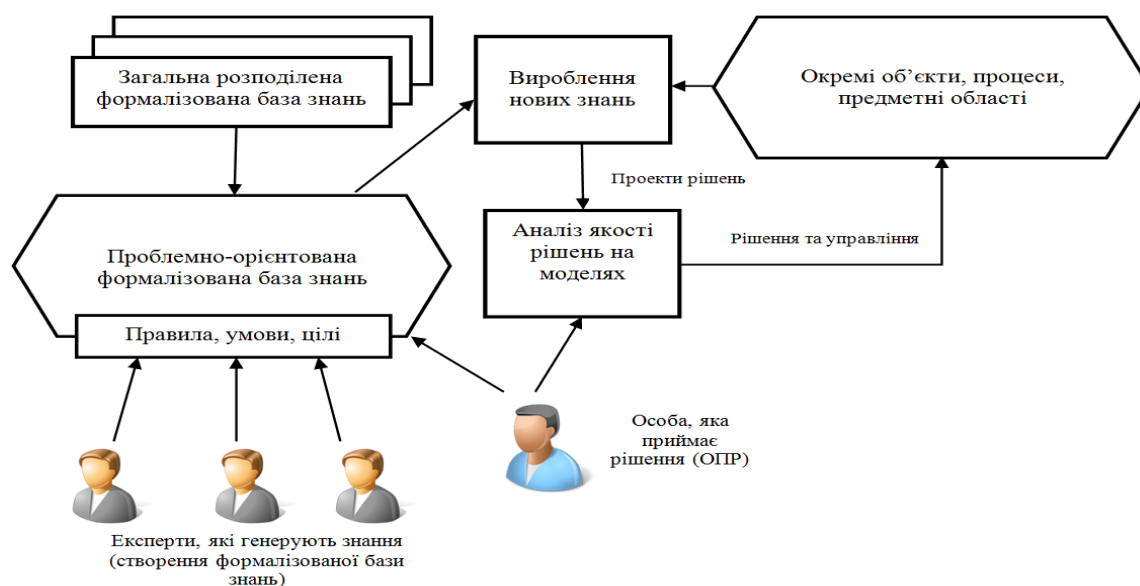


Рисунок 1.7 – Схема прийняття рішень у ССУ

1.4 Моделі ситуаційного управління

Індивідуальні моделі ситуаційного управління. Ситуаційне управління відноситься до категорії когнітивних (пізнавальних) та управлінських процесів (cognitive and governing processes) на різних рівнях у системах різноманітного призначення і є циклічним процесом. Ситуаційне управління здійснюється у рамках систем ситуаційного управління (ССУ). В переважній більшості випадків у процесі ситуаційного управління в ССУ можна виділити такі основні фази:

- оцінки ситуації, результатом якої має бути усвідомлення ситуації;
- осмислення ситуації, результатом якої має бути розуміння ситуації;
- аналіз ситуації, результатом якої має бути прогнозування розвитку ситуації;
- прийняття рішення (рішень) стосовно реакції на ситуацію в цільовому середовищі;
- впровадження прийнятих рішень у цільовому середовищі;
- контроль виконання впроваджених рішень з переходом до аналізу нової ситуації.

Ситуаційне управління здійснюється на основі когнітивного моделювання як методології прийняття рішень у ситуаціях невизначеності або неповної визначеності. Методологія когнітивного моделювання, запропонована Р. Аксельродом [149], орієнтована на формування найбільш ефективних управлінських рішень, пошук сценаріїв розвитку подій на основі виділення понять (концептів, чинників), що кількісно та якісно характеризують ситуацію, а також оцінки взаємного впливу цих чинників. У рамках ССУ когнітивне моделювання здійснюється колективами людей (ОПР, експертів, аналітиків), ґрунтується на моделюванні їх суб'єктивних думок про ситуацію і включає:

- методологію структуризації ситуації: модель подання знань експерта у вигляді знакового оргграфа (когнітивної карти) (F, W) , де F – множина чинників ситуації, W – множина причинно-наслідкових зв'язків між чинниками ситуації;
- методи аналізу ситуації.

Ситуаційне управління здійснюється в рамках перцептивного (сприйняття та усвідомлення) циклу ситуації в оточуючому середовищі вперше описаного в роботі Ульріка Нейссера (U. Neisser) [150], [151]. Перцептивний цикл, запропонований Нейссером (рис. 1.8) [150], в одній точці містить активну схему (схему фенотипу як активну конструкцію). Схема одночасно є і моделлю фактичної ситуації, і моделлю очікувань того, яка інформація може бути підібрана відповідно до ситуації. Ця схема спрямовує перцептивне дослідження, випробовуючи середовище, модифікуючи схему, яка завершує цикл. У цій точці циклу далі використовується поняття <когнітивних карт>/ <орієнтувальних схем> для представлення знань про те, де знайти інформацію, яка в даний час поза спостереженням. Вона спрямовує просування, щоб перетворити потенційно доступну інформацію на наявну інформацію. Керівництво може відбуватися більш-менш звично, без чітких рішень “куди піти далі” на кожному кроці. Схема представляє сфокусовану (послідовну) увагу; зразки (паттерни) сканування проявляються у відповідь на ситуації. Сигнали, однак, можуть також привернути увагу (паралельний процес неусвідомленої уваги).



Рисунок 1.8 – Модель перцептивного циклу, запропонована У. Нейссером

Екологічний підхід Сміта і Хенкока має більш цілісний підхід, розглядаючи СУ як «генеративний процес створення знань та здійснення поінформованої дії» ([152], с. 138). Їх опис ґрунтується на моделі перцептивного циклу Нейсера [150],

яка описує взаємодію людини зі світом і впливову роль схем у цих взаємодіях. Згідно з моделлю перцептивного циклу, взаємодія зі світом (дослідження) спрямовується внутрішніми схемами. Результат взаємодії змінює оригінальні схеми, що, у свою чергу, спрямовує подальші дослідження. Цей процес спрямованої взаємодії і модифікації має нескінченний циклічний характер. Використовуючи дану модель, Сміт і Хенкок [152] припускають, що СУ не властиве ні світові, ні людині, але проявляється через взаємодію людини зі світом. Сміт і Хенкок ([152], с. 138) описують СУ як «зовнішньо спрямовану свідомість», тобто «інваріантну складову в адаптивному циклі знань, дій і інформації». Сміт і Хенкок [152] стверджують, що процес виконання і підтримка СУ зосереджуються навколо внутрішньо підтримуваної схеми, яка містить інформацію про певні ситуації. Ці схеми полегшують передбачення ситуаційних подій, спрямовуючи увагу особистості на сигнали в навколишньому середовищі і направляючи кінцевий хід дій суб'єкта. Потім людина проводить перевірки для підтвердження того, що розвиток ситуації відповідає очікуванням. Будь-які несподівані події слугують подальшому пошуку та поясненню, що, у свою чергу, змінює існуючу модель оператора. Модель перцептивного циклу СУ представлена на рис. 1.9.

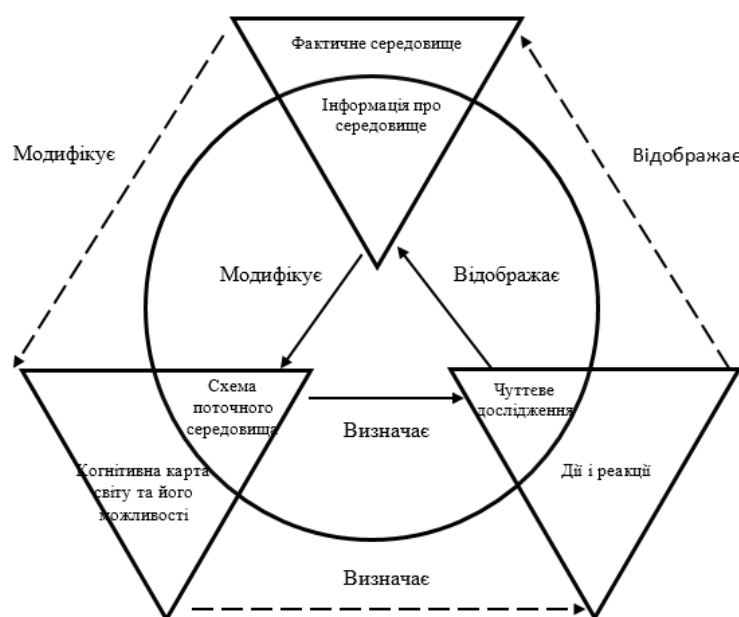


Рисунок 1.9 – Екологічна модель перцептивного циклу

Трирівнева модель СУ Міки Ендслі (M. Endsley) [153] є найбільш відомою (рис. 1.10). Ця модель описує СУ як внутрішньо притаманний когнітивний (пізнавальний) продукт обробки інформації, що базується на трьох ієрархічних рівнях розпізнавання ситуації, відокремлених від процесів оцінки ситуації, які використовуються для досягнення мети СУ.

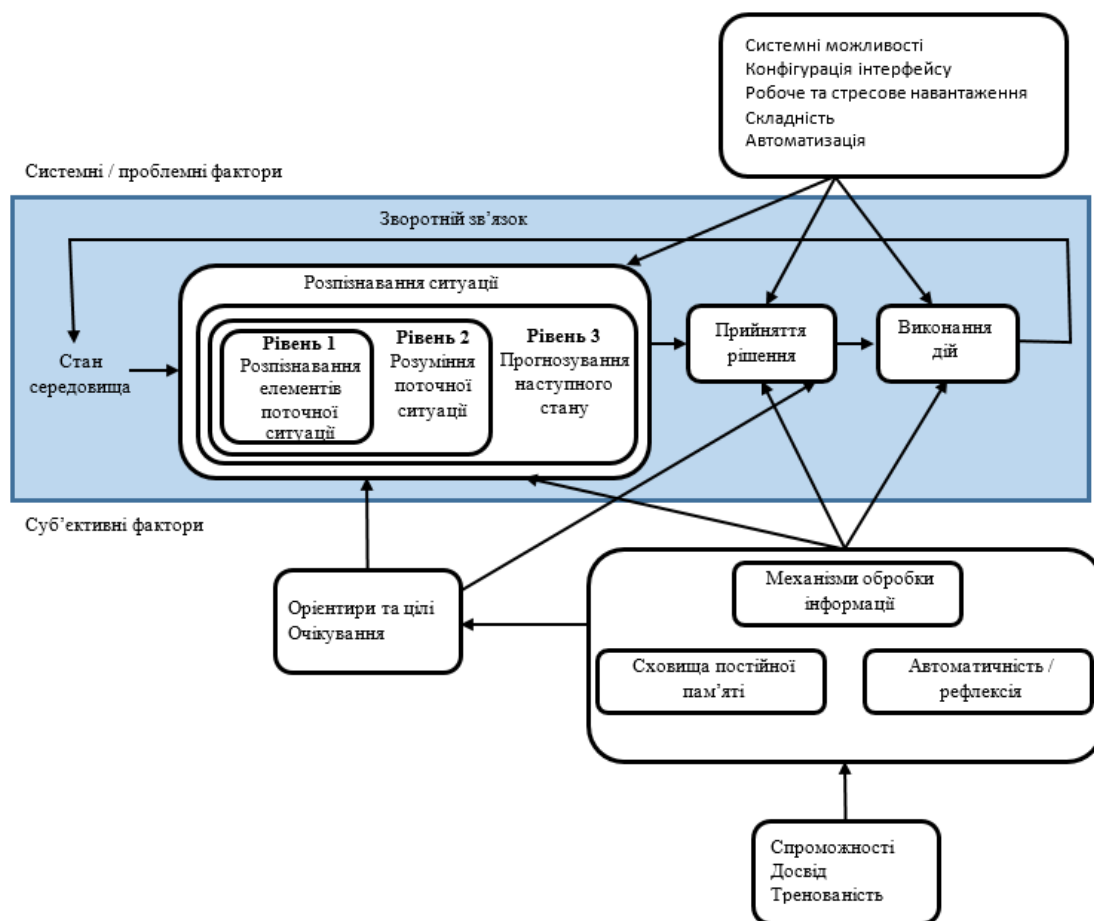


Рисунок 1.10 – Трирівнева модель ситуаційного управління при динамічному прийнятті рішень

Модель зображує СУ як складову ланцюжка (послідовності) обробки інформації, що настає після сприйняття й приводить до прийняття рішень та виконання відповідних дій. Відповідно до моделі, на здійснення та підтримку СУ впливають різні фактори, включаючи індивідуальні фактори (наприклад, досвід, навчання, робоче навантаження тощо), фактори завдання (наприклад, складність) та системні фактори (наприклад, дизайн інтерфейсу).

Процес впровадження та підтримки ситуаційного управління відбувається навколо внутрішньої притаманної схеми (ментальної моделі), яка містить інформацію про певні ситуації. Ця схема полегшує передбачення ситуаційних подій, спрямовуючи увагу ОПР на сигнали в навколишньому середовищі та направляючи їх можливий хід дій. Потім ОПР проводить перевірки, щоб переконатись, що розвиток ситуації відповідає його очікуванням. Неочікувані події слугують підказками для подальшого дослідження та пояснення, які, у свою чергу, змінюють існуючу ментальну модель.

Динамічна модель ситуаційного управління Ендслі [153], адаптована до перцептивного циклу Нейссера, представляє СУ [154] у контексті циклу перцепції (рис. 1.11), що дозволяє аналізувати СУ як циклічну модель зі зворотним зв'язком, подібну до перцептивного циклу.



Рисунок 1.11 – Динамічна модель ситуаційного управління Ендслі, адаптована до перцептивного циклу Нейссера

Процесний підхід до СУ фокусується на тому, як СУ змінюється з часом, та стосовно інших процесів, таких як прийняття рішень. Як показано на рис. 1.12 [154], СУ – це процес, який генерує горизонт подій. Ключовим механізмом процесної

моделі СУ є взаємний вплив «дія↔зворотний зв'язок» перцептивного циклу (рис. 1.7). Він включає фрейми (виділені сцени, кадри), контексти (внутрішній зміст), об'єкти та горизонт подій, не тільки спрямовуючи дії, модифікуючи оточення, але й підбираючи зразки сигналів/об'єктів та модифікуючи СУ. Крім того, у складних системах СУ фізичні маніпуляції можуть змінювати прояви СУ, при цьому можуть відбуватися транзакції між різними проявами СУ.

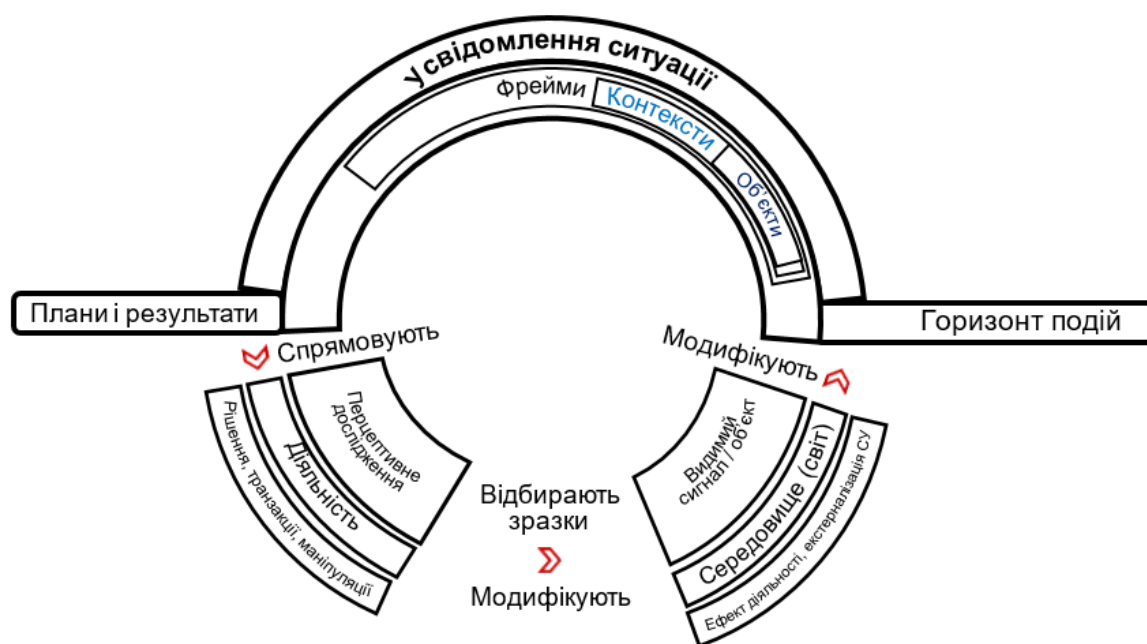


Рисунок 1.12 – Модель ситуаційного управління орієнтована на створення горизонту подій

Формалізований підхід до СУ запропонував Д.А. Поспелов [139]. Він увів поняття поточної та повної ситуацій, на основі яких описав процес СУ як процес вироблення керуючих впливів на об'єкт управління шляхом трансформації вхідної інформації про поточну ситуацію з використанням логіко-трансформаційних правил (ЛТП).

Повні ситуації позначались через S_i (i – унікальний номер ситуації), а поточні ситуації – через Q_j (j – унікальний номер ситуації). Нехай у розпорядженні системи управління є n різних способів впливу на об'єкт управління (однокрокових

рішень). Кожне таке рішення ми будемо позначати як U_k (k є унікальний номер впливу). Елементарний акт управління можна представити в такому вигляді:

$$S_i : Q_j \xRightarrow{U_k} Q_l.$$

Сенс цього співвідношення полягає в такому. Якщо на об'єкті управління склалася ситуація Q_j , і стан системи управління та технологічна схема управління, які визначаються S_i , допускають використання впливу U_k , то воно застосовується, і поточна ситуація Q_j , перетворюється в нову ситуацію Q_l . Подібні правила перетворення в подальшому називаються логіко-трансформаційними правилами (ЛТП) або кореляційними. Повний список ЛТП задає можливості системи управління впливати на процеси, що протікають в об'єкті.

Модель процесів ситуаційного управління, побудована на основі концепції Д.А. Поспелова, представлена на рис. 1.13. Опис поточної ситуації, що склалася на об'єкті управління, аналізується на предмет необхідності втручання.

Після оцінки ситуації визначається необхідність втручання системи управління в процес, що протікає в об'єкті управління. Якщо поточна ситуація не вимагає такого втручання, то подальша обробка не здійснюється. В іншому випадку опис поточної ситуації проходить класифікацію з використанням класифікаційної інформації. На основі класифікації поточна ситуація відноситься до одного або декількох класів, яким відповідають однокрокові рішення. Ця інформація використовується далі для вибору одного з відомих ЛТП, яке повинно бути використано для впливу на об'єкт управління. Якщо таке правило єдине, то воно видається для виконання. Якщо ж таких правил декілька, то обґрунтований вибір кращого ЛТП із множини альтернатив проводиться після обробки попередніх рішень, яке використовується для впливу на об'єкт. Якщо обґрунтувати вибір ЛТП із множини альтернатив за описом поточної ситуації неможливо, то випадковим чином вибирається одне з ЛТП, яке дає не надто великий вплив на об'єкт, або ж приймається рішення про відмову від будь-якого впливу на об'єкт.

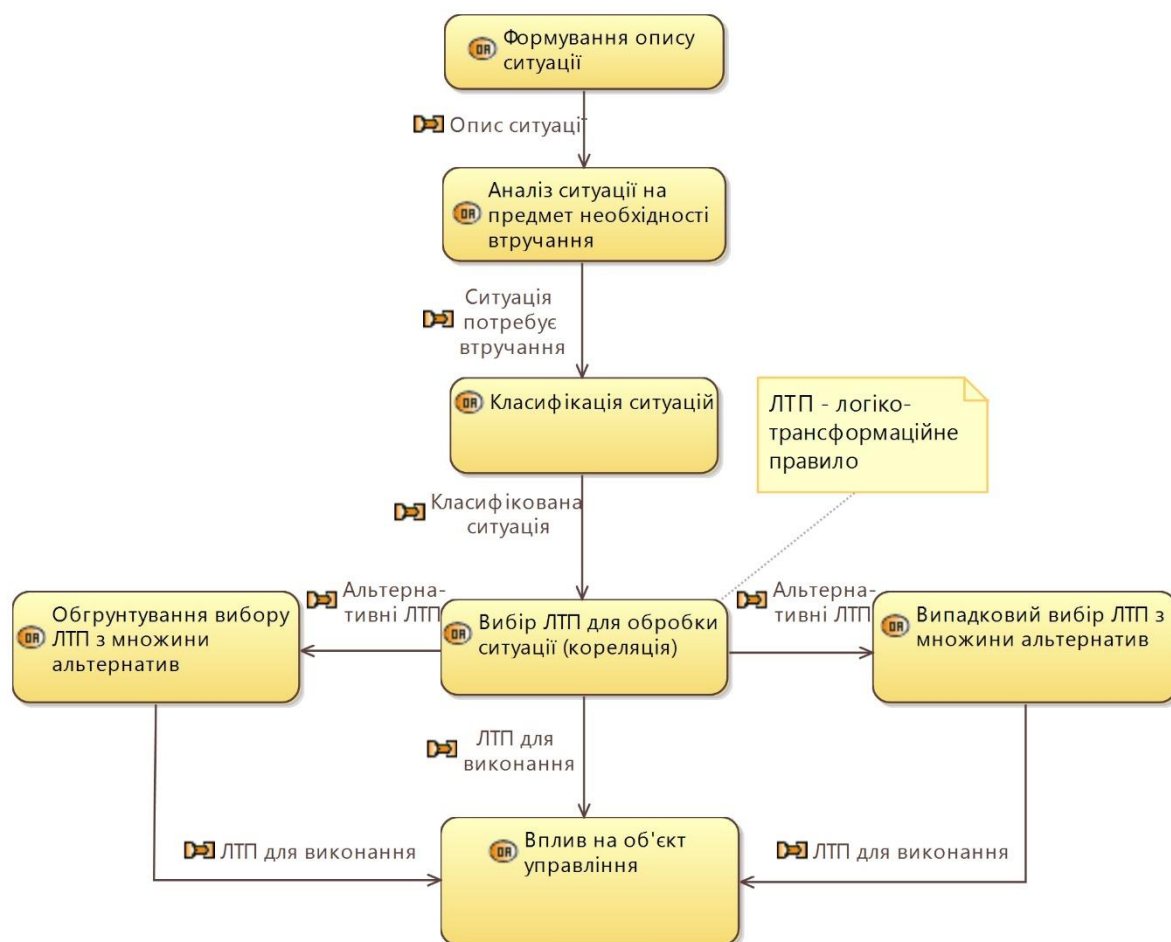


Рисунок 1.13 – Процеси ситуаційного управління за моделлю Д.А. Поспелова (в нотації ARCADIA)

Процеси ситуаційного управління за моделлю Д.А. Поспелова можуть бути приведені до процесів перцептивного циклу ситуаційного управління. Приведена модель процесів наведена у табл. 1.3.

Таблиця 1.3 – Відповідність процесів ситуаційного управління

Процеси ситуаційного управління	Процеси перцептивного циклу
Сприйняття змін у середовищі (предметній сфері)	Усвідомлення ситуації
Формування опису (усвідомлення) ситуації	Фрейми, контексти, об'єкти
Аналіз ситуації	Перцептивне дослідження

Продовження таблиці 1.3

Класифікація ситуації	Відбір та модифікація зразків (патернів)
Вибір ЛТПП для обробки ситуації: прецедентний, обґрунтований, випадковий	Екстерналізація СУ, формування сценаріїв та сигналів
Вплив на об'єкт управління	Формування горизонту подій

Модель ситуаційного управління, запропонована А.О. Морозовим, передбачає залучення до СУ груп експертів. Ефективні команди експертів значною мірою займаються взаємним моніторингом результатів або відстеженням їх взаємної оперативної ситуації та оточення. Почаючи з ранніх моделей СУ і дотепер важливою проблемою залишається визначення спільних задач для команди експертів і розподіл роботи між ними.

Найважливішим компонентом СУ в команді є комунікація, причому реалізація СУ залежить від обміну інформацією, необхідної для досягнення цілей місії, визначення індивідуальних завдань, ролей і можливостей команди поряд з іншими факторами формування продуктивності команди. СУ передбачає більше, ніж просто вербальний обмін; воно включає інформацію у вигляді стану завдання, відображення інформаційних та багатьох інших сигналів, які існують у складних групових середовищах. Таким чином, відбувається циклічний процес, у рамках якого члени команди здійснюють індивідуальне СУ, розділяють СУ з іншими членами команди, а потім змінюють як командне, так і індивідуальне СУ на основі усвідомлення ситуації іншими членами команди.

СУ в команді виникає як наслідок взаємодії відповідних попередніх знань та очікувань індивідуума, наявної інформації з навколишнього середовища та навичок когнітивної обробки, які включають зосередження уваги, сприйняття, виділення даних, розуміння та проєкцію (інтерпретацію) (рис. 1.14). З цього еволюційного

розуміння виникла потреба передбачати СУ через різні деталізації та рівні аналізу (від мікро до макро).

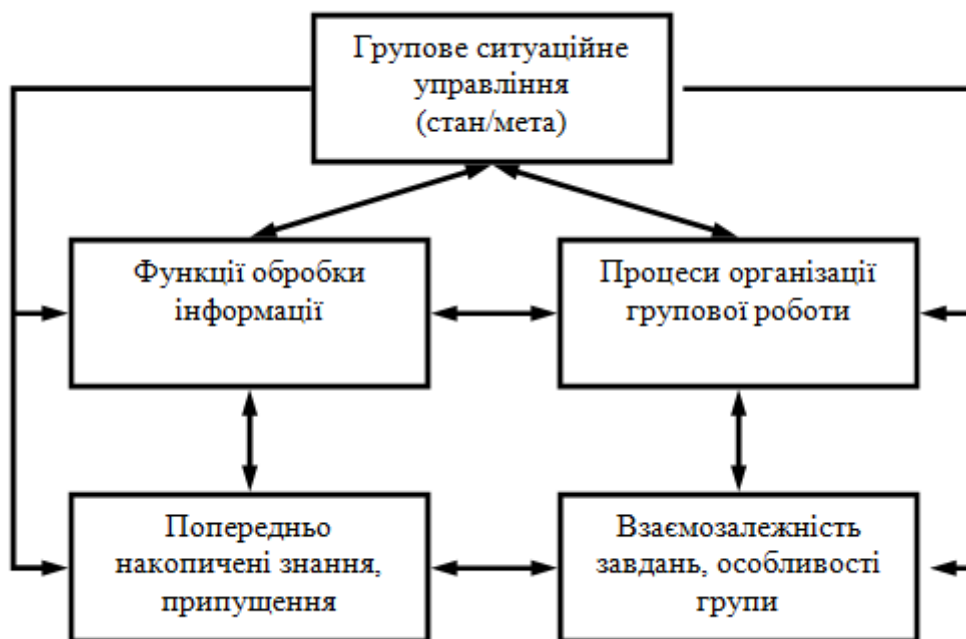


Рисунок 1.14 – Колективне (групове) ситуаційне управління

Розподілене СУ. Системне мислення представляє різні проблеми. Соціально-технічні (кіберлюдські, кіберсоціальні) системи описують поєднання людей із технічними елементами, які взаємодіють так, щоб підтримувати організаційну діяльність. На основі цього підходу у системному мисленні розроблена модель розподіленого СУ. Важливою особливістю моделі розподіленого СУ є те, що вона передбачає періодичну необхідність повернення назад до індивідуальної деталізації.

Індивідууми здійснюють безперервний циклічний процес сприйняття і дій у середовищі, керованому схемою, і загальне керування оновленням і запуском схеми. Із внутрішньо притаманною схемою і завданням активується схема, причому перша побудована протягом часу під впливом різних ситуацій. Останні запускаються під час діяльності функціями, які можна узагальнити в широкому класі ситуацій.

У розподіленій моделі СУ розглядається як нова властивість співпрацюючих систем, що виникає в результаті взаємодії між залученими агентами, як людськими, так і технологічними [47], [78], [155], [156], [157], [158]. Розуміння того, що СУ не

обмежується людьми в системах, часто викликає концептуальні труднощі, але ця ідея має широке застосування у підході до розподіленого пізнання [159].

Згідно з роботою [160], обізнаність системи включає в себе мережу інформації, в якій різні компоненти цієї системи мають різні види та право власності. Крім того, кожен компонент самостійно вносить різну інформацію до ситуації.

Ключем до розподіленого СУ є те, що ці різні види пов'язані таким чином, що відповідна інформація надається безпосередньо агенту в потрібний час. Це пояснюється тим, що замість того, щоб мати спільне розуміння ситуації (яке передбачає, що члени команди розуміють ситуацію або елементи ситуації однаково), модель розподіленого СУ замість цього пропонує таке [160]:

- члени команди мають унікальні, але сумісні типи розуміння ситуації;
- члени команди розуміють ситуацію по-різному, як це обумовлено власним особистим досвідом, цілями, ролями, завданнями, навчанням, навичками, схемою тощо;
- однакове розуміння – це те, що поєднує разом подібні розподілені системи.

З аналізу моделей ситуаційного управління (ситуаційно-орієнтована поведінка) видно, що процеси ситуаційного управління реалізуються в рамках перцептивного циклу (мають циклічний характер) і включають етапи емпіричного усвідомлення стану середовища (цільової області), побудови його моделі та застосування цієї моделі при формуванні раціональної поведінки у середовищі на основі періодичного оновлення усвідомлення поточного стану середовища (цільової області). Усвідомлення стану середовища та формування на його основі раціональної поведінки здійснюються з використанням механізмів логічного умовиводу, що відповідають етапам перцептивного циклу. Такими механізмами логічного умовиводу у циклі ситуаційної взаємодії з середовищем є індукція, дедукція та абдукція [161], [162]. Тоді модель ситуаційного управління з урахуванням механізмів умовиводу буде мати вид, представлений на рис. 1.15 [7].

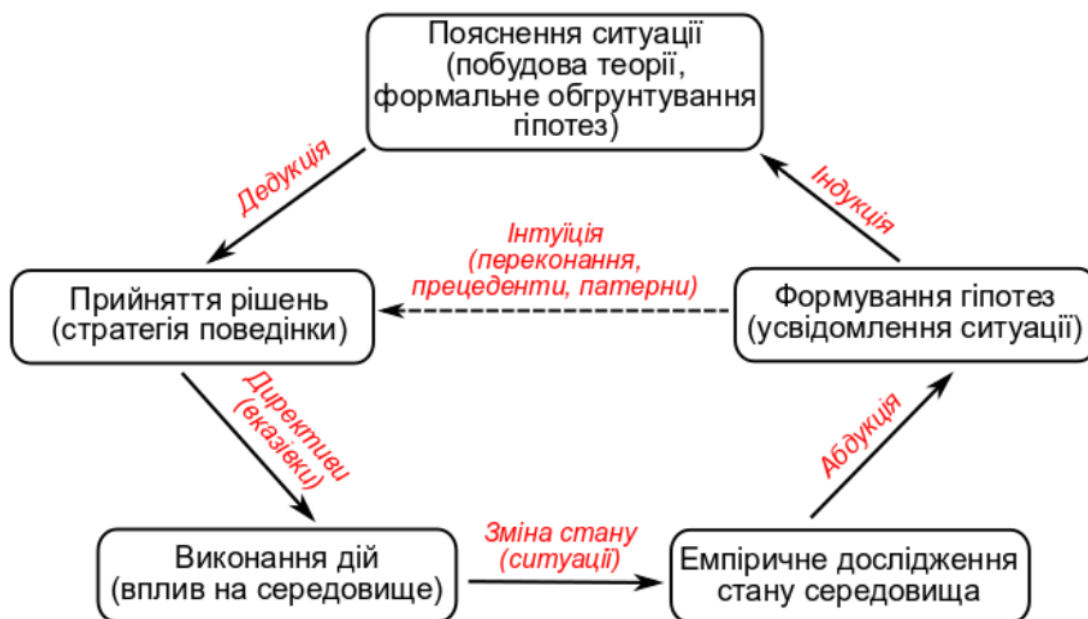


Рисунок 1.15 – Модель процесів циклу ситуаційного управління (модель механізмів перетворення інформації в циклі СУ)

Абдуктивний умовивід широко застосовується у сценарно-прецедентних ергатичних (поліергатичних) системах [163], [164], [165], [166]. Абдуктивне міркування найчастіше використовується для відкриття емпіричних законів, які встановлюють необхідні регулярні зв'язки, що спостерігаються між властивостями і відносинами явищ. Теоретичні закони не можуть бути відкриті таким шляхом, оскільки вони містять абстрактні поняття, які не можна спостерігати на досвіді. Тому шлях до них йде через гіпотези або системи гіпотез, які перевіряються зазвичай шляхом логічного висновку з емпіричних законів. Як свідчить історія науки, саме так фактично відбувалися відкриття теоретичних законів і побудова цілісних теорій і теоретичних систем [167]. Оскільки абдуктивний умовивід є механізмом формування правдоподібних гіпотез, то на його основі може бути побудована квазіадекватна модель поведінки.

Більш обґрунтованим є формування поведінки в рамках СУ на основі застосування індуктивно-дедуктивних механізмів умовиводу.

Підхід на основі доказових міркувань використовується на етапі прийняття рішень у ССУ і є загальним підходом до багатокритеріального аналізу рішень (БКАР, multi-criteria decision analysis, MCDA) на основі доказів для вирішення

проблем, що мають як кількісні, так і якісні критерії при різних невизначеностях, включаючи незнання і випадковість. Вона була використана для підтримки різних заходів аналізу, оцінювання та оцінки, таких як оцінка впливу на навколишнє середовище [168], [169] та організаційна самооцінка [170] на основі ряду (множини) якісних моделей.

Індукція та абдукція є способами підсилювальних (амплікативних) міркувань. Абдуктивним, або змістовнозростаючим, є тип міркування, в якому висновок логічного виведення перевищує те, що вже викладено в антецеденті (посилці). Таким чином, істинність висновку не гарантується, коли посилка є істинною. Це цілком узгоджується з істинністю припущення, що висновок є помилковим. Раніше філософи переважно зосереджувалися на проблемі правомірності (проблема Юма): враховуючи, що абдуктивні міркування не обов'язково доводять істину. Як можна це довести? Тобто, як ми можемо показати, що раціонально використовувати амплікативну аргументацію?

Інтуїтивні рішення базуються на стереотипних знаннях і досвіді та використовують прецедентний умовивід (Case Based Reasoning).

Таблиця 1.4 – Порівняння моделей СУ

Модель СУ Етап СУ	Модель перцептив- ного циклу Нейссера	Екологіч- на модель перцеп- тивного циклу	Трирівне- ва модель ситуацій- ного управлін- ня Ендслі	Модель створен- ня гори- зонту подій	Модель СУ Поспело- ва	Модель СУ Морозова
Емпіричне досліджен- ня стану середовища	Явне	Явне	Явне	Явне	Зовнішнє	Зовнішнє

Продовження таблиці 1.4

Формування гіпотез (описів ситуації), абдукція	Неявне	Неявне	Неявне	Явне	Явне	Експерт-не
Пояснення ситуації (аналіз ситуації, побудова теорії, формальної моделі)	Неявне	Неявне	Неявне	Явне	Явне	Експерт-не
Класифікація ситуацій (патерни, прецеденти)	Неявне	Неявне	Неявне	Явне	Корелятор	Експерт-на система

Продовження таблиці 4.1

Прийняття рішень (Вибір ЛТП для обробки ситуації: прецедент- ний, обґрун- тований, випадко- вий)	Явне	Явне	Явне	Явне	Неявне	Експерт- на система
Виконання дій (вплив на об'єкт управління)	Прямий	Прямий	Прямий	Прямий	Детермі- нований	Адаптив- не

1.5 Система ситуаційного управління як кіберматична система

Процес СУ пов'язаний із трансформацією інформації з метою вироблення та впровадження ефективних рішень у різних сферах людської діяльності на основі використання кібернетичних підходів та інформаційних технологій. Отже, ССУ можуть бути віднесені до кіберматичних систем.

Кіберматика (Cybermatics) – це цілісна область досліджень для систематичного вивчення кібер-об'єктів у кіберпросторі, їх властивостей і функцій, а також їхніх зв'язків і відношень з об'єктами у фізичних, соціальних і ментальних (розумових) просторах (рис. 1.16). Загалом, кібероб'єкт (сутність) можна визначити

як такий, що існує в кіберпросторі, або є суто синтезованим комп'ютером, або тісно пов'язаним з реальним об'єктом у фізичному, соціальному та ментальному (розумовому) просторах [171]. Ця сфера включає кібернетичну науку, фізику, соціологію, психологію, медицину, науки про життя, техніку, технології тощо. Кіберматику можна розглядати як всеосяжну сферу, в якій усі дослідники у сфері кібернетики проводять спільні дослідження і координують роботу над ключовими проблемами у новій кібернетичній сфері та інших нових сферах, пов'язаних з кібернетикою.

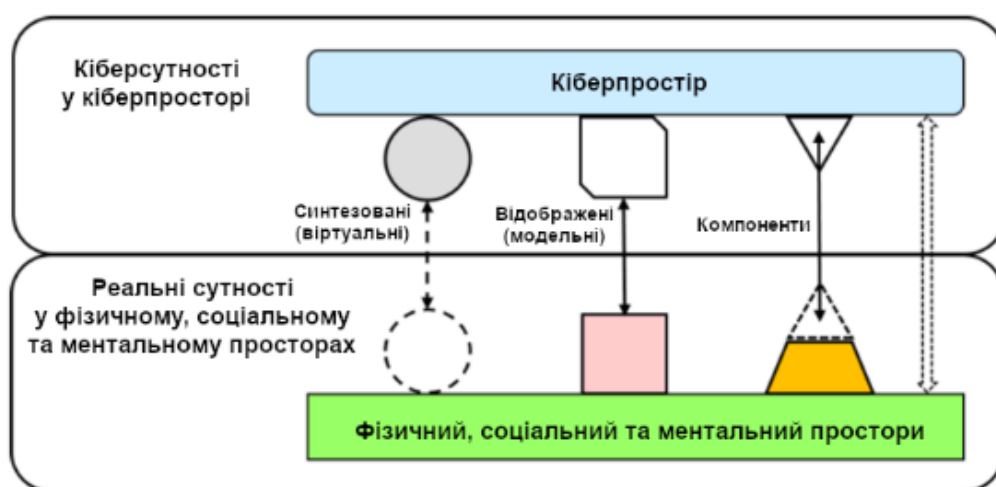


Рисунок 1.16 – Відношення між сутностями кіберсвіту та реальними сутностями фізичного, соціального (людського) та ментального світів

Кіберматика орієнтована в основному на кіберсущності (cyber-entities, CE), які існують у кіберсвіті і можуть мати відношення з об'єктом у кіберпов'язаних світах. Таким чином, кіберматика поділяється на дві основні категорії досліджень, відповідно: категорію кіберсвіту і кіберпов'язану категорію, як показано на рис. 1.17.



Рисунок 1.17 – Загальні категорії кіберматики для кіберпов'язаних світів

Кіберлюдські системи (КЛС, Cyber-Human Systems, CHS) – це категорія кіберматичних систем та прогресивний напрям інформаційних технологій. КЛС сформувалась на основі таких напрямів досліджень, як обчислення, орієнтовані на потреби користувачів, взаємодія людини з комп'ютером, універсальний доступ, цифрове суспільство і технології, а також певною мірою цифровий уряд, конфіденційність інформації та взаємодія людини з роботами. Область (вимір) людської діяльності простягається від досліджень, які підтримують, розширюють можливості людини, до реакції на потреби індивідів у складі команд як цілісних цілеспрямованих груп, та у суспільстві як неструктурованих об'єднань пов'язаних людей. Область (вимір) комп'ютеризації простягається від стаціонарних обчислювальних пристроїв, поруч з якими повинна бути людина, через мобільні пристрої, які повсюдно супроводжують людей, до обчислювальних систем датчиків та візуальних/аудіопристроїв, вбудованих у навколишнє фізичне середовище. Простір (вимір) середовища простягається від дискретних фізичних обчислювальних пристроїв до віртуальних середовищ занурення з системами змішаної реальності в середині цього простору. На рис. 1.18 [172] показано підмножину напрямів досліджень КЛС у цьому тривимірному просторі, що показує напрями розвитку та конвергенції зазначених вимірів діяльності.

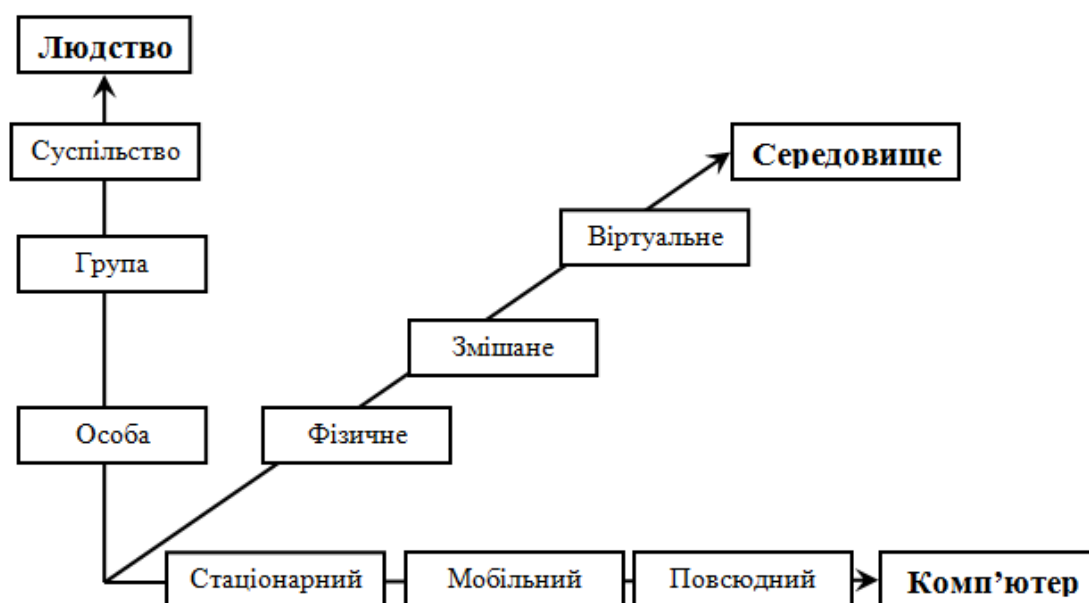


Рисунок 1.18 – Три виміри кіберлюдських систем

У світі, де сфера застосування комп'ютерів та комп'ютерних мереж постійно розширюється, комп'ютерна обробка інформації та комп'ютерні обчислення відіграють центральну роль в організації роботи, навчання, життя, досліджень і спілкування людей. Технології все більше проникають у суспільство і стають звичайним явищем практично у будь-якій діяльності. Межа між людьми і технологіями зменшується до рівня, коли соціально-технічні системи стають природним розширенням людського досвіду – другою натурою, допомагаючи, піклуючись і вдосконалюючи людей. Як наслідок, обчислювальні технології та життєвий цикл людей, організацій і суспільства спільно еволюціонують, перетворюючи один одного в ході цього процесу. Дослідження КЛС (CHS) зосереджені на потенційно перетворюючих та проривних ідеях, нових теоріях та технологічних інноваціях у галузі комп'ютерних наук та інформатиці, які прискорюють як створення, так і розуміння складних і все більш тісних зв'язків між людьми та технологіями з загальною метою розвитку людських можливостей: сенсорних (сприйняття) та пізнавальних, фізичних та віртуальних, соціальних та суспільних [173], [174], [175].

Розвиток може мати багато форм і залучати різноманітні обчислювальні платформи. Ці платформи можуть включати “традиційні” пристрої і системи, такі як

звичайні комп'ютери та портативні або мобільні пристрої. Однак все частіше межа зсувається до кіберлюдських технологій у світі «постмобільного та web-інтерфейсу», включаючи роботів, носіїв, вбудовані датчики та комп'ютери, цифрове виробництво та програмування, віртуальну та доповнену реальність об'єктів і середовищ, розподілені і мережеві соціально-технічні системи, соціально-інтелектуальні обчислення і хмарні обчислення [18], [55], [176]. Ці технології все частіше використовують розумні об'єкти та системи із покращеним "посередництвом", разом з іншими менш традиційними інтерфейсами, такими як системи, що взаємодіють із користувачами за допомогою різноманітних і, можливо, декількох можливостей (модальностей), включаючи інноваційні комп'ютерні або візуальні дисплеї; інтерфейси тактильних, аудіо- та інтелектуальних машин; а також інші методи трансформативної взаємодії. КЛС здатні працювати у масштабах від окремого пристрою з одним користувачем, до мережевих і хмарно-інформаційних систем, що підтримують роботу колаборативних груп, до великих, еволюційних, неоднорідних соціально-технічних систем, що підтримуються первазивною (всеохоплюючою) мережею, та до систем, які об'єднують фізичний і віртуальний світи і людей.

Дослідження КЛС сприяє науковому розумінню обчислювальних та комунікаційних систем разом з теоретичним та практичним розумінням поведінкових, соціальних та проєктних наук для кращого проєктування та розробки різноманітних систем шляхом:

- посилення індивідуальних спроможностей людини за допомогою пристрою або середовища, що надає можливість підвищити продуктивність, досягти своїх цілей, покращити добробут і розширити творчі самовираження, гарантуючи, що комп'ютер більше не буде відволікати або перешкоджати;
- збагачення життєвих спроможностей і самодостатності людей з обмеженими можливостями, включаючи людей із тимчасовими або ситуативними обмеженнями;
- поліпшення інтелекту систем із більшою автономністю, які вимагають різного рівня контролю з боку людини; це включає в себе більш симбіотичні відносини між людиною і машиною через розвиток систем, які можуть відчувати і

вивчати людські пізнавальні і фізичні стани, володіючи здатністю відчувати, вчитися і адаптуватися у своєму середовищі;

- розширення обчислювальних середовищ, у тому числі віртуальної та/або доповненої реальності, для забезпечення та вдосконалення наукової, інженерної й освітньої продукції та інновацій;

- розширення можливостей людей та комп'ютерів для спільної роботи, поширення й засвоєння інформації і знань в міру необхідності, вирішення раніше нерозв'язаних проблем та здійснення спільних дій, які неможливо робити окремо;

- вдосконалення комп'ютерних мереж та комунікацій, які допомагають командам, групам, організаціям і громадам досягати колективних цілей;

- використання комп'ютерної обробки для підвищення згуртованості, інноваційності, безпеки та сталості розвитку суспільства.

Крім того, дослідження КЛС мають на меті поліпшити фундаментальне розуміння того, як і за допомогою яких процесів інтерактивні системи повинні бути розроблені для досягнення симбіозу людини і комп'ютера та комп'ютерно-опосередкованого людського спілкування, співпраці та конкуренції. У рамках КЛС досліджуються проекти соціально-технічних систем, які приводять до нових способів пізнання себе, груп, організацій, спільнот і суспільства, а також ширшого впливу цих проектів як на окремих людей, так і на суспільство в цілому. Це розширює використання комп'ютерної обробки новими спільнотами і сприяє розумінню новостворених спільнот, що стало можливим завдяки комп'ютерній обробці.

Завдяки партнерству та залученню різних і нових дисциплін, дослідження КЛС привносять нові технології, нові перспективи і нові моделі дослідження, практику і теорію у дослідження і освіту в галузі комп'ютерних наук та інформаційних технологій. Не зважаючи на різноманітність підходів, дослідження КЛС обґрунтовуються теоретично і вкорінені в емпіричних методах для отримання широко застосовуваних і поширюваних результатів, які просувають теоретичні і практичні знання у сфері КЛС.

Найважливішою задачею побудови конвергентної ССУ є створення системної моделі конвергенції технологій. Основою для такої моделі є опис знань предметної сфери СУ. Конвергенція частинних моделей у загальній моделі знань СУ забезпечує методологічний базис для побудови ССУ різного призначення.

1.5 Конвергентний підхід при побудові комп'ютерних систем

1.5.1 Конвергенція знань і технологій

Конвергенція, у загальному розумінні, – це глибока інтеграція знань, засобів та усіх раціональних видів людської діяльності для досягнення спільної мети, можливостей отримання відповіді на нові питання щодо зміни відповідної фізичної або соціальної екосистеми. Такі зміни у відповідній екосистемі відкривають нові тенденції, шляхи та можливості розділення на наступному етапі цього процесу [177], [178].

Конвергенція кібернетичних підходів та інформаційних технологій у ССУ здійснюється на основі використання предметних знань інженерії систем та управління життєвим циклом систем.

У роботах М. Рокко та В. Баїнбріджа [177], [179], [180] визначено основні напрями конвергенції та дивергенції знань із різних галузей діяльності та їх вплив на сталий розвиток людства.

Конвергенція знань і технологій на благо суспільства (КЗТС, convergence of knowledge and technology for society, CKTS) є основною можливістю для прогресу в ХХІ столітті. КЗТС визначається як розгортання та трансформативна взаємодія між, здавалося б, різними дисциплінами, технологіями, спільнотами та сферами людської діяльності для досягнення взаємної сумісності, синергізму та інтеграції, а також створення, завдяки цьому процесу, додаткової цінності та розгалуження для досягнення спільних цілей. За останні кілька десятиліть розвиток конвергенції почався з фізичних нанотехнологій, після чого поетапно відбувається зближення

нанотехнологій, біотехнологій, інформації та когнітивної науки (NBIC) у нових технологіях.

КЗТС є третім рівнем конвергенції. Вона визначає загальний процес для розвитку творчості, інновацій та суспільного прогресу на основі п'яти загальних принципів:

- 1) взаємозалежність усіх компонентів природи та суспільства;
- 2) аналіз рішень для досліджень, розробок та застосувань на основі динамічної системно-логічної дедукції;
- 3) підвищення творчого потенціалу та інновацій через еволюційні процеси конвергенції, що поєднує в собі існуючі принципи, та дивергенції, що генерують нові;
- 4) використання міждомених (міжгалузевих) мов високого рівня для створення нових рішень та підтримки перенесення нових знань;
- 5) цінність продуктивних фундаментальних досліджень, втілених у вирішення важливих проблем.

КЗТС – це універсальний підхід щодо вирішення комплексних проблем у суспільстві знань. Вона дозволяє суспільству відповідати на запитання та вирішувати проблеми, які неможливо вирішити в межах ізольованих областей діяльності, а також створити на цій основі нові компетенції, знання та технології.

Конвергенція може виникати як між різними галузями (віддалена конвергенція), так і в середині однієї галузі між окремими незалежними напрямками.

1.5.2 Конвергентні інфраструктури

Конвергентна інфраструктура або конвергентна архітектура [181] – це реалізація комплексного підходу до управління центрами обробки даних, який забезпечує поєднане в одному програмному інструменті управління п'ятьма ключовими складовими комп'ютерної системи: мережею, сервером, обчисленнями,

зберіганням інформації та віртуалізацією. Конвергентні системи стали популярними серед постачальників послуг мобільного зв'язку, які намагаються відійти від володіння власним обладнанням та управління ним. На відміну від володіння всім необхідним обладнанням, об'єднана інфраструктура – це більше система самообслуговування, що дозволяє ІТ-адміністраторам використовувати ресурси за потребою у потрібний час, не займаючись управлінням системою в цілому [182].

Традиційні системи управління центрами обробки даних (ЦОД) складаються із множини різних контейнерів, які повинні бути індивідуально налаштовані окремою ІТ-командою перед їх інтеграцією з іншими компонентами [183]. Конвергентна інфраструктура поєднує в собі декілька ІТ-ресурсів, завдяки чому провайдери керованих сервісів можуть отримати доступ до всього, необхідного для повного управління ЦОД в одному застосунку. Це також звільняє від необхідності зводити разом функціональний стек різних технологій. Завдяки інтеграції апаратного та програмного забезпечення у єдиній системі, підвищується ефективність управління використанням ресурсів.

Конвергентні системи будуються за модульним принципом і представляють усі доступні ресурси системи в єдиному пулі, забезпечуючи високу ефективність управління використанням ресурсів в цілому [182]. Такий підхід є набагато ефективнішим та точнішим, ніж необхідність постійного визначення покомпонентної доступності ресурсів.

Метою конвергентної інфраструктури є суттєве спрощення процесу управління ЦОД та усунення будь-яких проблем із сумісністю апаратного забезпечення. Це особливо привабливо для постачальників мобільних сервісів, які створюють власні хмарні платформи або розміщують та експлуатують сервіси у власній приватній хмарі.

Робота в об'єднаній інфраструктурі надає такі переваги:

- зниження витрат на інтеграцію, комунікації та підтримку параметрів середовища завдяки зменшенню загальної кількості інфраструктурних елементів;
- покращення видимості споживання ресурсів;
- краща масштабованість;

- підтримка вбудованих та гарантованих рішень постачальника;
- можливість самостійно налаштовувати різні компоненти конвергентної ІТ-інфраструктури.

Також треба враховувати і недоліки конвергентного підходу. По-перше, оскільки конвергентні системи попередньо налаштовуються відповідно до потреб конкретної інфраструктури, конфігурацію якої змінити не можна. Цей недолік не є суттєвим, якщо операційна діяльність залишається незмінною протягом тривалого періоду в одній конфігурації об'єднаної інфраструктури, але виникають проблеми, якщо відбуваються коливання налаштувань. По-друге, додавання додаткових компонентів може бути ускладненим і витратним, що, по суті, знижує переваги конвергентної системи.

Конвергентна платформа дозволяє керувати та моніторити потужність ЦОД за допомогою розпізнавання даних. Ця функція допомагає адміністраторам системи оцінити в будь-який момент часу кількість доступних ресурсів ЦОД: сховищ інформації, пам'яті, обчислювальних потужностей, мережевих ресурсів тощо.

Існує безліч варіантів розгортання конвергентної інфраструктури. Найпопулярніший варіант розгортання передбачає використання еталонної архітектури конвергентної інфраструктури постачальника сервісів. Архітектура конвергентної системи постачальника повинна підтримувати швидке розгортання у правильний спосіб і правильне використання компонентів. Вона включає попередньо налаштовані рекомендації щодо обладнання, які ізолюють частину робочого навантаження в ЦОД. Рішення, перевірені постачальником, є надзвичайно важливою та корисною функцією, оскільки вони вилучають усі припущення з процесу розгортання та скорочують час реалізації.

Також можна використовувати попередньо підготовлені конфігурації, розміщені в ЦОД для швидкого налаштування. Крім того, є можливість відмовитись від попередньо встановлених параметрів та придбати окремі засоби, на яких буде розгорнутись окрема об'єднана інфраструктура.

Загалом, головною перевагою конвергентної інфраструктури є те, що вона допомагає постачальникам сервісів оптимізувати, централізувати та впорядкувати

управління центрами обробки даних. Конвергентна інфраструктура усуває всі перешкоди, пов'язані з традиційним підходом до управління ЦОД.

Крім того, об'єднана інфраструктура є чудовим варіантом управління центром обробки даних для постачальника сервісів, які забезпечують роботу в хмарі або з нею. Багато характеристик конвергентної платформи роблять її придатною для розгортання у хмарному середовищі, а саме здатність масштабуватися вгору і вниз та можливості автоматичного забезпечення. Можна використовувати конвергентну хмарну інфраструктуру для підтримки різноманітних служб хмарних обчислень, включаючи платформу як послугу (PaaS), інфраструктуру як послугу (IaaS) та програмне забезпечення як послугу (SaaS).

Додатково до зазначених вище переваг конвергентна інфраструктура має такі переваги:

- спрощена система управління центром обробки даних;
- доступ до масштабованої системи зберігання;
- швидший період забезпечення – від трьох тижнів до менше години;
- можливість швидшого реагування на зміни в галузі та бізнес-запити;
- простіша інтеграція з послугами хмарних обчислень;
- загалом більший контроль над системою;
- менше годин, витрачених на встановлення та тестування систем.

1.5.3 Гіперконвергентні інфраструктури

Гіперконвергентні інфраструктури розвивають концепцію конвергентних структур, додаючи в неї поняття модульності. Гіперконвергентна інфраструктура має модульну архітектуру. Завдяки цьому, всі необхідні віртуалізовані обчислювальні ресурси, мережеві і системи зберігання даних (СЗД) працюють автономно в середині окремих модулів, які представляють собою готові віртуалізовані обчислювальні ресурси [184], [185]. Вони зазвичай об'єднуються у

групи, щоб забезпечити відмовостійкість, високу продуктивність і гнучкість у створенні ресурсних пулів.

Під терміном «гіперконвергентна інфраструктура» (Hyper-Converged Infrastructure, HCI) розуміють програмно-керовану IT-інфраструктуру, яка включає в себе такі обов'язкові компоненти, як гіпервізор віртуалізації обчислень, програмно-визначені сховище даних і мережу. Гіперконвергентні технології з відкритим кодом є ефективним інструментарієм для побудови гнучкої IT-інфраструктури хмарного рівня в сучасній організації (підприємстві) без використання публічних ресурсів.

1.5.4 Порівняння конвергентних і гіперконвергентних інфраструктур

Конвергентна інфраструктура та гіперконвергентна інфраструктура – це назви, які часто неправильно використовуються для посилення на одну і ту ж систему управління даними. Ці дві системи можуть досягти однакових цілей – всебічного управління ЦОД, але вони використовують різні підходи.

Гіперконвергентна інфраструктура – це програмний підхід, керований програмним забезпеченням, який відокремлює мережеві, обчислювальні компоненти та застосунки від фізичного обладнання й надає розширені можливості віртуалізації. Гіперконвергентні системи також мають більш спеціалізовані засоби для вирішення таких проблем, як розрив хмар та відновлення після аварій. За допомогою єдиного інтерфейсу провайдери керованих сервісів можуть управляти як фізичною, так і віртуальною інфраструктурою як локально, так і у хмарі.

Гіперконвергентна інфраструктура не потребує складної системи налаштувань і добре підходить як для малих, так і для середніх організацій, що займаються наданням інформаційних сервісів. Можна інтегрувати гіперконвергентну систему з декількома різними системами, але при нарощуванні потужності, з додаванням більшої кількості ресурсів, немає можливості обрати, які компоненти постачальника найкраще підходять для цільової системи.

Конвергентна інфраструктура та гіперконвергентна налаштовані по-різному. У конвергентній інфраструктурі кожен компонент є будівельним блоком для більшої системи, тому теоретично можна видалити та замінити компонент або два, і загальна інфраструктура залишатиметься функціональною. Оскільки все повністю інтегровано в гіперконвергентну інфраструктуру, то не можна відокремити жоден компонент один від одного. Конвергентна інфраструктура більш орієнтована на апаратне забезпечення, тоді як гіперконвергентна є більш програмно орієнтованою.

1.6 Висновки до розділу 1

Створення систем функціонування, які ґрунтуються на понятті ситуації і визначаються ситуаційними факторами, потребує загальносистемного аналізу ситуаційної діяльності та особливостей різних типів ситуаційних систем. Розгляд існуючих класифікацій ситуаційних систем показує, що спосіб побудови і організації такої системи визначається цільовим призначенням і кінцевим продуктом функціонування такої системи. Виходячи з цього факту, була запропонована узагальнена класифікація ситуаційних систем, серед яких цільовим призначенням систем ситуаційного управління є аналіз впливу поточної ситуації на стан цільової предметної області управління такої системи, вироблення та реалізація керуючих впливів на цільову предметну область управління з метою утримання її параметрів у заданих межах. Принциповою відмінністю ССУ від систем автоматичного та автоматизованого управління є часткова невизначеність як самих ситуацій, так і їх впливу на цільову область або об'єкт.

Аналіз моделей ситуаційного управління та ситуаційно-орієнтованої поведінки показав, що процеси ситуаційного управління реалізуються в рамках перцептивного циклу, тобто мають циклічний характер і включають етапи емпіричного усвідомлення стану середовища (цільової області), побудови його моделі та застосування цієї моделі при формуванні раціональної поведінки у

середовищі на основі періодичного оновлення усвідомлення поточного стану середовища. Усвідомлення стану середовища та формування на його основі раціональної поведінки здійснюються з використанням механізмів логічного умовиводу, що відповідають етапам перцептивного циклу. Такими механізмами логічного умовиводу у циклі ситуаційної взаємодії з середовищем є індукція, дедукція та абдукція.

Для побудови ССУ застосовуються методи системної інженерії та інженерії системи систем. Для реалізації системних принципів побудови ССУ слід застосовувати конвергентний підхід, який надає можливості поєднання у цільовій системі або у системі систем потрібних компонентів та їх гнучкої адаптації до задач ситуаційного управління.

Основні результати розділу опубліковані автором у працях [1, 6, 7, 13, 18, 25, 28, 40, 47, 55, 78].

РОЗДІЛ 2

АСПЕКТИ ПОБУДОВИ КОНВЕРГЕНТНИХ СИСТЕМ СИТУАЦІЙНОГО УПРАВЛІННЯ

2.1 Динамічні моделі ситуаційного управління

Процеси конвергенції базуються на корисних спроможностях та принципах, притаманних іншим технологіям. Процес ситуаційного управління (СУ) можна розглядати як ланцюжок зростання цінності інформації, в процесі якого формуються рішення щодо управління ситуацією. Прийняті рішення по суті складають інформаційний продукт. Підтримка СУ засобами комп'ютерних технологій надає додаткову цінність інформації на різних фазах та етапах СУ за рахунок спроможностей формування і розвитку формалізованих знань предметної області управління з можливостями їх повторного використання. Взагалі процес СУ можна охарактеризувати як процес перетворення та збільшення цінності інформації на основі конвергенції компонентів СУ засобами комп'ютеризації.

Відповідно до процесної моделі вхідна інформація під час СУ перетворюється у вихідну інформацію із відповідним контекстом під впливом керуючої інформації з використанням наявних ресурсів.

Необхідність використання в ситуаційному управлінні засобів комп'ютерної техніки ставить задачу конвергенції різних кібернетичних підходів та інформаційних технологій для реалізації процесів і процедур ефективного управління у складних інформаційних організаційно-технічних системах (соціальних, політичних, економічних, технічних тощо).

Процес ситуаційного управління пов'язаний із функціональним перетворенням (трансформацією) інформації з первинного представлення [186], [187], що поступає на вхід до системи ситуаційного управління (ССУ), до представлення, що відображає результати управлінської діяльності у процесі СУ. Узагальнена функціональна модель процесу трансформації інформації у ССУ наведена на рис. 2.1 [8].

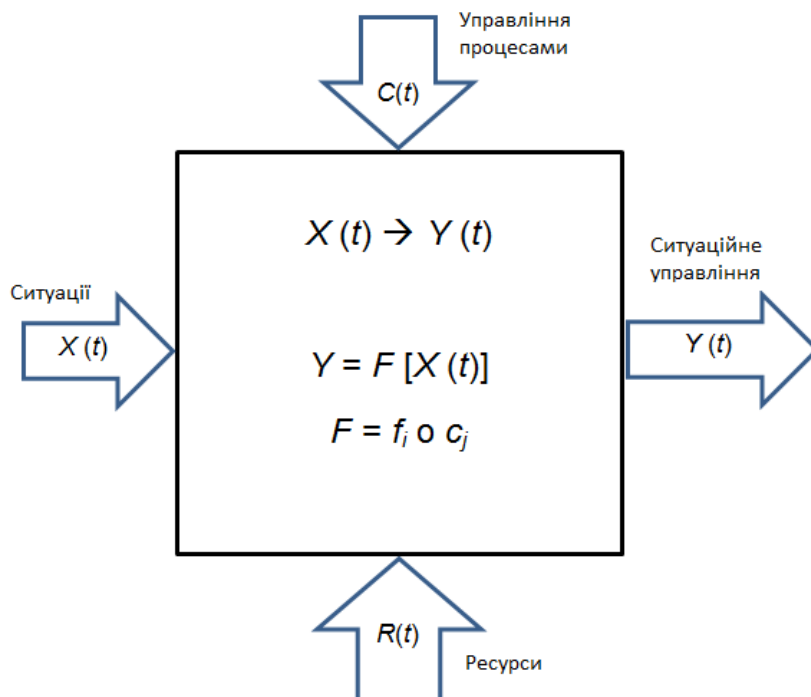


Рисунок 2.1 – Узагальнена модель процесу трансформації інформації в ССУ

Вхідна формація $X(t)$ про ситуацію надходить із цільової області як множина сигналів та/або сигнальних патернів (знаків, символів, лексем тощо). Вхідна інформація $X(t)$ перетворюється на вихідну інформацію управління ситуацією $Y(t)$ у вигляді інформаційних об'єктів описового або директивного характеру, під керуванням імперативної інформації $C(t)$, що визначається нормативами, регламентами та правилами перетворень із урахуванням обмеженості ресурсів $R(t)$. Отже, $Y(t)$ є результатом функціонального перетворення F вхідної інформації $X(t)$:

$$Y(t) = F[X(t)]. \quad (2.1)$$

Функція перетворення F – це складена (комполитна) функція множини функціональних композицій $\{g_i \circ c_j \mid g_i \in G\}$ функцій перетворення інформації та

$c_j \in C$ функцій управління трансформацією за умов обмеженості ресурсів $r_k \in R$. Функція F може задана як об'єднання композицій чи іншим способом.

Інформація про набори функцій G , C і R представляється піддоменами знань цільової області знань. Інший піддомен цільової галузі знань – це знання про сценарії та процедури процесів СУ. Конвергенція різних засобів та інструментів комп'ютерних систем визначається та відбувається шляхом реалізації відповідного сценарію в ході СУ.

Формальну динамічну модель функціональної трансформації інформації в ССУ можна представити у вигляді [2]

$$Y(t) = P(C(F(R(t), X(t)), F(R(t), X(t))), \quad (2.2)$$

де Y – вихідна керуюча інформація СУ (результат функціонування ССУ), P – функція управління (організації) процесів у системі, C – функція управління обробкою інформації в системі, F – функція обробки інформації СУ, R – функція варіації (змін) із часом ресурсів, X – вхідна ситуаційна інформація, t – часова змінна з області визначення функції P (період життєвого циклу процесу СУ).

Функціональна трансформація інформації здійснюється шляхом реалізації процесів СУ в ССУ через взаємодію з цільовою сферою ССУ. Узагальнена модель взаємодії ССУ з цільовою сферою СУ показана на рис. 2.2 [7].



Рисунок 2.2 – Узагальнена модель ССУ

Процес СУ S1 (рис. 2.2) реалізується як різновид проєктної діяльності, що включає такі підпроцеси:

- усвідомлення ситуації;
- розпізнавання та оцінки ситуації;
- планування заходів СУ на основі визначення цілей управління та вироблення варіантів дій проведення управління;
- прийняття рішень щодо вибору варіантів сценаріїв СУ;
- впровадження прийнятих рішень шляхом проведення операцій СУ;
- моніторинг впровадження рішень;
- збереження результатів СУ;
- аналіз та оцінки результатів СУ.

Структура підпроцесів процесу СУ зображена на рис. 2.3 [7].

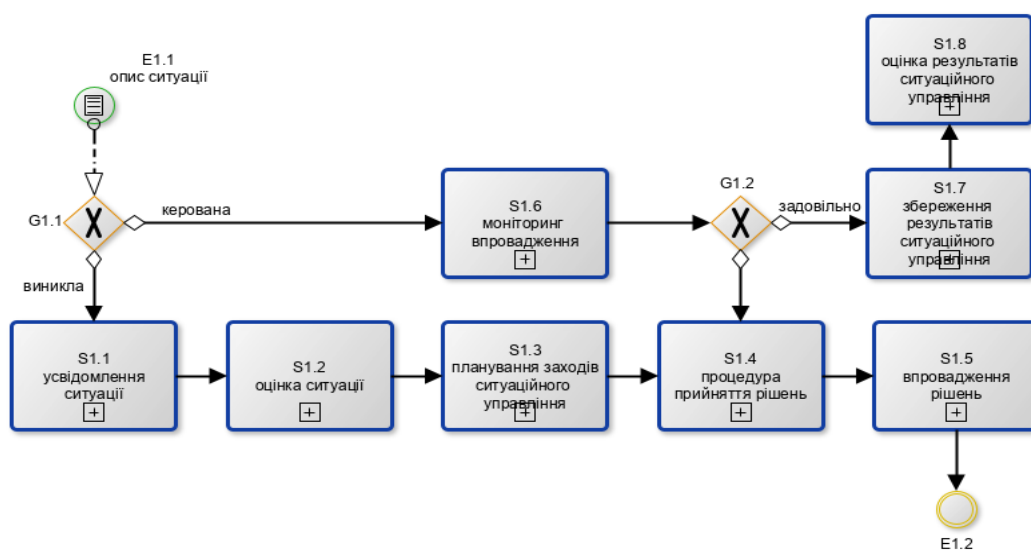


Рисунок 2.3 – Структура підпроцесів СУ

Для реалізації кожного з підпроцесів необхідно формувати відповідний набір засобів щодо реалізації та забезпечити їх конвергенцію до завершення підпроцесу, з яким вони пов'язані. Ситуаційне управління як проєктна діяльність пов'язане з досягненням цілей, необхідних для встановлення визначеного стану керованої системи, виходячи з поточної ситуації. Складові процесів ситуаційного управління як проєктної діяльності представлені на рис. 2.4.



Рисунок 2.4 – Процеси ситуаційного управління як проектної діяльності

2.2 Трансформація інформації в системах ситуаційного управління

Цілеспрямовані системи, зокрема, ССУ, є найпоширенішим класом динамічних систем, які характеризуються певним рівнем автономності. Моделі і математичний апарат для опису функціонування цілеспрямованих систем представлені, зокрема, в роботах [188], [189]. Процес цілеспрямованої діяльності у спрощеному вигляді може бути описаний такою моделлю [56]: визначення цілі – оцінка (диференціація) середовища – мотивація досягнення цілі – визначення доступних ресурсів – визначення підцілей – визначення плану (шаблону) досягнення цілі – виконання плану – виявлення відхилень – коригування плану – оцінка результату діяльності. Кожен із цих етапів може бути представлений різними способами, але головною вимогою досягнення цілі є здійснення цілеспрямованої діяльності у визначеному контексті. Втрата контексту (змісту, семантики) означає втрату цілі й метрики для оцінки її досягнення.

Створення цілеспрямованих систем, заснованих на знаннях, вимагає врахування особливостей (контексту) предметної сфери, в якій застосовується система. Засновані на знаннях системи поєднують в собі знаковий (семіотичний), синтаксичний, семантичний, структурний аспекти [57].

Знак. Для узагальнення поняття знаку в ієрархії об'єктів предметної сфери можна скористатись положеннями загальної теорії патернів [190]. Тоді, виходячи з логіко-філософської інтерпретації поняття знаку (патерну), розширення семіотичного трикутника Г. Фреге [191] (рис. 2.5 а) до семіотичної піраміди [192] (рис. 2.5 б) забезпечує об'єктивізацію знака в межах багатозначного інформаційного середовища.

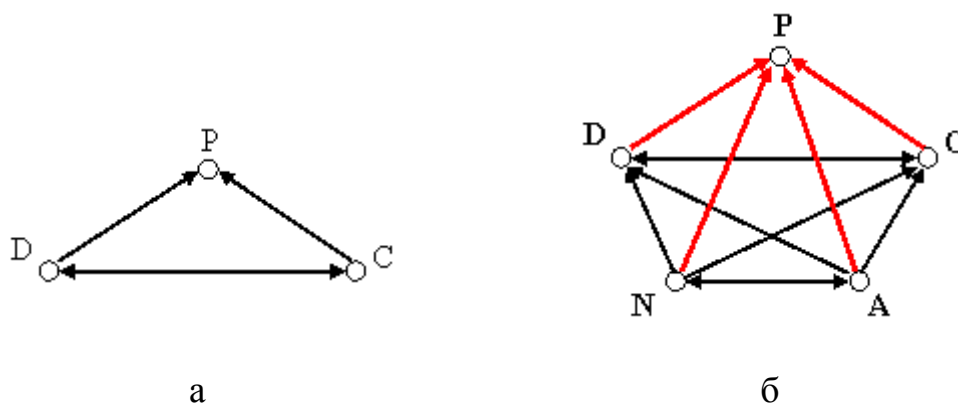


Рисунок 2.5 – Семіотичні графи

Вершина семіотичної піраміди (рис. 2.5 б) представлена патерном Р – диференційованим об'єктом середовища (предметного домену), що пов'язаний із компонентами, які його визначають і розташовані в основі семіотичної піраміди. D – позначення (пряма семантика, використовується для позначення множини об'єктів) – денотат патерну Р. С – контекстна семантика (конкретизований зміст) – концепт патерну Р. N – ім'я патерну в конкретному контексті. А – семантичне посилання на патерн, що визначає місце патерну в ієрархії семантичних доменів.

Патерни. В загальній теорії патернів У. Гренандера [190] використовуються об'єкти або патерни чотирьох рівнів. Перший рівень складають об'єкти, які називаються твірними (generators). Вони моделюють відносно прості об'єкти

реального світу, що мають зв'язки, які можуть з'єднуватись із зв'язками інших об'єктів.

Із твірних, шляхом об'єднання їх зв'язків (bonds) конструюються об'єкти другого рівня – регулярні конфігурації (configurations).

Об'єктами третього рівня є зображення (images), отримані таким чином: подібні між собою регулярні конфігурації об'єднуються в множину, на якій визначаються класи еквівалентності. Відношення еквівалентності, яке визначає клас еквівалентності на множині регулярних конфігурацій, називається зображенням.

Четвертий рівень об'єктів складають власне патерни-образи (patterns). Патерном-образом називається множина зображень, інваріантних відносно перетворень подібності. Якщо на деякій множині регулярних конфігурацій існує лише одне зображення, то воно являє собою єдиний образ цієї множини.

Твірні, регулярні конфігурації, зображення і образи є відповідно патернами першого, другого, третього і четвертого рівнів. Формальні методи теорії патернів, що застосовуються на кожному з чотирьох рівнів, відрізняються високим ступенем гнучкості й дозволяють моделювати зв'язки, об'єднання і перетворення подібності логічних об'єктів реального світу. Шляхом групування патернів-образів одного семантичного рівня можна створювати ієрархічні семантичні системи довільної складності. Для патернів-образів кожного наступного рівня таких систем як твірні використовуються патерни нижчих рівнів. Патернам присвоюються імена. Наприклад, патерн-твірна алфавітно-цифрового знаку отримує відповідне ім'я та інші атрибути семіотичної піраміди.

Метафора. Метафора є одним із головних інструментів пізнання і опису реального світу, застосовується для компактного представлення понять і положень у різних галузях знань. Метафорі приписується певна семантика, що визначається явно або виводиться із контексту середовища (оточуючих об'єктів) [193]. Метафора є асоціативним образом (патерном), що утворюється на основі подібності різних властивостей об'єктів. Існують різні класифікації метафори, зокрема, Дж. Лакофф і М. Джонсон вирізняють два типи просторово-часових метафор [194]:

- 1) онтологічні, які описують семантичні ієрархії;

2) орієнтаційні, які забезпечують організацію об'єктів.

На основі метафори утворюються омонімічні ряди. Конкретизація метафори у контексті предметної ситуації створює патерн знання, що визначає зміст (семантику) цілеспрямованої діяльності.

Знання. Саме поняття «знання» є багатозначним, метафоричним. Семантика патерну «знання» виводиться з його контексту, тобто з активних зв'язків із іншими патернами середовища. Представлення об'єкта-патерну знань у вигляді послідовності об'єктів-символів є просторовою згортою часової послідовності відтворення абстрактного подання реальних об'єктів середовища.

Динаміка. Поведінку цілеспрямованої системи можна представити як динамічний процес, що описується векторним полем у фазовому просторі. Кожна точка такого фазового простору задає певний стан системи. Вектор окремої точки фазового простору визначає швидкість зміни стану системи. Нульове значення вектора зміни стану системи визначає положення рівноваги, яке може співпадати з тупиковим станом або станом досягнення цілі.

Поведінка є послідовністю актів цілеспрямованої діяльності, кожен із яких є синтетичним результатом поєднання накопичених знань, результату оцінки стану оточуючого середовища, прогнозу наступного стану, цілей і підцілей динамічної системи. Таким чином, поведінку можна описувати, спираючись на положення теорії катастроф, із урахуванням положень теорії біфуркацій динамічних систем і теорії особливостей гладких відображень. Знання є певною мірою статичним компонентом у кожному акті цілеспрямованої діяльності. Але за результатами кожного акту окремий патерн знань може бути змінений (реконфігурований) або залишитись незмінним.

Планування. Одним із головних етапів цілеспрямованої діяльності є планування – створення патерну-шаблону траєкторії зміни станів у фазовому просторі. Кожний шаблон поведінки представляється або стандартним патерном-шаблоном (інваріантом патерна, що відповідає певній проблемній ситуації), або новим патерном-шаблоном, адекватність якого потрібно підтвердити. При плануванні цілеспрямованої діяльності використовуються індуктивні (синтетичні,

висхідні), дедуктивні (аналітичні, низхідні), абдуктивні (горизонтальні) логічні зв'язки між патернами.

Доцільність. Доцільність впливає із моделі цілеспрямованої діяльності з урахуванням контексту середовища і параметрів (умов) здійснення діяльності (цілей, мотивацій, ресурсів та ін.). Для оцінки рівня досягнення підцілей і головної цілі, у залежності від контексту, використовуються метрики, що визначаються контекстом середовища функціонування системи.

Інформація – це широке поняття, яке стосується різних аспектів сприйняття та представлення навколишнього середовища в процесах цілеспрямованої діяльності. Наприклад, словник Merriam-Webster дає чотири різні визначення інформації [195]. Феномен інформації з семантичної точки зору досліджував Лучіано Флоріді [196]. У словнику інформаційних технологій [197] подано два визначення інформації:

1) щодо обробки інформації – це знання про об'єкти, а саме факти, події, речі, процеси та ідеї, у тому числі поняття, які в певному контексті мають особливе значення;

2) щодо теорії інформації – це знання, яке зменшує або усуває невизначеність про виникнення конкретної події з заданого набору можливих подій.

Проблема створення спільного інформаційного середовища пов'язана з необхідністю створення єдиної інформаційної моделі. За визначеннями чітко виділяються різні види інформації. Зазвичай розмежування між ними ґрунтується на атрибутах середовища передачі інформації, її структурі, способі сприйняття, засобах обробки, зберігання, стані та інтерпретації (за призначенням та семантикою).

Джерелом інформації є подія або явище, яке генерує сигнал (за відсутністю сигналу інформація відсутня). Сигнал реєструється відповідними сенсорами (датчиками) і далі може бути поданий у вигляді структурованого набору даних. Після цього дані про сигнал інтерпретуються відповідно до області знань їх застосування з використанням моделей інтерпретації знань для збагачення та генерації нових знань. Далі знання перетворюються на мудрість (інтелект) у поєднанні з індивідуальністю їх носія. Притаманна людям мудрість передається

доступним способом послідовникам. Так схематично можна подати трансформацію інформації.

Під час трансформації інформація проходить такі етапи: сигнал (подія) → дані (структура) → знання (усвідомлення, розпізнавання) → знання (розуміння) → рішення (мудрість). Перший етап має природне походження, а інші є штучними когнітивними (пізнавальними) конструкціями.

Для визначення і конкретизації процедур (видів діяльності) у процесах трансформації інформації необхідно побудувати таксономію та онтологію поняття (концепту) інформація.

Значення поняття "інформація" змінюється у різних філософських школах, його прагматичний зміст залежить від місця та різних прагматичних контекстів. Одна з найпопулярніших ієрархічних моделей трансформації інформації була введена Расселом Лінкольном Акоффом і об'єднала концепції даних-інформації-знань-мудрості в єдину формулу DIKW (Data-Information-Knowledge-Wisdom) [198]. Ця модель була корисною на певному періоді з точки зору прагматики, але не була математично коректною та строгою.

У даній роботі пропонуються онтологія та модель трансформації інформації для основних понять інформатики. Пропонована категорійна модель інформації I-СДЗМ (інформація як сигнали-дані-знання-мудрість), що відповідає англійському акроніму I-SDKW (Information as Signal, Data, Knowledge, Wisdom (Rationality, Intelligence)), [25] базується на семантичному змісті використання інформації під час прийняття рішень взагалі та ситуаційному управлінні зокрема. Приймемо деякі твердження.

Твердження 1. Усі види сприйняття – це сприйняття інформації.

Твердження 2. Результатом обробки сприйнятої інформації є інформація.

Твердження 3. Інформація про об'єкти (сутності) та їх властивості відноситься до однієї з категорій: сигнал, дані, знання, мудрість (інтелект).

Відмінності між категоріями полягають у способі їх використання: сигналів для сприйняття та впливу, даних для структурованого запам'ятовування та

зберігання, знання для розпізнавання та розуміння, а мудрість для прийняття рішень та мотивації дій.

У роботі [199] виділено чотири кількісні та дві якісні формалізації поняття інформації.

Кількісні поняття:

1. Інформація за концепцією Фішера.
2. Інформація за концепцією Шеннона.
3. Інформація за концепцією складності Колмогорова.
4. Інформація за квантовою концепцією (кубіт).

Якісні поняття:

5. Інформація про стан агента.
6. Семантична інформація.

Інформація за концепцією Фішера визначається кількістю інформації, що несе спостережувана випадкова величина про невідомий параметр, від якого залежить імовірність спостережуваної випадкової величини.

Інформація за концепцією Шеннона обчислюється як ентропія дискретної випадкової величини, що вимірюється рівнем невизначеності, пов'язаної зі значенням цієї дискретної випадкової величини.

Інформація за концепцією складності Колмогорова обчислюється як довжина найкоротшої програми, що видає двійкові рядки на еталонній універсальній машині Тьюрінга.

Інформація за квантовою концепцією є узагальненням класичного біта і описується квантовим станом у двостановій квантово-механічній системі, формально еквівалентній двовимірному векторному простору комплексних чисел. Чистий стан кубіта – це лінійна суперпозиція базисних станів 0 та 1.

Інформація про стан агента являє собою оцінку стану цільової предметної області через формальне логічне обґрунтування понять знань та переконань у контексті теорії інформації, результатів анкетування або поширення загальних повідомлень.

Семантична інформація визначається за множиною критеріїв структурованості, змістовності та правдивості.

Компонування та узгоджене використання різних понять у практичній діяльності у відповідному контексті можуть бути здійснені за допомогою адекватної онтології інформації. Різні концепції можуть бути реалізовані через онтологію. У роботі [200] запропоновано п'ять різних онтологій, пов'язаних зі словосполученням «онтологія інформації» (information ontology). Найбільш загальною з них є онтологія артефактів інформації (Information Artifact Ontology) [201]. Інші онтології мають окрему спеціалізовану семантику.

Інформація є ключовим елементом прийняття рішень і використовується на кожному етапі циклу прийняття рішень у процесі ситуаційного управління. Існують різні цикли прийняття рішень, але кожен із них являє собою послідовність етапів, що використовуються суб'єктом прийняття рішень на повторній основі, для передачі та реалізації рішень і подальшого аналізу їх результатів [36], [202]. Для практичного прийняття рішень необхідна узгоджена онтологія [203] інформації для використання цього концепту в адекватному контексті відповідно до ситуації. Узгоджена онтологія інформації може бути розроблена шляхом композиції категорій інформації з різним контекстом і спільним концептом верхнього рівня. Концепти нижніх рівнів повинні відображати специфіку конкретного прояву (категорії) інформації. Відповідно до твердження 3 виділяються чотири категорії інформації: сигнал, дані, знання та мудрість. Особливості цих категорій за атрибутами, способом використання та метою обробки представлені в табл. 2.1. [25].

Пропонована онтологічна модель інформації I-SDKW включає в себе онтології сигналів (сприйняття символів, знаків), даних (отримання та структурування сигналів), знань (інформованість, усвідомлення, розуміння, використання даних) і мудрості (прийняття рішень, розумна поведінка, інтелект), що охоплюють різні цикли мислення, такі як навчання, інтелектуальний аналіз та цикли прийняття рішень.

Таблиця 2.1 – Характеристики категорій інформації

Категорії інформації	Властивості категорії		
	Атрибути	Використання	Фокус оцінювання
Сигнал	Джерело, середовище, потужність, час	Передача, приймання, комунікації	Атрибути
Дані	Об'єм, тип	Запам'ятовування і зберігання	Структура
Знання	Семантична повнота	Розуміння, інтерпретація, мотивування	Логічна модель і виразність
Мудрість	Ефективність, продуктивність, корисність, практичність	Прийняття рішень, раціональна поведінка	Практичність, результативність, цілеспрямованість

Таким чином, інформація трансформується через отримання, збір, агрегування, фільтрацію, представлення, усвідомлення, тлумачення, оцінювання, використання, збагачення, компонування та передачу на вищі рівні застосування. Таксономія категорій інформації показана на рис. 2.6 [25].

Модель трансформації I-SDKW включає різні етапи представлення й використання інформації. Інформація як сигнали, пов'язані з первинним сприйняттям послідовностей, символів і знаків за допомогою сенсорних систем (природних або штучних). Інформація як дані, пов'язані із придбанням, реєстрацією та структуруванням сигналів. Інформація як знання, пов'язане зі знанням, усвідомленням, розумінням і використанням даних. Інформація як мудрість охоплює різні цикли мислення, такі як навчання, розум і цикли прийняття рішень. У цій моделі інформація з'являється і перетворюється в різних проявах. Таким чином, інформація проявляється або сигналом, або структурою, або теорією (системою), або рекомендаціями, або кращими практиками тощо.

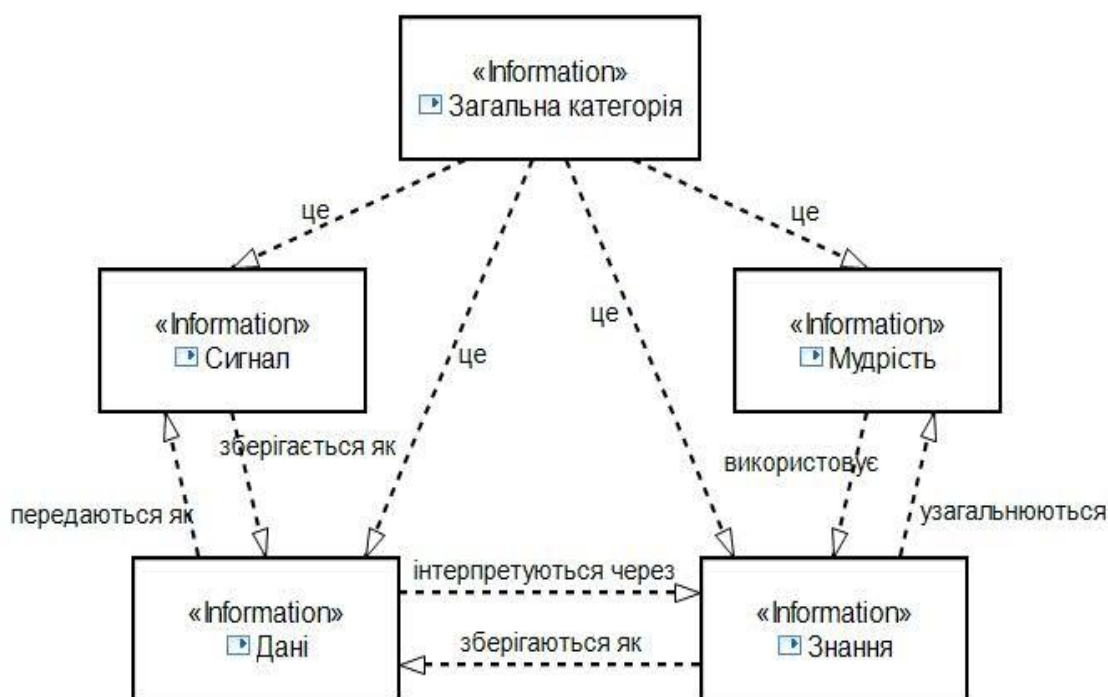


Рисунок 2.6 – Відношення між категоріями інформації

Онтологія є результатом систематизації тезауруса цільового домену знань. Інформація про проблемну область структурується з сигналів (послідовностей, символів, знаків і значків) до структур даних, далі до контрольованих словників, потім до таксономій, після цього до тезауруса і, нарешті, до онтології. Тому можливо побудувати перцептивну ієрархію від сигналу до мудрості і ефекторної ієрархії від мудрості до сигналу. Фрагмент верхнього рівня онтології I-SDKW зображений на рис. 2.7 [25].

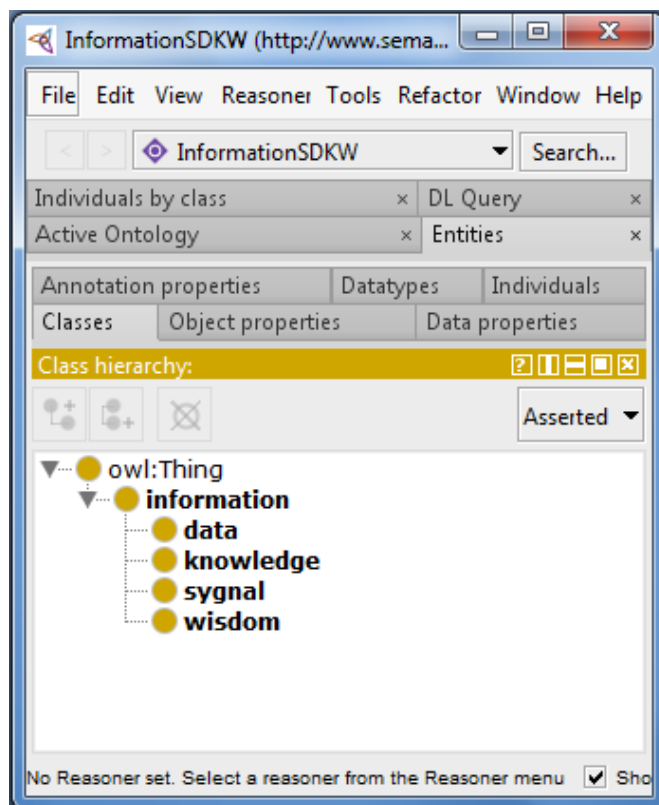


Рисунок 2.7 – Фрагмент онтології верхнього рівня для моделі I-SDKW

Онтологія сигналу. Сигнал у різних областях, таких як системи зв'язку, обробка сигналів та електротехніка, «передає інформацію про поведінку або атрибути якихось явищ» [204]. Це визначення сигналу відрізняється від визначення сигналу як варіації фізичної величини, що використовується для подання даних у [198]. Можна сказати, що сигнали також містять інформацію про дані, знання та мудрість.

Атрибутами сигналу є джерело, середовище поширення, спектр, спосіб реєстрації, потужність, рівень, час, спосіб подання. Ці атрибути дозволяють визначити сигнал для його подальшого перетворення, зберігання та інтерпретації. Сигнал в інтерпретації семіотичної теорії Пірса асоціюється зі знаком та його категоріями. Типи знаків визначаються теорією знаків Пірса [205]. Сигнал є відображенням події, пов'язаної зі зміною стану об'єкта, що спостерігається та розпізнається сенсорами системи приймання сигналу. Отже, подія, яка не розпізнається сенсорами, не є сигналом. Сигнал є первинною формою обміну інформацією. Як первинна інформація про зміни у стані об'єкта спостереження

сигнал описується функцією часу $s(t)$, що відображає параметри цих змін у часовій області. На основі первинного представлення сигналу у часовій області можуть бути отримані спектральні та інші атрибути (характеристики) сигналу.

Онтологія даних. Дані – це повторно інтерпретоване представлення інформації певним формалізованим способом, придатним для комунікацій, інтерпретації та обробки людиною або автоматичними засобами [206], тобто, дані є інформацією. Опис даних формується на різних рівнях: атомарні дані – структури даних – тип даних GPD – визначений тип даних (конкретний, абстрактний, алгебраїчний). Вичерпну прагматичну онтологію типів даних загального призначення відповідно до стандарту ISO про типи даних описано у [207].

При прийнятті рішень у процесах ситуаційного управління важливо мати додаткову інформацію, що описує контекст даних, зокрема, їх джерело, місце зберігання та спосіб використання. Дані характеризуються множиною допустимих операцій, що позначається акронімом CRUD (create, read, update, delete). Отже, формально дані є відображенням F_d сигналу $s(t)$ у структуру даних D зі структурними елементами даних E і відношеннями R між ними: $F_d : s(t) \rightarrow D = \langle E, R \rangle$. Зворотнє перетворення формує сигнал на основі даних $F_d^{-1} : D \rightarrow s(t)$.

Онтологія знань. Знання як феномен інформації складають різні дані із правилами та обмеженнями стосовно способів їх обробки та інтерпретації. Автор визначає знання як обізнаність, усвідомлення, розуміння, вміння та здатності людей або машин, пов'язані з реальними або абстрактними об'єктами та предметами середовища, отримані та систематизовані шляхом обробки інформації. Отже, знання носять загальний характер.

Існують різні визначення знань [208], [209]. Як результат трансформації інформації можна дати таке визначення поняття «знання».

Знання – це обізнаність, усвідомлення, розуміння, вміння та навички суб'єктів (природних, соціальних, штучних агентів) про реальні чи абстрактні об'єкти та суб'єкти середовища, здобуті й систематизовані через обробку інформації [25]. Під

суб'єктами (акторами) будемо розуміти автономні проактивні сутності середовища (предметної області), що характеризуються цілеспрямованою раціональною поведінкою на основі моделі світу. Проактивність передбачає не лише просте реагування на зміни середовища, але й формування цілеспрямованої поведінки і здатність проявляти ініціативу

Слід відрізнити знання та розуміння. Розуміння – це володіння певною інформацією в різних формах представлення про об'єкти та суб'єкти середовища й відношення між ними. Знання – це структурована та систематизована інформація з певних предметних областей (доменів), представлена з використанням відповідних засобів і (або) інструментів маніпуляції, спрямованих на отримання нової інформації про ці предметні області. Засоби та інструменти маніпулювання знаннями можуть бути реалізовані як формальні абстрактні логічні теорії і технології на їх основі.

Одним із комплексних сучасних формальних методів представлення і маніпулювання знаннями є описові (дескрипційні) логіки DL [210]. База знань відповідно до формалізму дескрипційних логік – це впорядкована пара $K=(T, A)$, до якої належать множини $TBox$ T і $ABox$ A . Множина $TBox$ є множиною термінологічних аксіом бази знань, а множина $ABox$ є множиною тверджень та фактів бази знань, тобто, тверджень про індивідів.

В окрему множину аксіом можна виділити ієрархію ролей $RBox$. Конкретна описова логіка має набір конструкторів та індуктивного правила, за допомогою якого складові концепти даної логіки будуються з атомарних концептів і атомарних ролей. Дескрипційні (описові) логіки пов'язані з модальними логіками (ML), але розроблені незалежно одна від одної. Деякі варіанти описової логіки є синтаксичними варіантами ML. Загалом об'єкти DL відповідають можливим світам ML, концепти DL відповідають модальним пропозиціям (твердженням) ML, обмежений роллю квантифікатор у DL – модальному оператору в ML із такою ж роллю, як доступні йому відношення.

Знання можуть бути явними, неявними або уявними (ментальними). Ключовим механізмом розширення знань є розумова діяльність або процеси

отримання знань та розуміння шляхом мислення, досвіду та відчуттів, що складають сутність когнітивних процесів.

Знання як категорію інформації складають структури даних D із заданими на них відношеннями M та функціями інтерпретації I , які визначають правила та обмеження їх обробки: $K = F_k(D, M, I)$. Тобто, база знань має інтерпретацію I (моделює предметну область): $I \models K \text{ iff } I \models A \ \& \ I \models T$. Структури даних D складають концепти та факти баз знань, відношення M дозволяють побудувати твердження на їх основі, а функції інтерпретації I побудувати логічні висновки на основі тверджень і фактів. Отже, формально знання є відображенням F_k структур даних D , що представляють концепти предметної області, у модель знань K із заданими відношеннями M та функціями інтерпретації I , тобто, $F_k : D \rightarrow K = \langle D, M, I \rangle$. Зворотне перетворення формує набори даних на основі знань $F_k^{-1} : K \rightarrow D$.

Онтологія мудрості. Розуміння концепту мудрості є філософською проблемою від античності до сучасності й стосується людської діяльності в різних сферах. Мудрість (інтелект) – це складна соціологічна і пізнавальна конструкція, заснована на п'яти принципах [211]:

- мудрість ґрунтується на мисленні та спостереженні;
- мудрість включає нерациональні та суб'єктивні елементи у судження;
- мудрість спрямована на справжні гуманістичні та добродесні наслідки;
- мудрість є виразною, естетичною та внутрішньо корисною;
- мудрість практична (повинна бути корисна для чогось).

Таким чином, мудрість індивідуальна і має ситуативний характер. У статті "Wisdom" Стенфордської енциклопедії філософії [212] описані різні теорії мудрості, корисні в різних ситуаціях. Ці теорії ґрунтуються на підходах оснований на гносеологічній стислості, епістемічній точності, використанні знань, раціональності, гібридності. Теорія мудрості як раціональна поведінка уникає серйозних проблем інших теорій і трактує мудрість як глибокий і всебічний різновид раціональності. Теорія глибокої раціональності (DRT) базується на чотирьох умовах [212]

«мудрості» джерела (носія) інформації S . Мудрим (корисним) джерело S вважається лише тоді, якщо (iff):

S має широкий спектр епістемічно обґрунтованих знань (переконань) на широке коло значимих наукових проблем;

S має широке розмаїття обґрунтованих знань (переконань) про те, як діяти і жити раціонально (епістемічно, морально і практично).

S прагне діяти і жити раціонально;

S має невеликий об'єм невиправданих знань (переконань) і чутливе до їх обмежень.

У такій інтерпретації мудрості переконання, по суті, є глибоко усвідомленими знаннями, які визначають ментальність (ментальну модель) суб'єкта.

Ступінь (міра) раціональності (мудрості) поведінки суб'єкта формально може бути оцінена з використанням теорії корисності [213], [214] або теорії перспектив (проспектів) [215], [216]. Зокрема, основне спостереження кумулятивної (накопичувальної) теорії перспектив (Cumulative Prospect Theory) і базової теорії перспектив полягає в тому, що суб'єкти зазвичай оцінюють можливі результати відносно певної еталонної точки (певного статус-кво), а не відносно кінцевого стану, що називають ефектом обрамлення (фреймінгом). Більше того, вони мають різне ставлення стосовно ризику до переваг (тобто, результатів вище еталона) та втрат (тобто, результатів нижче еталона) і в основному більше піклуються про потенційні втрати, ніж про потенційні переваги (відмова від втрат). Крім того, суб'єкти схильні переоцінювати екстремальні, малоймовірні події, але недооцінюють «середні» події. Це положення, на відміну від базової теорії перспектив, передбачає, що суб'єкти надмірно переоцінюють малоймовірні події незалежно від їх відносних результатів.

Кумулятивна теорія перспектив включає ці спостереження в модифіковану теорію очікуваної корисності через заміну кінцевої переваги перевагою відносно еталонної (опорної) точки, заміну функції корисності функцією значимості, що залежить від відносної переваги, і заміну кумулятивних імовірностей зваженими кумулятивними ймовірностями. У загальному випадку це призводить до наступної

формалізації суб'єктивної корисності ризикованого результату $U(p)$, описаного мірою ймовірності p :

$$U(p) := \int_{-\infty}^0 v(x) \frac{d}{dx} (w(F(x))) dx + \int_0^{+\infty} v(x) \frac{d}{dx} (-w(1-F(x))) dx, \quad (2.3)$$

де v – функція значимості, w – вагова функція, $F(x) := \int_{-\infty}^x dp$ – інтеграл вимірювання ймовірності для всіх значень x , тобто кумулятивна ймовірність. Це узагальнює первинне формулювання [182] від скінченної множини окремих результатів до нескінченних (тобто, безперервних) результатів.

Таким чином, мудрість B є відображенням моделі знань K , доповненої функціями v та w , що визначають міру корисності U знань про варіанти дій у тій чи іншій ситуації: $F_b : K \rightarrow B = \langle K, U \rangle$. Зворотне перетворення формує фрагменти знань (логічні твердження) на основі запропонованих раціональних (мудрих) рішень $F_b^{-1} : B \rightarrow K$.

Трансформація інформації, що стосується різних концептів, полягає в тому, що інформація сприймається як сигнали, структурується і зберігається як дані, інтерпретується як знання і використовується як мудрість. Описана онтологія перетворення інформації охоплює всі стадії трансформації інформації – від сприйняття сигналу через бітові структури до раціональних міркувань (тверджень), корисних для ситуаційного управління.

Перетворення інформації пов'язане з двонаправленим перетворенням різних видів інформації у процесі її використання. Сприйняття інформації починається саме з отриманням сигналу від джерела з навколишнього середовища, оскільки відсутність сигналу означає також відсутність інформації про джерело («неможливо в темній кімнаті знайти чорного кота, якого там немає»). Таким чином, сигнал про явище несе первинну інформацію про це явище у формі символу, знаку, образу або динамічного процесу і перетворюється у відповідні структури даних. Ці структури

даних можуть бути скомпоновані за значеннями (контекстом) у вигляді набору термінологічних аксіом бази знань (TBox) і набору тверджень про індивідів (асерційної частини) бази знань (ABox) [211] деякої предметної області та зв'язаний з ними набір фактів, який формується базою знань. Проблема полягає в тому, що і як спостерігачі можуть щось дізнатись про цільову область знань.

Модель І-СДЗМ (I-SDKW) надає підхід для вирішення цієї проблеми. Фрагмент верхнього рівня онтології І-СДЗМ (I-SDKW) у форматі OntoGraf зображений на рис. 2.8 [6], [25]. У цій моделі інформація з'являється і перетворюється в різних своїх проявах. Тобто, інформація проявляється або сигналом, або структурою, або теорією (системою), або рекомендаціями, або найкращою практикою тощо. Але самого знання недостатньо для прийняття мудрого (корисного) рішення. Мудрість полягає в тому, що рішення має відповідати контексту ситуації.

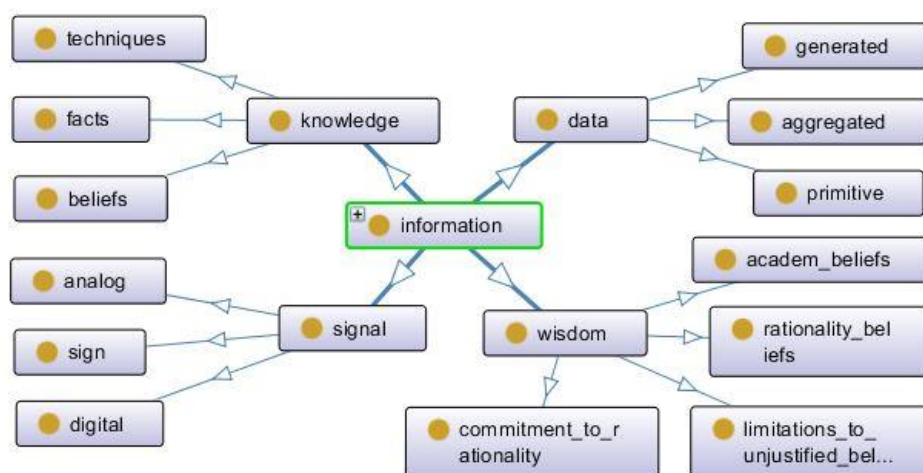


Рисунок 2.8 – Фрагмент онтології I-SDKW із атрибутивними концептами

Онтологія є результатом систематизації тезаурусу проблемної області знань. Інформація про проблемну область структурується шляхом трансформації інформації, починаючи з реєстрації сигналів (послідовностей, символів, знаків та символів) з наступним формуванням відповідних структур даних, подальшою побудовою керованих словників, їх систематизацією і, врешті, створенням тезаурусу та онтологій [19], [25]. На основі сформованих структур можна побудувати

перцептивну модель із прямою трансформацією інформації від сигналу до мудрості та зворотною трансформацією від мудрості до сигналу.

Модель трансформації інформації I-SDKW базована на семантичних аспектах використання інформації в ситуаційних системах у процесах різних видів ситуаційного управління. Пропонована парадигма трансформації інформації I-СДЗМ (I-SDKW) враховує атрибутивні властивості інформації та способи як її представлення, так і використання. Це особливо актуально для обробки інформації під час ситуаційного управління, коли ситуаційна інформація про навколишнє середовище може мати різні форми та призначення. Основні інформаційні особливості походження (атрибутивності), структурування, використання та фокусування семантичного значення створюють основу для формальної класифікації та адекватної інтерпретації інформації під час ситуаційного управління, правильного прийняття рішень та їх реалізації. Для цих цілей були розроблені таксономія інформації та формальна онтологія. Таким чином, практичне використання запропонованих моделей пов'язане з побудовою адекватних інформаційних моделей для систем ситуаційного управління, їх належної обробки та інтерпретації.

Інформація є предметом і продуктом реалізації технологій ситуаційного управління в ССУ. Отже, виникає необхідність використання адекватної моделі злиття та виділення предметної інформації у процесах СУ. Проблемам злиття інформації в динамічних ССУ присвячена велика кількість публікацій. Зокрема, у роботі «Управління злиттям інформації на високому рівні та проектування систем» [217] проведено узагальнення та наведені приклади використання (переважно у військовій сфері) підходів та моделей стосовно злиття інформації. Відповідно до моделі I-SDKW [6], [25] розрізняють моделі злиття даних та злиття знань.

Ситуаційне управління пов'язане з консолідацією та обробкою інформації різнорідного походження для отримання семантичної інформації стосовно цільової системи. Для цього використовуються адекватні засоби (принципи, методи та інструменти) консолідації інформації. Найбільш поширений підхід до консолідації інформації ґрунтується на методах злиття інформації.

Відповідно до моделі інформаційної групи злиття даних (Data Fusion Information Group, DFIG) [217], [218] визначено шість рівнів злиття інформації:

- рівень 0 – оцінка даних;
- рівень 1 – оцінка об'єкта;
- рівень 3 – оцінка впливу;
- рівень 4 – уточнення процесу;
- рівень 5 – уточнення користувача;
- рівень 6 – управління місією (завданням).

Уточнення та практичне злиття інформації ґрунтується на строгій онтології понять відповідно до моделі І-СДЗМ (I-SDKW) [25], [33].

Семантика (зміст) ситуації визначається її модальністю, яка виражає відношення між ситуацією і дійсністю (об'єктивна модальність) та суб'єктом і ситуацією (суб'єктивна модальність). Алгебраїчна інтерпретація модальної логіки стала основою для розробки семантики модальних систем [219] та їх застосування у системах ситуаційного управління і підтримки прийняття рішень. Урахування модальностей ситуацій та їх формальне представлення у вигляді модальних систем дозволяють вийти за обмеження семантики класичних логічних систем та наблизити формальний опис ситуацій до реального [20].

Одним із поширених способів управління системами знань є онтології [220]. Формальна онтологія представляється кортежем

$$O = \langle T, R, F \rangle, \quad (2.4)$$

де T – сутності (концепти) прикладної області (ПрО), яку описує онтологія O , R – відношення між концептами заданої ПрО, F – функції інтерпретації, задані на концептах і/або відношеннях онтології O . Таким чином, для представлення знань у формі онтології встановлюються відношення між поняттями ПрО та визначаються логічні функції приписування відповідного змісту (семантики) для описів (висловлювань) цієї ПрО [15], [26], [40], [221], [222], [223], [224].

Найпростішими логічними функціями інтерпретації є класичні логіки висловлювань [225]. Але ще за часів античності відзначалось, що логічні міркування на основі класичних логік не відображають всього багатства та різноманіття живої мови з урахуванням модальностей висловлювань. Однією із сфер застосування модальної логіки є прийняття рішень [20], зокрема, в системах ситуаційного управління. В роботі [226] виділено три фази ситуаційного управління: моделювання ситуацій, розпізнавання ситуацій та реагування на ситуації.

Реалізація фаз ситуаційного управління ґрунтується на використанні узгоджених із класифікацією ситуацій формалізованих знань прикладної області ситуаційного управління. Формалізовані знання представляють різні рівні інформаційного забезпечення процесів функціонування ССУ (рис. 2.4).

При побудові ССУ потрібно створювати моделі, які представляють систему на різних рівнях (рис. 2.9) [46]. Місія, або цільове призначення, системи представляє стратегічний рівень інформаційного забезпечення ССУ, що визначає загальний контекст функціонування ССУ. Цілі представляють тактичний рівень інформаційного забезпечення ССУ і визначають умови і критерії досяжності результатів діяльності ССУ. Операційний рівень представляє функціональні задачі діяльності, що визначають алгоритми і процедури трансформації ситуаційної інформації в управлінські рішення, які реалізуються через процеси діяльності ССУ і підтримуються відповідною архітектурою ССУ.

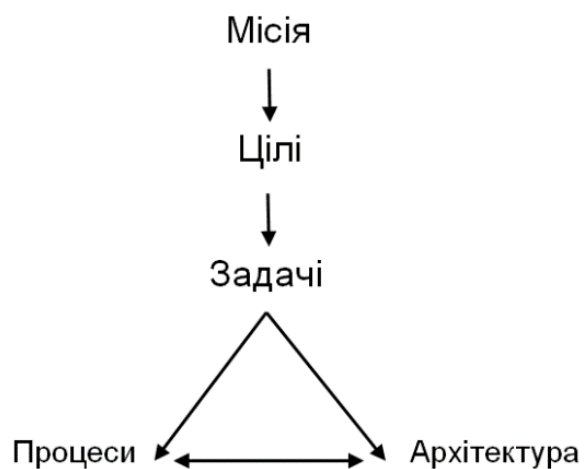


Рисунок 2.9 – Рівні моделювання ССУ

Отже, конвергентна модель знань ССУ повинна поєднувати в собі стратегічний, тактичний і операційний рівні інформаційного забезпечення. Місія ССУ визначається контекстом ситуації, що представляється атрибутами концептів предметної моделі знань. Для представлення місії ССУ розроблена класифікація ситуацій на основі їх типів і модальностей. Представлення типів ситуацій ґрунтується на ієрархії первинних таксономічних категорій (Т-категорій) [146]. Для представлення модальностей відношень при описі ситуацій використовуються логічні модальності [20]:

- алетичні (alethic) – оперують базовими модальностями необхідно (N – necessary), можливо (P – possible), неможливо (I – impossible), ймовірно можливо (C – contingent);

- деонтичні (deontic) – оперують модальностями обов’язково (O – obligatory), дозволено (P – permissible), заборонено (F – forbidden);

- аксіологічні (axiological) – оперують із суб’єктивними модальностями добре (G), нейтрально (N), погано (B);

- епістемічні (epistemic) – оперують із модальностями знання, припущення, незнання і можуть описувати стан знань агентів у агентних системах. Знання (незнання) означає, що група агентів мають (не мають) певні знання. Припущення означає, що для групи агентів, які мають певні знання, можуть бути справедливими певні твердження;

- часові (temporal) – оперують із модальностями істинності тверджень у різних часових контекстах – лінійного слідування, інтервалів, дерев обчислень тощо;

- просторові (spatial) – оперують із модальностями простору: там, тут, ніде, поруч, вдалині тощо;

- докастичні (doxastic) – визначають множину довірчих тверджень $B: \{b_1, b_2, \dots, b_n\}$, на основі яких формуються різні типи міркувань (логічних побудов) – точні, неточні, «самовпевнені», послідовні, нормальні, особливі, «дивні», «обережні» тощо;

– гібридні (розширені) – використовують додаткові символи номіналів, які є істинними лише в одному з можливих світів семантики Кріпке.

Поєднання модальностей у визначенні ситуацій обумовлене семантикою конкретного світу (семантикою Кріпке) і визначає обмеження інтерпретації та контекст ситуації. Онтологічна модель знань ситуаційного управління була реалізована у редакторі онтологій Protégé. Фрагмент онтології концепту «ситуація» представлено на рис. 2.10 [40].

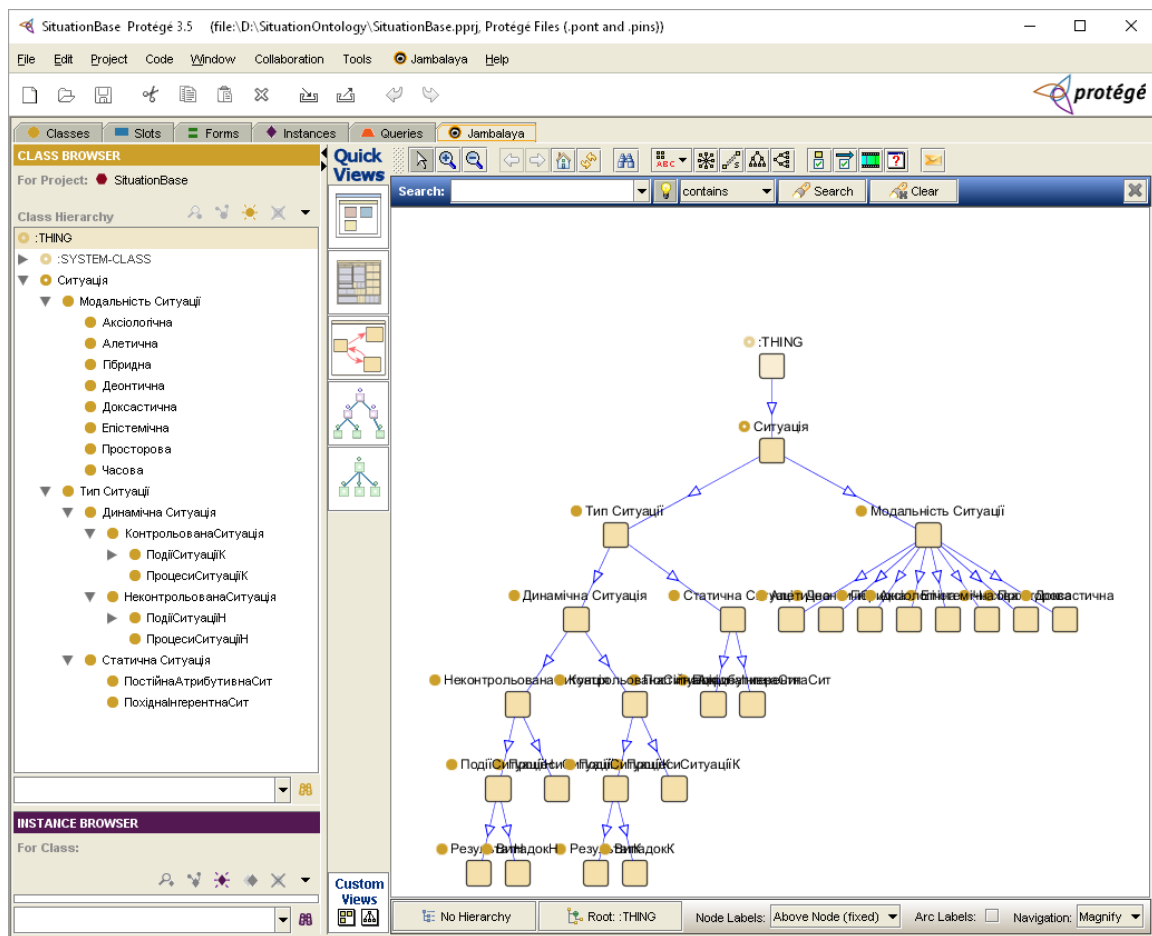


Рисунок 2.10 – Фрагмент онтології концепту «Ситуація»

Онтологія концепту «ситуація» створює основу для опису семантики ситуації у контексті реалізації процесів ситуаційного управління в ССУ (рис. 2.10). Семантика ситуації описується математичним апаратом ситуаційної теорії. Опис ситуації в ситуаційній теорії базується на понятті інфону. Інфон – це деталь (фрагмент) інформації про ситуацію, яка представлена у вигляді [227]

$$\langle\langle R, A, Pol \rangle\rangle, \quad (2.5)$$

де R – n -місне відношення, $A = \langle a_1, \dots, a_n \rangle$ – кортеж відповідних об'єктів для R , значення $Pol = \{0 \text{ або } 1\}$ представляє полярність, що вказує на відсутність або існування відношення R стосовно кортежу об'єктів A .

Для відображення того, що інфон σ актуалізується ситуацією s , пишуть $s \models \sigma$ і стверджують, що s підтримує σ . Базові типи інфонів:

- *TIM*: тип часової локалізації;
- *LOC*: тип просторової локалізації;
- *IND*: тип індивіду;
- *REL_n*: тип n -місного відношення;
- *SIT*: тип ситуації;
- *INF*: тип інфону;
- *TYP*: тип типу;
- *PAR*: тип параметра;
- *POL*: тип полярності (0 або 1).

Інфони будуються з базових типів. Наприклад, якщо l є локалізацією, тоді l має тип *LOC*, а інфон $\langle\langle of\text{-}type, l, LOC, 1 \rangle\rangle$ є фактом. Для представлення модальності в описі ситуації вводимо новий тип: *MOD (modality)*. Об'єкти (*uniformities*) модальності в онтології типів позначимо, як m_0, m_1, m_2 тощо.

Джерелами отримання ситуаційної інформації можуть бути архівні дані, корпоративні дані, транзакційні дані, соціальні та інші медіа, генеровані дані, публічна інформація тощо. Сама ситуаційна інформація відноситься до однієї з категорій: структурована, напівструктурована, квазіструктурована, неструктурована.

Консолідація даних у системах ситуаційного управління здійснюється на основі онтології ситуаційної інформації. Для побудови онтології використовується таксономія таких концептів:

- джерела даних;
- формат даних;

- тип даних;
- категорія інформації;
- рівень важливості інформації;
- модальність інформації;
- ступінь довіри до інформації.

Ситуаційне числення (Situational Calculus) є формальною логічною системою (логікою першого порядку), розробленою для представлення і обґрунтування динамічно змінюваних світів, доменів і систем. У логіці першого порядку (Л1П) висловлювання або істинні, або хибні і такими залишаються завжди. Нічого не може призвести до будь-яких змін. Ситуаційне числення представляє змінні сценарії як множину формул логік другого порядку (Л2П). Основними поняттями ситуаційного числення є дії, флюенти, ситуації.

Дії (Actions) можуть бути виконані у середовищі (світі, системі). Дії визначаються функціональним символом і можуть бути оцінені кількісно. Дії представляються термами першого порядку. Наприклад, $Open(x, y)$ визначає дію, при якій об'єкт x відкриває об'єкт y ; $Do(open(A, B), s)$ визначає стан, який є результатом того, що A відкрив B , коли світ перебуває у стані (ситуації) s .

Флюенти (Fluents) – це змінні величини, що описують стан середовища (світу). Флюент є відношенням або функцією, значення якої можуть змінюватися залежно від ситуації. Флюенти (змінні) визначаються з використанням предикатів або функціональних символів, що приймають терм ситуації як аргумент. Наприклад, $closeTo(x, y, s)$ означає, що x близький до y в ситуації s ; $Pos(x, s)$ визначає розташування x у ситуації s .

Ситуації (situations) в ситуаційному численні представляють історію проявів (результатів) дій. Динамічне середовище моделюється у розвитку через серію (послідовність) ситуацій як наслідків різних дій, виконуваних у цьому середовищі. Послідовність цих дій є скінченною. Ситуація описує не стан, а історію (контекст стану) і є термом першого порядку, що визначає ймовірнісну історію світу. Ситуація представлена бінарним функціональним символом do : $do(a, s)$ визначає історію, що

є результатом додавання нової дії a в історію s . Ця ситуація також визначає стан світу.

Домен кодується у Л2П (логіці другого порядку) трьома типами формул:

- аксіомами передумов дій та аксіомами результатів (наслідків) дій;
- аксіомами послідовних станів, одним із яких є флюент (fluent);
- основоположними аксіомами ситуаційного числення.

2.3 Архітектурний аспект побудови систем ситуаційного управління

При розробці архітектури ССУ, в залежності від особливостей предметної області її застосування, використовується одна або композиція різних модельно-орієнтованих концепцій та методологій розробки:

- модельно-орієнтована інженерія систем (MBSE);
- подійно-орієнтована розробка архітектури (Event-driven Architecture);
- розробка, керована цілями (Goal-Driven Development);
- розробка, керована вимогами (Requirements-Driven Development);
- керована моделями архітектура (MDA);
- ситуативна (ad-hoc) архітектура.

Оскільки ситуаційне управління загалом стосується непередбачуваних випадкових подій, то цільова ситуаційна система повинна бути адаптована до специфіки конкретної ситуації. Така адаптація може бути реалізована на основі ad-hoc архітектури системи. Варіанти ситуативної (ad-hoc) архітектури можуть бути розроблені для конкретної ситуаційної проблеми або задачі, без узагальнення варіанта рішення (варіант зберігається лише для вибору на основі конкретних випадків), не маючи наміру їх адаптувати до інших цілей. Таким чином, архітектурні представлення повинні відображати точки зору, узгоджені з контекстом (семантикою) ситуації, позиціями зацікавлених сторін та наявними необхідними засобами управління.

Архітектурна модель служить простим, але потужним інструментом щодо застосування системного підходу для планування робіт зі створення і використання інформаційних систем та їх стикування. Схема архітектури дозволяє концентруватися на окремих аспектах системи і в той же час не втрачати відчуття загального контексту, тобто, погляду на підприємство в цілому.

На практиці такий підхід може використовуватися таким чином. По-перше, дану модель зручно застосовувати для класифікації всієї інформації, що описує підприємство та інформаційні системи цього підприємства, виявлення "білих плям" та координацію робіт. По-друге, дану модель можна використовувати на метарівні для порівняння різних реалізацій створення архітектур підприємства. Нарешті, вона може бути зручним засобом для використання в окремих проєктах.

Розробка ІТ-архітектури підприємства включає в себе компоненти, пов'язані з функціональною архітектурою, інформаційними технологіями (ІТ) та управлінням архітектурним процесом, і заснована на стратегії розвитку підприємства. Таким чином, ІТ-архітектура підприємства є цілісним описом ключових стратегій організації, пов'язаних з інформацією, прикладними системами і технологіями, а також їх впливом на функції і бізнес-процеси організації. Розробка ІТ-архітектури підприємства ведеться у відповідному контексті існуючих в організації структур управління і взаємодії [228].

Значний внесок у розвиток концепції архітектури підприємства був зроблений Дж. Захманом (John A. Zachman). З 1987 року, коли була запропонована перша версія моделі Захмана [229], розвинута згодом у роботах 1992–1996 рр., вона була використана достатньо великою кількістю великих компаній, що входять у список 2000 найбільших корпорацій світу.

Модель Захмана заснована на дисципліні класичної архітектури і забезпечує загальний словник і набір перспектив або структур для опису сучасних складних корпоративних систем. У своїй роботі Дж. Захман визначив архітектуру підприємства як «набір описових моделей, які застосовані для опису підприємства відповідно до вимог управлінського персоналу і які можуть розвиватися протягом певного періоду». Термін «архітектура» тут не випадковий, він підкреслює існуючу

аналогію між внутрішньою структурою абстрактного об'єкта-підприємства і складного штучного об'єкта, такого, як будівля або космічна станція.

Для зручності опису Захман запропонував так звану модель архітектури підприємства. Модель переслідує дві основні мети: з одного боку, логічно розбити весь опис архітектури на окремі розділи для спрощення їх формування та сприйняття, з іншого – забезпечити можливість розгляду цілісної архітектури з виділених точок зору або відповідних рівнів абстракції.

У той час, коли були опубліковані роботи Захмана, традиційним підходом при формуванні опису системи було використання концепції «життєвого циклу», що включає такі етапи, як планування, аналіз, проектування, розробка, документування, впровадження і промислова експлуатація. На кожному з цих етапів розглядаються питання, пов'язані як з функціями системи, так і даними. Захман запропонував замість традиційного підходу, пов'язаного з розглядом окремих аспектів роботи системи в різні моменти часу, використовувати розгляд системи з різних перспектив.

Історично модель Захмана вперше була створена саме для ІТ-систем. Цей підхід у подальшій роботі був узагальнений для розгляду не тільки ІТ-систем, але і для опису підприємства в цілому, так що запропонована модель може використовуватися як засіб для опису архітектур складних систем будь-якого типу.

Основна ідея полягає в тому, щоб забезпечити можливість послідовного опису кожного окремого аспекту системи в координації з усіма іншими. Для будь-якої достатньо складної системи загальне число зв'язків, умов і правил зазвичай перевершує можливості для одночасного розгляду. У той же час окремий, у відриві від інших, розгляд кожного аспекту системи найчастіше призводить до неоптимальних рішень як у плані продуктивності, так і вартості реалізації. Остання, третя, версія моделі архітектури підприємства Захмана отримала підзаголовок «Онтологія підприємства», що вказує орієнтацію на використання моделей знань при побудові та організації діяльності підприємства.

Важливим принципом моделі Захмана є необхідність послідовного переходу при поглибленні деталізації розгляду. Пропуск окремих елементів майже завжди

призводить до невдачі. На практиці це часто трапляється при спробі розробки програми на підставі тільки усного опису вимог користувача. Основні характеристики даної моделі:

- простота для розуміння як технічними, так і нетехнічними фахівцями;
- цілісність відносно підприємства;
- підтримка обговорень складних питань із використанням відносно невеликої кількості нетехнічних понять;
- можливість застосування для планування, що дозволяє краще приймати рішення;
- застосовність для вирішення завдань, тобто, можливість працювати з абстракціями і сутностями, виділяючи та ізолюючи окремі параметри системи без втрати сприйняття підприємства як цілого;
- незалежність від конкретних інструментів; завдяки цьому кожен інструмент і методологія можуть бути відображені на дану модель і можуть явно показати, що вони роблять і чого вони не роблять.

Модель Захмана передбачає шість аспектів архітектурного опису [229]:

- цілі організації, базові правила, за якими вона працює, та мотивації, які обумовлюють діяльність;
- персонал, підрозділи та інші елементи організаційної структури, зв'язки між ними, розподіл повноважень і відповідальності в рамках структури організації;
- сутності і дані, з якими має справу організація, та перелік засобів, які для цього застосовуються;
- виконувані організацією і різними її підрозділами функції та операції над даними і процеси, які їх реалізують;
- географічний розподіл елементів організації і зв'язки між географічно розділеними її частинами у межах мережі поширення діяльності організації;
- часові характеристики і обмеження на діяльність організації, значущі для її координованої діяльності події, що відповідають часовим циклам здійснення процесів діяльності в організації.

У свою чергу, для кожного аспекту передбачається опис на шести рівнях абстрагування:

- перспективного планування і створення моделей контексту діяльності;
- перспективного управління діяльністю і створення моделей концепції здійснення діяльності;
- архітектурного представлення і створення моделей архітектурного опису системи;
- технологічного представлення і створення моделей специфікації технологій;
- технічного представлення реалізації компонентів технологій і створення моделей конфігурування інструментальних засобів технологічної підтримки;
- представлення діяльності і створення моделей реалізації операцій для конкретних видів діяльності.

Матричне представлення онтології архітектурної моделі Захмана проілюстровано на рис. 2.11.

Модель Захмана [229] послужила основою для створення цілого ряду інших методик і моделей опису архітектури підприємства, таких як Федеральна архітектурна модель (FEAF) США [230], методика опису архітектури Open Group (TOGAF) [231], методика опису архітектури міністерства оборони США (DoDAF) [232], UPDM [113]. Зокрема, групи метамodelей DoDAF V2.0 підтримують різні аспекти розгляду та шість ключових процесів DoD, що виконуються в рамках Системи розвитку інтеграції спільних спроможностей (CPICC, The Joint Capabilities Integration Development System – JCIDS), Системи оборонних закупівель (Defense Acquisition System – DAS), ресурсного планування, розподілу, бюджетування та забезпечення (Planning, Programming, Budgeting, and Execution PPBE), інженерії систем (Systems Engineering), здійснення операцій (операційної діяльності, Operations) та управління портфелями інвестицій в ІТ та спроможності (Portfolio Management (IT and Capability)). Існуючі моделі та методики опису архітектури КС задають класифікацію основних областей і єдині принципи для їх опису у системній взаємодії, описи використовуваних політик, стандартів, процесів, моделей для визначення різних елементів архітектури.

Імена класифікації Проективна аудиторія	Що (інформація)	Як (функція)	Де (розташування, мережа)	Хто (персонал)	Коли (час)	Чому (мотивація)	Імена класифікації Назви моделей
Проекція (ракурс) виконання (планувальники контексту діяльності)	Ідентифікація засобів Список Типи засобів	Ідентифікація процесів Список Типи процесів	Ідентифікація розповсюдження Список Типи розповсюдження	Ідентифікація відповідальності Список Типи відповідальності	Ідентифікація координації Список Типи координації	Ідентифікація мотивації Список Типи мотивацій	Контекст сфери діяльності (списки ідентифікації сфери)
Проекція управління діяльністю (власники концепції діяльності)	Визначення засобів (Сутність діяльності; зв'язки між діяльностями)	Визначення процесу (Перетворення діяльності; Вхід/Вихід) діяльності	Визначення розповсюдження (Локалізація діяльності; прив'язка діяльності)	Визначення відповідальності (Робоча роль; результат виконання ролі)	Визначення координації (Інтервал діяльності; моменти часу діяльності)	Визначення мотивацій (Результат діяльності; цілі операцій)	Концепція діяльності (моделі визначення обов'язків)
Архітектурна проекція (конструктори логіки діяльності)	Представлення засобів (Системна сутність; системні зв'язки)	Представлення процесу (Системні перетворення; Вхід/Вихід системи)	Представлення розповсюдження (Розташування системи; системний зв'язок)	Представлення відповідальності (Системна роль; результат виконання системної ролі)	Представлення координації (Системний інтервал; системний момент)	Представлення мотивації (Системний результат; системні цілі)	Системна логіка (моделі представлення системи)
Інженерна (технологічна) проекція (будівники (інтегратори) технологій (фізики діяльності))	Специфікація засобу (Сутність технології; прив'язка технології)	Специфікація процесу (Технологічне перетворення; Вхід/Вихід технології)	Специфікація розповсюдження (Розташування технології; прив'язка технології)	Специфікація відповідальності (Технологічна роль; результат виконання технологічної ролі)	Специфікація координації (Технологічний інтервал; технологічний момент)	Специфікація мотивації (Технологічний результат; технологічні цілі)	Фізика (реалізація) технології (моделі опису (специфікації) технологій)
Технічна проекція (реалізація компонентів діяльності)	Конфігурація засобу (Сутність інструменту; прив'язка інструменту)	Конфігурація процесу (Інструментальне перетворення; Вхід/Вихід засобу)	Конфігурація розповсюдження (Розташування інструменту; прив'язка засобу)	Конфігурація відповідальності (Роль інструменту; результат застосування засобу)	Конфігурація координації (Інтервал використання засобу; момент засобу)	Конфігурація мотивації (Результат використання засобу; цілі засобу)	Інструментальні компоненти (моделі конфігурування засобів/інструментів)
Проекція діяльності (користувачі). Вид діяльності	Варіанти реалізації засобів (Сутності операцій; зв'язки (відношення) операцій)	Варіанти реалізації процесів (Перетворення (трансформація) операцій, Вхід/Вихід операцій)	Варіанти розповсюдження (Місце виконання операцій, зв'язки між операціями)	Варіанти розподілу обов'язків (відповідальності) (Операційні ролі, продукти здійснення операцій)	Варіанти координації (розподілу часу) (Інтервали операцій, міти (моменти) часу операцій)	Варіанти мотивацій (Результати операцій, цілі (наміри) операцій)	Приклади операцій (реалізації). Вид діяльності
Проективні аудиторії Назви діяльності	Набори засобів	Потоки процесів	Мережі розповсюдження	Відповідальні за завдання	Часові цикли	Мотиваційні наміри (задуми)	

Рисунок 2.11 – Матриця онтології архітектурної моделі Захмана

Моделі, описані в DoDAF, згруповані за такими точками зору (аспектами розгляду):

- загальна точка зору (AV);
- точка зору спроможностей (CV);
- операційна точка зору (OV);
- сервісна точка зору (SvcV);
- системна точка зору (SV);
- точка зору на дані та інформацію (DIV);
- проектна точка зору (PV);
- точка зору на стандартизацію (StdV);

Співставлення архітектурної моделі Захмана з точками зору на компоненти моделі DoDAF наведено у табл. 2.2.

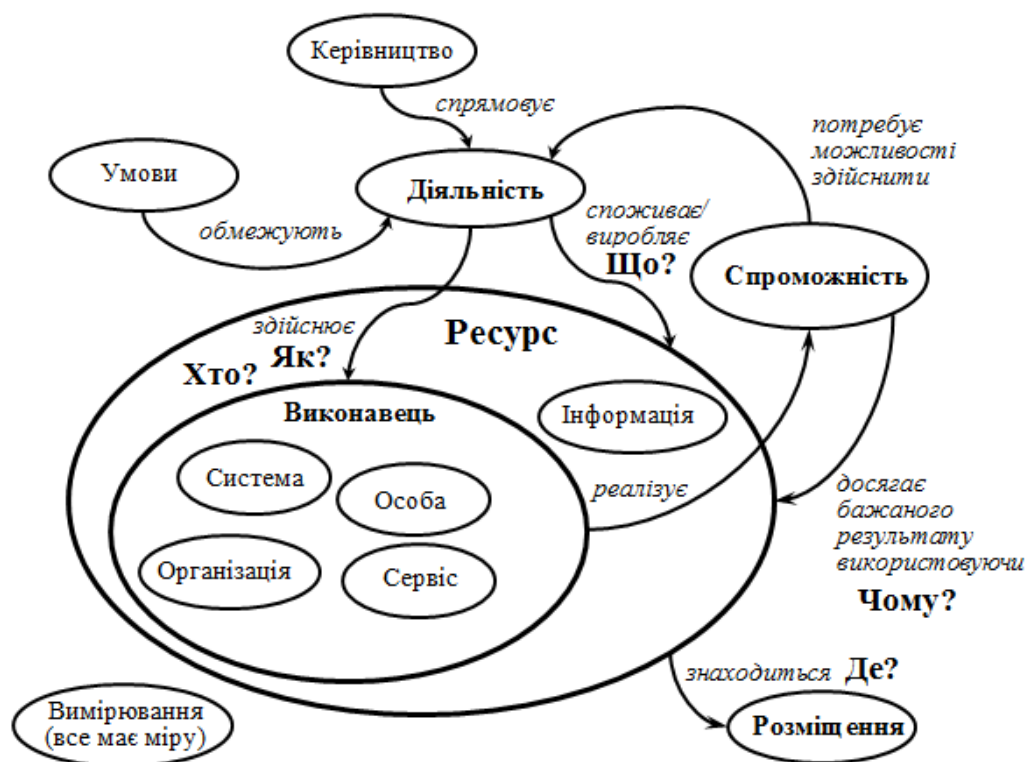


Рисунок 2.12 – Взаємозв’язки між сутностями метамоделі даних DoDAF і архітектурними аспектами моделі Захмана

Таблиця 2.3 – Відображення груп даних метамоделі DoDAF на точки зору та ключові процеси

	Точки зору	Ключові процеси DoD
Групи даних метамоделі	AV, CV, DIV,OV,PV,StdV, SvcV, SV	JCIDS (J), DAS (D), PPBE (P), System Engineering (S), Operations (O), Portfolio Management (IT and Capability) (C)
Виконавець	CV, OV, PV, StdV, SvcV, SV	J, D, P, S, O, C
Діяльність	OV	J, O, C
Потік ресурсів	AV, CV, DIV,OV,PV,StdV	J, S, O

Продовження таблиці 2.3

Дані та інформація	AV, DIV	J, D, P, S, O, C
Спроможність	CV, PV, SV, SvcV	J, D, P, S, O, C
Сервіси (служби)	CV, StdV, SV	P, S, C
Проект	AV, CV, PV, SvcV, SV	D, P, S, C
Тренінг/Навичка/Освіта	OV, SV, SvcV, StdV	J, S, O
Цілі	CV, PV	J, D, P, O, C
Правила	OV, StdV, SvcV, SV	J, D, S, O
Вимірювання	SvcV, SV	J, D, S, O, C
Розміщення	SvcV, SV	P, S, O

Узагальнюючою архітектурною моделлю для UPDM, DoDAF, MoDAF і NAF є уніфікована архітектурна модель OMG UAF [233]. Модель UAF застосовується для предметних областей (доменів), де використовуються моделі DoDAF, MoDAF та NAF (архітектурна модель НАТО). Модель UAF є розвинутим засобом для моделювання складних системних архітектур, забезпечує гнучкість моделювання і реалізована як профіль UML на основі SysML. Модель UAF є основою для аналізу складних кіберконвергентних систем і надає:

- спільну термінологію в рамках конвергентних предметних областей (доменів);

- загально прийнятні поняття (концепти) та моделі;

- спільні мовні засоби в рамках конвергентної системи;

- уніфікацію термінології при застосуванні декількох еталонних моделей;

- підтримує реалізацію моделей методологій на основі MBSE/SysML.

Фундаментальними принципами проєктування на основі UAF є:

- управління на основі вимог;

- вплив попередніх архітектурних моделей;

- управління на основі онтології IDEAS (International Defence Enterprise Architecture Specification for Exchange);

- використання нотації метамоделі цільової області DMM (Domain Metamodel) на основі діаграм класів UML;
- повторне використання семантики концептів UML-метамоделі;
- повторне використання семантики процесів нотації BPMN.

Через складність одночасного управління кількома точками зору із співставленням задач та метамodelей стандартні точки зору на архітектуру трансформуються у більш керований формат представлення у вигляді матриці UAF (табл. 2.4). Матриця UAF [233] – це спосіб показати, як різні точки зору відповідають доменам (горизонтальні рядки) та типам моделей (стовпці), що описують специфікацію точки зору на архітектуру системи. Призначення матриці UAF полягає в тому, щоб представити інформацію, яка присутня в архітектурних моделях, що стали основою для UAF.

Специфікації виду в домені метаданих (табл. 2.4) не моделюються як частина UAF, але є архітектурними артефактами, які підтримують визначення та розробку архітектури. Для оцінки поведінки та обмежень архітектури, тобто нефункціональних вимог, необхідно визначити фактичні екземпляри архітектурних елементів. Інструментальні засоби для архітектурного проєктування на основі UAF повинні забезпечувати моделювання поведінки та оцінку вимірювань і обмежень за допомогою параметричних діаграм або аналогічних засобів. Інформаційна модель охоплює всі домени і може бути визначена у будь-якій з її форм: концептуальній, логічній або фізичній. Очікується, що більшість розробників інформаційної моделі будуть використовувати концептуальну або логічну форми моделі даних при використанні інструменту абстрактного моделювання. Параметрична модель фіксує вимірювання та оточення архітектури і охоплює всі домени. Фізична модель даних визначається відповідними зовнішніми засобами, не пов'язаними з UAF. Вид таксономії метаданих (Md-Tx) надає засоби для поширення архітектурної моделі на інші домени.

Таблиця 2.4 – Матриця співставлення доменів і типів моделей ландшафту UAF

Типи моделей Домен	Таксономія (Tx)	Структура (Sr)	Зв'язність (Cn)	Процеси (Pr)	Стани (St)	Сценарії взаємодії (Is)	Інформація (If)	Параметри (Pm)	Обмеження (Ct)	Дорожня карта (Rm)	Відстежува- ність (Tr)
Метаданих (Md)	Md-Tx	Md-Sr	Md-Cn	Md-Pr	Md-St		Моделі даних: концептуальна модель даних, логічна модель даних, фізична схема (модель)	Оточення (Pn-En) та вимірювання (Pm-Me)	Md-Ct	Md-Rm	Md-Tr
Стратегії (St)	St-Tx	St-Sr	St-Cn		St-St				St-Ct	St-Rm	St-Tr
Операцій (Op)	Op-Tx	Op-Sr	Op-Cn	Op-Pr	Op-St	Op-Is			Op-Ct		Op-Tr
Сервісів (Sv)	Sv-Tx	Sv-Sr	Sv-Cn	Sv-Pr	Sv-St	Sv-Is			Sv-Ct	Sv-Rm	Sv-Tr
Персоналу (Pr)	Pr-Tx	Pr-Sr	Pr-Cn	Pr-Pr	Pr-St	Pr-Is			Pr-Ct	Pr-Rm	Pr-Tr

Продовження таблиці 2.4

Ресурсів (Rs)	Rs-Tx	Rs-Sr	Rs-Cn	Rs-Pr	Rs-St	Rs-Is	Моделі даних: концептуальна модель даних, логічна модель даних, фізична схема (модель)	Оточення (Pn-En) та вимірювання (Pm-Me)	Rs-Ct	Rs-Rm	Rs-Tr
Безпеки (Sc)	Sc-Tx	Sc-Sr	Sc-Cn	Sc-Pr					Sc-Ct		Sc-Tr
Проектів (Pj)	Pj-Tx	Pj-Sr	Pj-Cn	Pj-Pr						Pj-Rm	Pj-Tr
Стандартів (Sd)	Sd-Tx	Sd-Sr								Sd-Rm	Sd-Tr
Фактичних ресурсів (Ar)		Ar-Sr	Ar-Cn	Моделювання/симуляція					Параметричне виконання / оцінка		

Зміст доменів ландшафту UAF представлений у табл. 2.5.

Таблиця 2.5 – Домени ландшафту UAF

Домен	Позначення	Опис домену
Метаданих	Md	Визначає метадані, необхідні для розробки відповідної архітектури, яка відповідає своєму призначенню
Стратегії	St	Процес управління спроможностями. Описується таксономія спроможностями, склад, залежності та еволюція
Операцій	Op	Ілюструє логічну архітектуру системи. Описує вимоги, операційну поведінку, структуру та взаємодію, необхідні для підтримки (представлення) спроможностей. Визначає всі операційні елементи незалежно від реалізації/рішення
Сервісів	Sv	Сервісно-орієнтований ландшафт (SOV) описує сервіси, необхідні для безпосередньої підтримки домену операцій, відповідно до опису ландшафту домену Операцій
Персоналу	Pr	Визначає та досліджує типи організаційних ресурсів. Показує таксономію типів організаційних ресурсів, а також зв'язки, взаємодію та зростання з часом
Ресурсів	Rs	Фіксує архітектуру рішення, що складається з ресурсів, наприклад, організаційних, програмних, артефактів, конфігурацій спроможностей та доступних ресурсів, що реалізують експлуатаційні вимоги. Проектування ресурсів деталізується на основі використання нотацій SysML або UML

Продовження таблиці 2.5

Безпеки	Sc	Охоронні активи та охоронні анклав. Визначає ієрархію активів безпеки та власників активів, обмеження безпеки (політика, закони та вказівки) та деталі, де вони знаходяться (анклави безпеки)
Проектів	Pj	Описує проекти та етапи проекту, як ці проекти забезпечують можливості, організації, що сприяють проектам, та залежності між проектами
Стандартів	Sd	Визначає набір правил, що регулюють порядок, взаємодію та взаємозалежність частин або елементів рішення
Фактичних ресурсів	Ar	Аналіз, наприклад, оцінка різних альтернатив, компроміси, верифікація та валідація щодо фактичних конфігурацій ресурсів. Ілюструє очікувані або досягнуті фактичні конфігурації ресурсів

Методика моделювання архітектури є інструментом для створення широкого спектра різних архітектур. Вона, як правило, включає в себе опис методів проектування ІТ-архітектури в термінах використання певних «будівельних блоків», опис того, як ці «будівельні блоки» пов'язані між собою, набір інструментів для опису елементів архітектури, загальний словник використовуваних термінів. Методики також можуть містити список рекомендованих стандартів і сумісних продуктів, які можуть використовуватися для реалізації різних елементів архітектури. Важливо розуміти, що методики не тільки задають набір документів і планів, необхідних для опису підприємства, але й визначають, як усі ці елементи опису пов'язані між собою.

Індустріальні стандарти для опису ІТ-архітектури підприємства розробляються також такими організаціями, як IEEE, ISO, описані в ITIL, COBIT, та ін. Але жоден із цих стандартів не займає домінуючого положення. Більше того,

жоден із них, узятий окремо, не дає групам розробників ІТ-архітектури всіх необхідних інструментів із методичної точки зору і точки зору шаблонів, використовуваних для опису архітектури. Однак цей накопичений арсенал методик і стандартів надає архітекторам широкі можливості вибору моделей, прикладів і досвіду різних індустрій.

Опис архітектури конвергентної ССУ служить детальним керівництвом, яке визначає основні, стандартні або типові елементи конвергентної ССУ, їх взаємозв'язок, а також процеси управління інформаційними системами [44]. Для опису архітектури конвергентної ССУ використовуються різні формалізовані моделі та засоби. Важливо, щоб при описі конвергентної ССУ використовувався такий формат, який би забезпечував легкий для розуміння спосіб представлення всіх аспектів і точок зору на конвергентну ССУ. Для опису та моделювання архітектури КС застосовуються підходи, методології та інструментальні засоби моделювання універсального призначення. До засобів моделювання архітектури систем висуваються такі вимоги:

- високий рівень деталізації для практичного використання фахівцями в області ІТ при розробці нових систем;
- простота для розуміння нефаховою аудиторією;
- динаміка розгляду (тобто, «Архітектура як є» – «Короткострокові і середньострокові завдання» – «Стратегічні плани»);
- можливість адаптації за новими вимогами бізнесу та врахування можливостей реалізації незапланованих проєктів.

Існують різні визначення системної архітектури [229], [231], [230], і одне з найбільш узагальнених визначень представлено у стандарті ISO/IEC/IEEE 42010:2011 «Systems and software engineering – Architecture description» [117]: архітектура – це фундаментальні концепції або властивості системи в її оточуючому середовищі, втілені в її елементах, зв'язках і принципах проєктування та еволюції системи.

ISO/IEC 42010 (крім терміну «архітектура») визначає перелік інших термінів:

- створення архітектури (architecting): процес задуму, визначення, формулювання, документування, передачі, сертифікації відповідності реалізації, підтримки і поліпшення архітектури на протязі всього життєвого циклу системи (тобто, «проектування»);
- опис архітектури (architecture description – AD): продукт діяльності, який використовується для відображення архітектури;
- мова опису архітектури (architecture description language – ADL): будь-яка форма вираження для використання в описах архітектури;
- модель (каркас) архітектури (architecture framework): домовленості, принципи і практичні прийоми для опису архітектур, встановлені у конкретній предметній області застосування і/або співтоваристві зацікавлених сторін;
- точка зору на архітектуру (architecture viewpoint): продукт діяльності, що встановлює угоди для побудови, інтерпретації та використання архітектурних поглядів для окреслення конкретних ставлень до системи;
- ставлення (concern): зацікавленість у системі стосовно однієї або більшої кількості зацікавлених осіб, пов'язаних із системою. Ставлення відноситься до будь-якого впливу на систему в її середовищі, включаючи впливи на розвиток, технологію, діяльність, операції, організацію, політику, економіку, екологію, юридичні, нормативні і соціальні аспекти;
- вид моделі: домовленості про тип моделювання. Архітектурний ландшафт складається з декількох моделей, кожна з яких слідує певному виду моделі;
- зацікавлена сторона: особа, команда, організація або їх групи, що мають зацікавленість у системі.

Стандарт ISO/IEC/IEEE 42010 встановлює чітку відмінність між архітектурами та описами архітектур. Контекст для опису архітектури стандарту ISO/IEC/IEEE 42010 представлений на рис. 2.13 [117].

Перманентний актуальний аналіз методик архітектурного моделювання, що відповідають стандарту ISO/IEC/IEEE 42010:2011 [117], проводиться на аналітичному сайті [234].

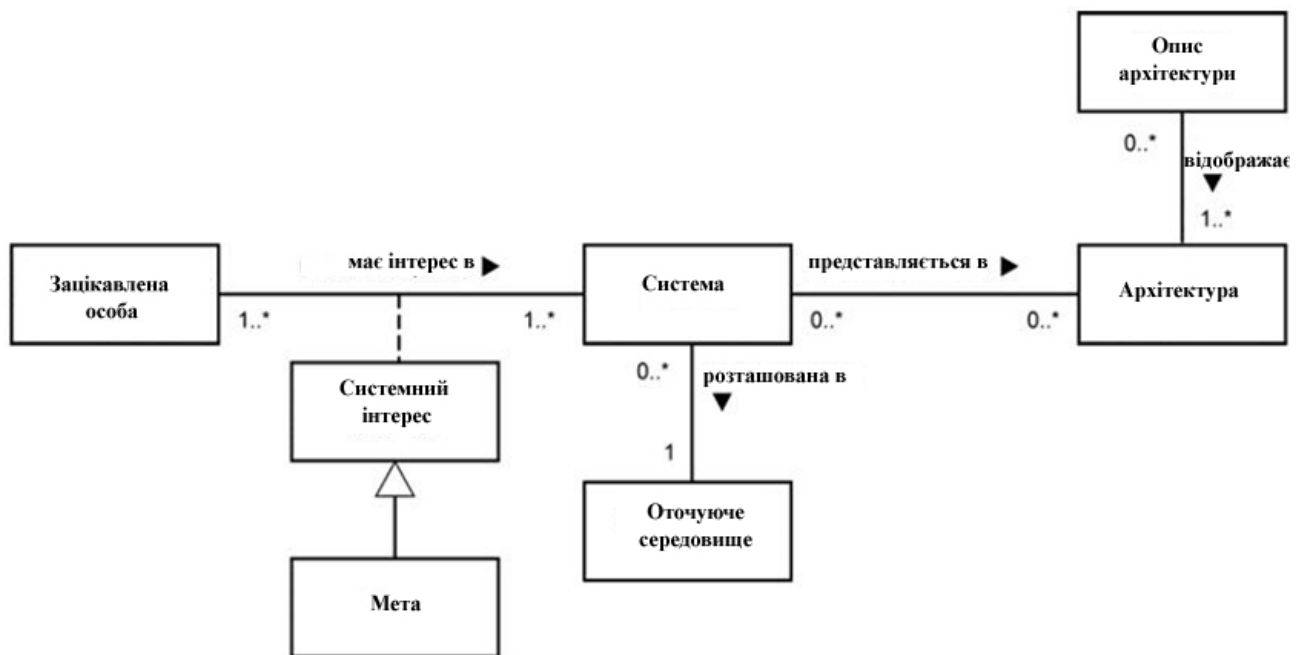


Рисунок 2.13 – Контекст опису архітектури стандарту ISO/IEC/IEEE 42010

Для структуризації певних інтересів системи в межах опису архітектури з різних точок зору використовуються мови опису архітектури (architecture description languages, ADL). Подібно структурі архітектури, ADL структурує певну множину інтересів для групи зацікавлених сторін, визначаючи один або більше видів моделі разом з іншими, пов'язаними аналітичними методами або інструментами. Подібно структурі архітектури або точці зору на архітектуру, ADL є ресурсом багаторазового використання, що не обмежується використанням у межах окремої системи або опису архітектури.

На сьогоднішній день розроблено велику кількість різних ADL з підтримкою широкого діапазону представлення моделей архітектур. Вони включають мови аналізу і опису архітектури, мови моделювання систем, мову ArchiMate та ін. Широке використання отримали сучасні ADL ArchiMate та SysML, узгоджені зі стандартом ISO/IEC/IEEE 42010 та з концептуальною моделлю, визначеною в цьому стандарті.

Мова ArchiMate впорядковує багаторівневі описи архітектур із позицій діяльності, застосунків, технологій або інфраструктури для структурних, поведінкових та інформаційних аспектів позицій у межах кожного з рівнів і

визначає вісімнадцять основних точок зору для рівнів архітектури. Кожна точка зору визначається через її власну метамодель, яка пов'язує різні точки зору між собою і визначеними зацікавленими сторонами, інтересами, цілями, шарами та аспектами.

Мова моделювання систем SysML створена на основі уніфікованої мови моделювання UML. Мова SysML визначає кілька типів структурних і динамічних діаграм: діяльності, послідовності, зміни станів, варіантів використання, визначення компонентів, внутрішніх компонентів, пакетів, параметричних діаграм і діаграм вимог. У термінах стандарту SysML кожен тип діаграми – це вид моделі. Мова SysML надає зацікавленим сторонам найважливіші конструкції для представлення цілей і видів таким чином, щоб забезпечити можливості створення нових точок зору відповідно до вимог цього стандарту.

2.4 Технологічний аспект побудови систем ситуаційного управління

Технологічний аспект побудови ССУ визначає технології та технологічні засоби холістичного представлення ССУ з різних точок зору, передбачених архітектурною моделлю системи. Технологічні засоби побудови конвергентних ССУ повинні відображати як структурну, інформаційну та процесну складові таких систем.

Для формального опису ССУ та задач, які розв'язуються, у процесі СУ використовуються функціональні, структурні, об'єктно-орієнтовані нотації. Для опису діяльності у ССУ використовуються процесні нотації, наприклад, BPMN, ARIS. Для опису архітектури ССУ використовуються архітектурні, функціональні, об'єктно-орієнтовані нотації.

Створення та використання ССУ реалізується в ході проєктної діяльності, яка визначається архітектурою і спеціальними вимогами предметної сфери СУ. Таким чином, проєктна діяльність щодо створення ССУ повинна поєднувати знання, які

стосуються опису архітектури організації, доступних технологій, спроможностей організації системи безпеки та вимог до конкретної реалізації ССУ. Спираючись на підходи до інженерії систем та управління життєвим циклом систем [13], проєкт щодо створення ССУ повинен передбачати розробку процесів управління діяльністю в рамках процесу управління життєвим циклом архітектури організації, в якій впроваджується ССУ, та власне процесів управління інформаційною безпекою.

Функціонування ССУ відбувається в умовах змінюваності ситуацій, пов'язаних із безпекою системи, в рамках моделі ситуаційного управління на основі перцептивного циклу когнітивної (пізнавальної) діяльності. Ситуативна змінюваність умов функціонування потребує відповідної ситуаційної адаптації архітектури ССУ до цих умов. Така адаптація може бути проведена на основі використання моделей знань предметної області у програмно-керованій архітектурі цільової системи в рамках реалізації концепції «архітектура як код» (architecture as code, AaC). Концепція AaC є розвитком та узагальненням концепцій програмного управління конфігураціями інформаційно-комунікаційних та програмно-насичених систем, яка отримала поширення в рамках технологій хмарних обчислень у вигляді IaC, IaaS, PaaS, SaaS тощо.

Побудова ССУ заснована на модельно-орієнтованому підході [235], [236], де прийнята архітектурна модель та модель знань предметної області визначають проєктні рішення.

Адаптований відповідно до положень стандарту ISO/IEC/IEEE 15288 [100] життєвий цикл ССУ включає такі стадії:

- задум;
- розробка;
- побудова (виробництво);
- використання та підтримка використання;
- модернізація;
- виведення з експлуатації.

Стадія формування задуму ССУ включає в себе процеси дослідження предметної сфери її застосування, визначення основних підходів до її розробки та

побудови. Знання, отримані на цьому етапі, повинні відображати специфіку виділеного (конкретного) тематичного домену предметної сфери у вигляді когнітивної мережі основних концептів. Семантика такої мережі може бути описана семантичною моделлю Кріпке (не плутати зі структурною моделлю Кріпке для недетермінованих скінченних автоматів) [237]:

$$\langle W, R, \models \rangle, \quad (2.6)$$

де $\langle W, R \rangle$ – шкала (фрейм) Кріпке на множині вузлів (світів) W із відношеннями R (множиною стрілок або упорядкованих пар) на $W : R \subset W \times W$, $\models$ – символ (відношення) істинності (оцінки, виконання). Наприклад, модальна формула $w \models A$ означає, що “ w задовольняється A ”, “ A виконується у w ” або “ w визначає (спонукає, викликає) A ”. Семантика Кріпке (реляційна семантика, фрейм-семантика) дозволяє створювати логічні моделі ситуацій у базисі інтуїціоністської (конструктивної) та модальних логік [20].

Після визначення семантики системи у вигляді моделі знань (2.4) проводиться розробка системи. Розробка ССУ як складної системи включає в себе такі групи процесів (рис. 2.14):

- аналіз потреб та моделювання;
- інженерія вимог (визначення та управління вимогами);
- побудова архітектури та затвердження.

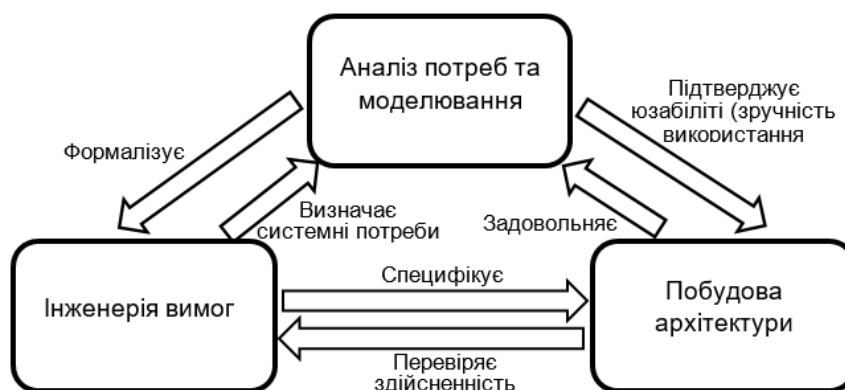


Рисунок 2.14 – Процеси розробки ССУ

Задум цільової системи та її ССУ представляється у вигляді переліку потреб та моделей, на основі аналізу яких проводиться формалізація вимог у відповідності з потребами. Виконання процесів інженерії вимог забезпечує визначення системних потреб та специфікацію архітектури ССУ у відповідності з архітектурою цільової системи та формалізованими моделями. Зокрема, одним із результатів інженерії вимог є затверджена політика безпеки цільової системи, що зберігається у базі знань (БЗ) ССУ. Побудова архітектури ССУ специфікується формалізованими вимогами до системи з БЗ ССУ та підтверджується результатами аналізу на основі моделей системи та тестових експлуатаційних задач.

2.6 Висновки до розділу 2

Побудова конвергентних ССУ ґрунтується на інформаційному, архітектурному та технологічному аспектах, реалізація яких забезпечує задоволення потреб цільової предметної області з урахуванням динаміки функціонування ССУ у процесі реалізації життєвого циклу системи або системи систем. Процес ситуаційного управління представлений як процес трансформації інформації про стан середовища у інформацію для керування станом середовища. Динаміка трансформації інформації відображена у запропонованій моделі.

Основою для створення моделі конвергентної системи є модель знань предметної сфери СУ. Конвергенція частинних моделей у загальній моделі знань СУ забезпечує методологічний базис для побудови ССУ цільового призначення. Створення систем підтримки цілеспрямованої діяльності, заснованих на знаннях, вимагає врахування особливостей неформалізованих або слабоформалізованих знакових систем, які використовуються для представлення знань. Одним з універсальних засобів представлення знань у цілеспрямованих системах є апарат теорії патернів. Умовою конкретизації семантики патернів у межах семантичних

доменів є визначення контексту через середовище оточення патернів. Механізми управління цілеспрямованою діяльністю ґрунтуються на використанні шаблонів-патернів і коригуючих патернів, які визначаються у точках біфуркації. Основою інформаційного забезпечення ССУ є розвинута онтологічна модель трансформації інформації в процесі ситуаційного управління від отримання сигналів про події у цільовому середовищі до формування на основі цих сигналів раціональних (мудрих) рішень, які трансформуються у сигнали керування засобами впливу на цільове середовище.

Архітектурний аспект побудови ССУ ґрунтується на використанні еталонних архітектурних моделей для комп'ютерно-насичених систем. Архітектура конвергентної ССУ описується з холістичних позицій і поєднує різні точки зору на систему. Основною метою побудови архітектури є створення необхідних спроможностей як окремих конституентних систем у складі ССУ, так і емерджентних спроможностей ССУ у цілому.

Технологічний аспект побудови конвергентних ССУ стосується вибору адекватних засобів модельно-орієнтованої розробки систем, враховуючи особливості сфери ситуаційного управління і, зокрема, змінюваність конституентних систем у складі ССУ при реалізації різних етапів ситуаційного управління.

У результаті дослідження інформаційного, архітектурного та технологічного аспектів побудови конвергентних ССУ визначено їх взаємозалежність та основні способи застосування. Сформульовано умови побудови архітектури ССУ на основі аналізу семантики (контексту) ситуації в цільовій області управління та ситуаційних потреб у формуванні та підтриманні необхідних спроможностей системи з урахуванням ситуативних обмежень цих спроможностей.

Основні результати розділу опубліковані автором у працях [2, 6, 7, 13, 15, 19, 20, 25, 30–41, 44, 46, 56, 57].

РОЗДІЛ 3

ФОРМАЛІЗАЦІЯ АРХІТЕКТУРИ КОНВЕРГЕНТНОЇ ССУ

3.1 Формальна модель багаторівневої ССУ

Багаторівнева система включає такі основні рівні: апаратний (технічний), операційний, сервісний (утилітарний), технологічний; інформаційний (контентний), регламентний (методичний), організаційний. Зокрема, технологічний рівень представлений спеціалізованими (предметно-орієнтованими) сервісами, які забезпечують функціональність цільової багаторівневої предметно-орієнтованої системи, її зовнішній інтерфейс, режими доступу, реалізують логіку предметної сфери і алгоритми реалізації предметних функцій. Технологічні сервіси організовуються за ієрархією, представленою на рис. 3.1 [50].



Рисунок 3.1 – Ієрархія технологічних сервісів предметно-орієнтованої інформаційної системи

Різноманітність і початкова невизначеність множини сервісів, необхідних для реалізації цільової моделі предметної діяльності, вимагають забезпечення широкого

вибору функціональності сервісів, їх доступності, гнучкості та автономності. Виконання цих вимог може бути забезпечено на основі агентно-орієнтованого підходу. Таким чином, предметно-орієнтована інформаційна система може бути представлена як мультиагентна система (МАС).

Враховуючи викладені вище вимоги, формально багаторівневу систему можна визначити кортежем [17]

$$S = \langle L, I^0, C^S, T \rangle, \quad (3.1)$$

де $L = \{l_1, l_2, \dots, l_n\}$ – множина функціональних рівнів системи, $I^0 = \{i_1^0, i_2^0, \dots, i_n^0\}$ – множина зовнішніх інтерфейсів системи, $C^S = \{c_1^s, c_2^s, \dots, c_n^s\}$ – множина функцій управління системою в цілому, T – цільова функція (функція призначення) системи (може бути задана неявно як метафункція у вигляді сукупності вимог до інших компонентів системи).

Кожний рівень, у свою чергу, може бути представлений кортежем

$$l_j = \langle U^l, I^l, C^a, C^l \rangle, \quad (3.2)$$

де U^l – множина сервісів рівня l_j , I^l – множина інтерфейсів рівня l_j , C^a – множина функцій управління додатками рівня l_j , C^l – множина функцій управління рівнем l_j .

Кожний сервіс u_k^l реалізується як композиція функцій деякої підмножини F_k^l множини функцій F^l , які реалізовані на рівні l_j .

Реалізація описаної моделі включає два аспекти:

- семантичний (змістовний);
- архітектурний (структурно-функціональний).

Семантика кожного рівня природно описується засобами семантичних мереж, які можна використовувати для представлення онтологій. Онтологічний підхід

широко застосовується для побудови змістовних моделей різних предметних областей. Для складних предметних областей застосовуються багаторівневі онтології. Онтологія, що має n рівнів, описується такою послідовністю [17]:

$$O = (O_n, \{<O_n, O_{n-1}>\}, \dots, \{<O_n, O_{n-1}, \dots, O_l>\}). \quad (3.3)$$

Тут O_n – модель онтології рівня n , $\{<O_n, O_{n-1}>\}$ – модель онтології рівня $(n-1)$, ..., $\{<O_n, O_{n-1}, \dots, O_l>\}$ – модель онтології першого рівня. Слід зауважити, що онтологія O є складовою частиною загальної математичної моделі предметної області, яка включає також модель знань предметної області $\{<O, K>\}$ та модель реальності предметної області $\{<O, K, R(K)>\}$.

Аналогічний підхід може бути застосований і для створення семантичної моделі багаторівневої інформаційної системи. Семантика кожного рівня буде визначати його функціональність і, певною мірою, структуру.

Спираючись на онтології, можна конкретизувати вимоги до реалізації структури системи. Враховуючи різноманітність задач, що вирішуються, їх варіативність та еволюціонування, необхідно забезпечити гнучкість і масштабованість багаторівневих систем. Це можна реалізувати на основі організації віртуальних структур у межах певної універсальної інформаційної метаструктури, зокрема, на основі МАС.

Розширенням концепції МАС для реалізації багаторівневих систем є визначення множини агентів кожного рівня, виходячи з функціональних вимог до цього рівня. Віднесення того чи іншого агента до певного рівня визначає тип агента. Таким чином, багаторівневу МАС формально можна описати як множину A агентів різного типу:

$$A = \{A^l\}, \quad (3.4)$$

де A^l – множина агентів, що відповідає рівню l .

Виходячи з семантики і функціональних вимог до кожного рівня, множина агентів A^l рівня l визначається відображенням семантики (онтології) O^l відповідно до рівня його функціональності F^l :

$$A^l : O^l \times F^l. \quad (3.5)$$

У результаті набір сервісів U^l кожного рівня буде визначатись функцією управління C^l агентами відповідного рівня:

$$U^l = C^l(A^l). \quad (3.6)$$

Слід зауважити, що деякі рівні багаторівневої системи можуть представлятись не лише програмними агентами, а й іншими об'єктами і суб'єктами системи, наприклад, адміністраторами, розробниками курсів, керівниками занять, особами, які навчаються, регламентами функціонування та ін., що мають іншу фізичну сутність (неелектронну), але включені у контури управління і взаємодії агентів системи. До таких рівнів можуть відноситись, наприклад, організаційний і методичний рівні.

3.2 Модель ситуаційного агента

Агенти в МАС повинні взаємодіяти у відповідності із процесами обробки інформації та управління, які застосовуються в інформаційній системі. Моделі процесів обробки інформації та управління зберігаються у базі знань (БЗ) інформаційної системи, яка поповнюється і оновлюється у процесі її функціонування із врахуванням попереднього досвіду використання. Таким чином, при функціонуванні інформаційної системи на основі МАС забезпечується її

еволюціонування. БЗ багаторівневої інформаційної системи являє собою ієрархію доменів знань про різні аспекти процесів її функціонування, які включають як знання, що стосуються конкретних сфер, в яких здійснюється використання інформаційної системи, так і знання, пов'язані з регламентами роботи в кожній із цих сфер. Ці знання використовуються для визначення переліку потрібних агентів та моделей їх поведінки і взаємодії у процесі функціонування інформаційної системи [51], [47], [238], [239], [240], [241], [242].

Враховуючи ієрархію знань, можна класифікувати агенти МАС за їх призначенням і використанням. Тому для управління вибором конкретної множини агентів для реалізації тих чи інших процедур в МАС слід провести відбір агентів у дану множину. Для цього використовуються знання про агенти та їх властивості із БЗ доступних агентів.

Враховуючи особливості застосування МАС, реалізація кожного етапу даного процесу підтримується відповідними організованими групами («ансамблями») агентів таким чином, що результат діяльності однієї колонії є вхідним матеріалом для діяльності іншої, і в кінцевому підсумку отримується остаточний результат процесу обробки інформації у заданому форматі [78]. Отже, обробка інформації в багаторівневій МАС є результатом суперпозиції функціонування множини колоній агентів на заданому інформаційному полі. Таким чином, постає задача формального визначення специфікацій задач предметної області використання МАС, на основі якого визначається початковий склад «екосистеми агентів» для конкретної задачі обробки інформації і управління. На основі специфікацій визначається склад кожної колонії, який обирається з «популяції» подібних між собою агентів.

Агент-диспетчер МАС повинен мати опис загальної формальної постановки задачі обробки інформації у вигляді запиту до моделі знань відповідної предметної сфери, на основі якого здійснюється підбір множини необхідних агентів, що реалізують кожний етап вирішення даної задачі. Агенти функціонують у межах агентної платформи, яка забезпечує управління життєвими циклами агентів і ефективними комунікаціями (обміном повідомленнями) між агентами.

При конфігуруванні середовища застосування багаторівневої МАС агенти реалізації етапів процесу обробки інформації здійснюють формування множини агентів реалізації процесів, агенти реалізації процесів – формування множини агентів реалізації процедур і далі до визначення множини необхідних транзакцій для кожної з операцій обробки інформації. При цьому на кожному рівні формується лише інформаційна база про агентів без їх активації та зв'язування у процесі взаємодії. Таким чином формується інформаційне агентне середовище. Далі, у відповідності з технологією використання багаторівневої МАС, здійснюється активація потрібних агентів.

Виходячи з різноманітності задач, які вирішуються в різних процесах обробки інформації, мультиагентне середовище обробки інформації і управління будується на основі гібридної архітектури, тобто включає в себе деліберативні (засновані на знаннях) та реактивні (з механізмами поведінки типу «стимул-реакція») агенти. Деліберативні агенти призначені для виконання аналітичних функцій, результатом яких є конкретизація предметного середовища використання МАС. Реактивні агенти реалізують управління процесом обробки на основі інформації (подій), отриманої із зовнішнього середовища або інших агентів у результаті здійснення комунікацій. Таким чином формуються рольові групи (асоціації) агентів із спеціалізацією функцій в середині груп як по вертикалі (рівнями ієрархії функцій), так і горизонталі (задачами, що вирішуються на даному функціональному рівні).

Формування знань агентів про своє середовище здійснюється з використанням механізмів монотонної і немонотонної логіки, що дозволяє сформувати більш повну модель світу з урахуванням можливих обмежень та специфіки предметної сфери, наближену до картини світу, яка формується в уяві людини. Автономність агентів дозволяє створювати подібні агенти в різних місцях агентного середовища і надалі здійснювати відбір найбільш ефективних агентів за визначеними показниками у різних моделях взаємодії (конкуренції або кооперації). Таким чином забезпечуватиметься безперервність функціонування, гнучкість та еволюційність мультиагентного середовища ситуаційного управління.

При реалізації багаторівневих систем на основі віртуальних структур потрібно забезпечити як мінімум достатність ресурсів підтримуючої (базової) метасистеми. Важливими задачами залишаються також задачі забезпечення функціональної повноти і мінімальної достатності функцій системи.

Використання агентно-орієнтованого підходу при створенні багаторівневих інформаційних систем забезпечує гнучкість, адаптованість, простоту масштабування і удосконалення таких систем.

Організація агентів на основі ієрархії функцій технологічного процесу обробки інформації в інформаційній системі дозволяє здійснювати ефективне управління мультиагентним середовищем, виходячи з потреб кожного рівня, які забезпечуються лише необхідною множиною агентів нижчих рівнів.

Управління мультиагентним середовищем здійснюється агентом диспетчером на основі бази знань про доступних агентів, зареєстрованих на платформі, з організацією рольових груп.

На основі моделі багаторівневої архітектури запропоновано підхід до побудови мультиагентних інформаційних систем, що забезпечує узгодження семантики, функціональності та підтримуючих їх архітектурних рішень, і дозволяє реалізувати наскрізний процес проєктування таких систем. Запропонована формалізація процесу проєктування є уніфікованою і може бути підтримана будь-якими відомими технологіями моделювання систем. Крім того, така модель природним чином представляє різні рівні абстракції при розгляді інформаційних систем, водночас зберігаючи змістовний зв'язок між цими рівнями, що забезпечує коректність переходу від одного рівня абстрактного подання до іншого, без втрати семантичної єдності всіх рівнів.

Означення. Ситуаційним агентом будемо називати цілеспрямований суб'єкт предметної області, що характеризується моделлю раціональної поведінки на основі усвідомлення, аналізу та прогнозування наслідків поточної ситуації.

Отже, метою ситуаційного агента є вплив на процеси ситуаційного середовища з метою зміни його стану у відповідності з уявленнями (знаннями, переконаннями) цього агента (рис. 3.2) [6].

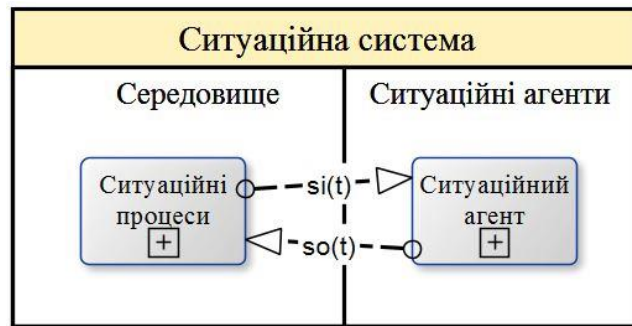


Рисунок 3.2 – Взаємодія середовища і ситуаційного агента

Ситуаційного агента можна розглядати як варіант «розумного» агента (smart agent) за класифікацією Н.С. Nwana [243]. Поведінка ситуаційного агента визначається моделлю трансформації інформації I-SDKW. Спираючись на модель BDI (переконання-прагнення-наміри) [244] поведінки агента, можна описати модель ситуаційного агента, що формує раціональну поведінку у відповідності з описаною I-SDKW-моделлю трансформації інформації.

Модель практичного міркування BDI була запропонована М. Братманом [245] як спосіб пояснення намірів, орієнтованих на досягнення поставлених цілей у майбутньому. В основі моделі BDI лежить так звана фолк-психологія (народна психологія, теорія теорії, теорія здорового глузду) [246], що відображає ментальні моделі світу (оточуючого середовища) у формі теорій. Модель BDI було покладено в основу розробки моделі BDI програмування поведінки розумних агентів. Поведінка ситуаційного агента визначається моделлю трансформації інформації I-SDKW. Поведінкова модель ситуаційного агента представлена на рис. 3.3 [6].

Далі, спираючись на модель BDI (beliefs, desires, intentions) поведінки агента, можна описати модель ситуаційного агента, що формує раціональну поведінку у відповідності з описаною I-SDKW-моделлю трансформації інформації.

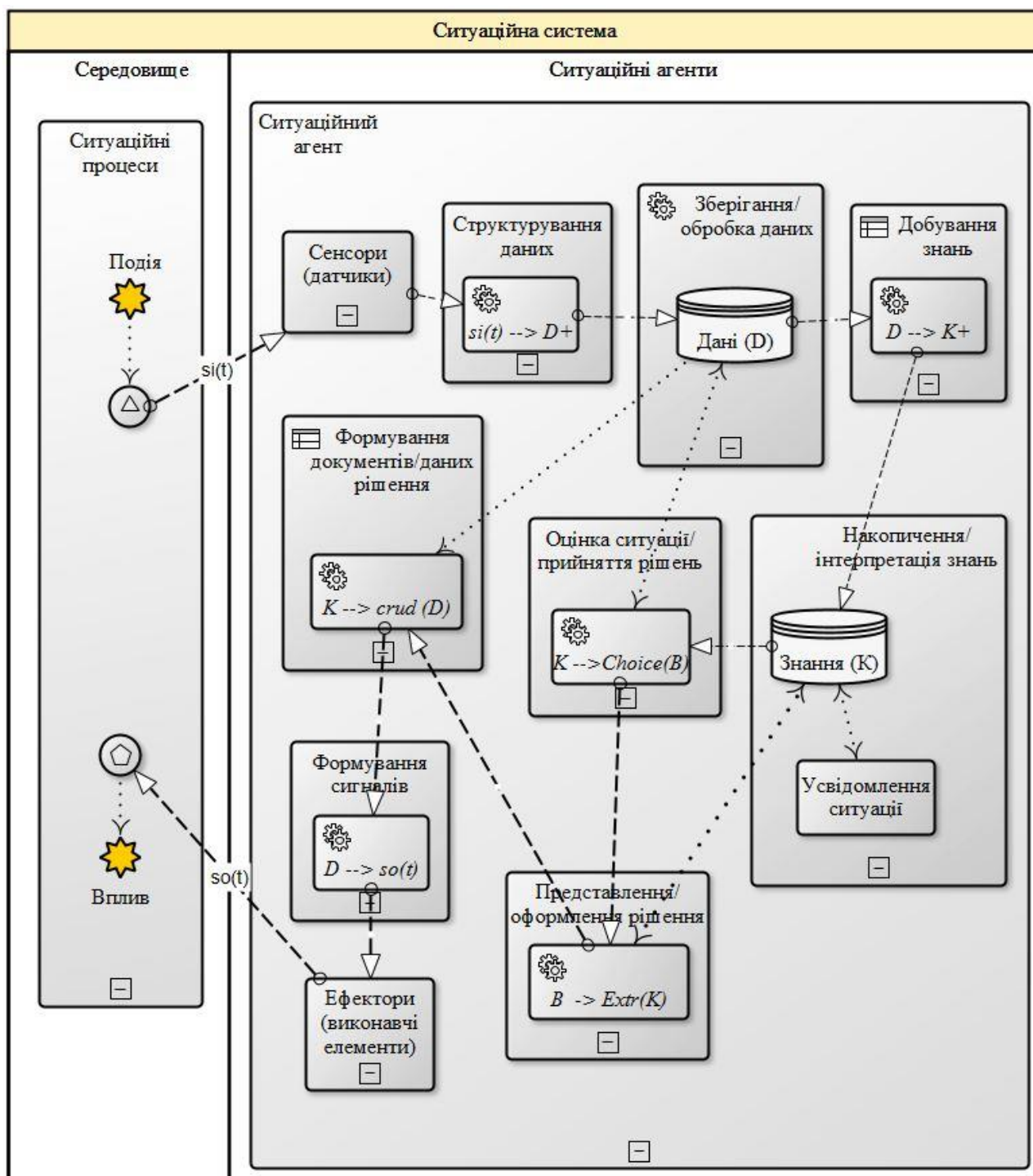


Рисунок 3.3 – Поведінкова модель ситуаційного агента

Фолк-психологія – це термін, який широко використовується в філософії свідомості і когнітивній науці, а також у науковій літературі у трьох різних контекстах:

1) для позначення певної сукупності когнітивних здібностей, які включають у себе здатність передбачати і пояснювати поведінку;

2) для позначення теорії поведінки, що обґрунтовує сукупності когнітивних здібностей, зокрема, здатність передбачати і пояснювати поведінку;

3) для позначення ненаукової психологічної теорії, що складається з поширених простонародних трактувань психологічних термінів, поведінки і особистісних якостей.

Фолк-психологія має важливе значення для оцінки та остаточного розуміння нових ситуацій в оточуючому середовищі. Зокрема, на основі контекстних ментальних моделей (контекстної моделі знань) у формі прототипу та зразкових уявлень (переконань) індивіди (агенти) здатні точніше представляти та усвідомлювати ситуації в навколишньому середовищі. Відповідно до контекстної моделі знань, загальна схожість між прототипом та заданим екземпляром категорії оцінюється на основі декількох точок зору (тобто вимірів: форми, розміру, кольору тощо). Мультиплікативна функція, яка моделює явище, має вигляд

$$s(P, E_i) = \prod_k s(P, E_{ik}) \quad (3.7)$$

Тут $S(P, E_i)$ – схожість між прототипом P та i -ми екземплярами, k – кількість вимірів (точок зору), $K = 1...k$, $S(P, E_{ik})$ – подібність між прототипом та i -м зразком у k -му вимірі.

При описі поведінки «розумних» агентів слід також враховувати їх обмежену раціональність, пов'язану із проблемою вибору, обмеженнями знань, доступними ресурсами тощо. Обмежена раціональність, яка є основою математичного моделювання при прийнятті рішень, використовується як певна оптимізація, коли прийняття рішень розглядається як цілком раціональний процес пошуку оптимального вибору з урахуванням наявної інформації. У роботі [247] описано моделі раціональності, які можуть набути більшої реалістичності, але з дотриманням достатньо строгої формалізації. Моделі раціональної поведінки більш широкого застосування (глобального типу), що будуються зазвичай, так і

локального застосування (обмеженого типу), повинні включати в себе деякі або всі елементи з такого переліку.

1. Множина альтернатив поведінки (альтернативи вибору чи рішення).

2. Обмежена підмножина альтернатив поведінки, яка розглядається або сприймається агентом. Тобто агент може зробити свій вибір у рамках більш обмеженої множини альтернатив у порівнянні з усіма об'єктивно доступними йому альтернативами.

3. Можливі майбутні ситуації чи результати вибору. Тобто, на початку немає необхідності розрізняти фактичний та прийнятний результат.

4. Функція окупності, яка представляє цінність або корисність, що надаються агентом при кожному з можливих результатів вибору. Для спрощення можна вважати, що корисність результату вибору вже є визначеною.

5. Інформація щодо того, які результати із множини можливих результатів фактично будуть обрані, якщо буде обрана певна альтернатива із множини альтернатив рішень. Ця інформація може бути неповною, тобто для кожної альтернативи поведінки може бути більше одного можливого результату, а саме подається інформація про відображення кожного елемента з множини обраних альтернатив поведінки на підмножину можливих результатів.

6. Інформація щодо ймовірності того, що конкретний результат настане, якщо буде обрано певну альтернативу поведінки.

Г. Саймон запропонував при прийнятті рішень агентами використовувати евристичну модель раціональної поведінки замість жорстких правил оптимізації [247]. Вибір евристичної моделі поведінки пов'язаний зі складністю ситуації та неможливістю обробити й обчислити очікувану корисність кожної альтернативної дії. Крім оцінки витрат, часто буває потрібно врахувати інші економічні фактори, які впливають на прийняття рішень.

Г. Діксон запропонував процес міркування, який лежить в основі обмеженої раціональності [248]. Якщо припустити, що агенти обирають дію, яка наближена до оптимальної, то можна використати наближену ϵ -оптимізацію, тобто агент обирає дії так, щоб значення критерію оптимальності було в межах ϵ від оптимального.

Якщо визначити оптимальний (найкращий можливий) виграш як U^* , то набір ε -оптимальних параметрів $S(\varepsilon)$ можна визначити як усі варіанти $s: U(s) \geq U^* - \varepsilon$. Поняття строгої раціональності (оптимальності) буде тоді частковим випадком, коли $\varepsilon = 0$. Перевага такого підходу полягає в тому, що при його використанні немає необхідності детально визначати процес міркування, а оптимальний вибір знаходиться в межах ε .

У моделі Братмана прагнення і наміри вважаються мотивами, з яких випливає дія, але намір сприймається як керівний мотив вищого порядку. Він визначає поняття зобов'язання як фактора, що розділяє прагнення і наміри, з урахуванням, що, по-перше, зобов'язання встановлює тимчасову сталість у реалізації певного плану, і, по-друге, наступні плани будуються на основі тих, які агент вже прийняв як обов'язкові для виконання.

Модель програмування BDI частково визначає практичну реалізацію зазначених концепцій. Однак тимчасова сталість у розумінні безпосередньо минулого часу з моменту початку реалізації конкретного плану не розглядається. Ієрархічність планів відбивається досить простим способом: кожен план складається з набору кроків, на кожному з яких може бути ініційований черговий новий план. У такій ієрархії природним чином закладена описана вище тимчасова сталість, адже кожен новий план по суті є продовженням попереднього, а агент продовжує слідувати загальному плану відповідно до його наступних варіантів.

Важливим аспектом моделі BDI в теоретичному сенсі є наявність логічних моделей, за допомогою яких можна визначити і потім вивчати поведінку агентів. Так, наприклад, дослідження в даній області привели до аксіоматизації деяких реалізацій моделі BDI і появи описів на основі формальної логіки, таких як BDICTL [249], де об'єднано мультимодальну логіку (що описує модальності переконань, прагнень і намірів) і темпоральну логіку CTL. Пізніше М. Вулдрідж [250] розширив BDICTL до LORA (the Logic of Rational Agents (логіка раціональних агентів)), використовуючи логіку дій. LORA дозволяє розглядати не тільки поведінку окремих агентів, але і процеси взаємодій в мультиагентній системі, таких як передача інформації.

Отже, неформально семантику компонентів архітектури BDI-агента можна описати таким чином.

Переконання описують ступінь інформованості агента або його уявлення про навколишній світ (модель світу), включаючи себе самого та інших агентів. Переконання можуть також будуватися на основі міркувань (логічного виводу), що дозволяють виводити наслідки і створювати нові переконання. Поняття переконання можна трактувати як особливий вид суб'єктивних знань агента, які можуть містити не лише правильні (істинні) твердження, а й хибні в загально прийнятому розумінні, твердження, які агент суб'єктивно вважає правильними. Модель знань, яка представляє переконання, впливає на поведінку агента. Переконання зберігається у частині бази знань, яку називають базою переконань, набором переконань, вірою.

Прагнення (бажання) відображають стан мотивацій агента. Вони подаються у формі цілей або ситуацій, яких хотів би досягти агент. Прикладами таких прагнень можуть бути: знайти кращу роботу, відпочити, забезпечити захист. Цілі є прагненнями, прийнятими агентом до виконання. Множина цілей є підмножиною активних прагнень агента, що складається з ситуаційно сумісних (здійснених, несуперечливих) елементів. Так, наприклад, агент не повинен мати своїми цілями одночасно бажання захистити систему і зламати систему або попрацювати і відпочити.

Наміри відображають усвідомлений вибір плану дій агентом для реалізації прагнень, які агент вважав за краще виконувати. Наміри є підмножиною прагнень, поставлених агентом у план обов'язкового виконання. План – це послідовність елементарних дій (алгоритм, процедура або фрагмент знань), які агент здатний виконати для того, щоб здійснити один або кілька із своїх намірів. Плани можуть бути вкладеними, тобто типовий план «відкрити двері» може припускати наявність вкладених планів «дістати ключі», «вибрати потрібний ключ». Це відображає особливість моделі Братмана, яка полягає в тому, що плани спочатку замислюються лише частково (в загальному), а деталізуються у процесі їх просування з урахуванням поточної ситуації.

Зовнішнім мотиватором (сигналом) для відповідної реакції агента є подія. Зокрема, подія може привести до оновлення (перегляду) переконань, зміни цілей або до вибору нового плану. Події генеруються (виникають) у зовнішньому середовищі і сигнали, пов'язані з подіями, обробляються агентом (рис. 3.3). Агент може також самостійно генерувати події, необхідні для реалізації власного життєвого циклу.

Представляючи компоненти архітектури BDI-агента у вигляді відповідних наборів концептів бази знань запропонованої моделі, можна вважати BDI-агентів окремим випадком ситуаційних агентів.

Статистика використання інформаційної інфраструктури показує, що в умовах мінливого завантаження (мінливості частоти використання різних додатків) вона використовується нерівномірно – ресурсів інфраструктури або не вистачає, або вони простоюють. З іншого боку, різноманітність і мінливість корпоративної бізнес-логіки обумовлює необхідність використання широкого спектра програмних продуктів. Ці фактори визначають досить високу сукупну вартість володіння інформаційних систем.

Таким чином, завдання підвищення ефективності використання інформаційних систем може бути сформульована як максимізація коефіцієнта використання наявної корпоративної інфраструктури і обчислювальних ресурсів із можливістю залучення, при необхідності, зовнішніх ресурсів. Підвищення ефективності використання корпоративної інформаційної інфраструктури при організації обробки інформації досягається на основі сервіс-орієнтованого підходу і, зокрема, з використанням хмарних технологій, коли необхідні ресурси надаються у вигляді хмарних сервісів.

Процес ситуаційного управління в ССУ являє собою складну інформаційну технологію, на різних етапах якої здійснюються процедури обробки даних різного характеру з застосуванням відповідного програмного забезпечення.

З точки зору характеру вирішуваних завдань (функціонального аспекту), можна запропонувати таку класифікацію додатків, що використовуються в корпоративних технологіях, у тому числі в технологіях ситуаційного управління:

- підтримка колективної роботи різних груп користувачів;

- комунікаційні;
- навчальні;
- орієнтовані на дані та знання;
- обчислювальні;
- мультимедійні;
- забезпечення безпеки;
- загальносистемні і керуючі.

Таким чином, для реалізації кожного етапу технології ситуаційного управління використовуються відповідні набори програмних додатків. Відповідно до виду окремого додатка до нього висуваються певні вимоги щодо ресурсів і підтримуючої інфраструктури корпоративних інформаційних систем.

У технологічному процесі ситуаційного управління реалізуються такі етапи:

- аналіз проблеми, пов'язаної з ситуаційним управлінням;
- підготовка інформації, необхідної для ситуаційного управління;
- підбір і підготовка колективу для здійснення ситуаційного управління;
- підготовка та налаштування інформаційного середовища ситуаційного управління;
- реалізація процедур діяльності відповідно до цілей ситуаційного управління;
- документальне та нормативне забезпечення прийнятих управлінських рішень;
- організація реалізації прийнятих управлінських рішень;
- аналіз результативності та якості ситуаційного управління зі збереженням результатів аналізу в базі знань.

Незалежно від декомпозиції конкретної складної системи на підсистеми при проектуванні ССУ, спочатку проводиться макропроектування та конструювання, а потім внутрішнє конструювання і проектування (структурне та функціональне проектування). На етапі макропроекту розробляється узагальнена модель складної системи та відповідних процесів, що дозволяє розробнику отримати відповіді на питання про ефективність різних стратегій управління об'єктами, коли ці об'єкти взаємодіють із середовищем у рамках застосування ССУ. Цей етап можна розбити

на етапи аналізу та синтезу. В ході аналізу вивчається об'єкт управління, будується модель впливу на навколишнє середовище, визначаються критерії оцінки ефективності використання наявних ресурсів та виявляються іманентні (внутрішньо притаманні) та необхідні обмеження. Кінцевою метою етапу аналізу є побудова моделі управління об'єктом та визначення її оцінюваних характеристик для вивчення конкретних завдань ситуаційного управління та їх вирішення. Під час синтезу на стадії зовнішнього проектування вирішуються завдання вибору стратегії управління, які ґрунтуються на вивченні та дослідженні моделі об'єктів управління, які є складними, комплексними системами.

Реалізація ССУ у вигляді мультиагентної системи (МАС), завдяки притаманній їй характеристиці, є адекватним підходом до вирішення задач СУ. МАС може бути використана як допоміжний інструмент для організації та підтримки інтегрованих середовищ СУ. Такий тип середовищ СУ будемо називати агентно-орієнтованими середовищами ситуаційного управління (АОССУ).

Ефективна ССУ повинна інтегрувати в собі нові інформаційні технології, можливості людини, останні досягнення в області прийняття рішень та на їх основі дозволяє здійснювати колективне прийняття рішень, аналіз і прогнозування їх результатів. Створення ефективних ССУ пов'язане з конвергенцією знань, інтелектуальних та професійних спроможностей залучених до СУ осіб із можливостями, які надаються інтелектуалізованими комп'ютерними засобами. Отже, основою АОССУ є конвергентні ситуаційні агенти, що розглядаються як цілісні компоненти ССУ, здатні вирішувати задачі різних рівнів на всіх етапах СУ.

Однією з умов успішної діяльності будь-якої організаційної структури, в тому числі ССУ, є ефективне управління, яке враховує специфіку її операційної діяльності. При СУ в рамках ССУ реалізуються такі етапи:

- Моніторинг об'єкта управління.
- Визначення учасників (аналітики, експерти, особи, які приймають рішення) процесів ситуаційного управління.
- Системний аналіз ситуації (аналіз поточного та прогнозування очікуваних станів об'єкта управління, моделювання еволюції подій в об'єкті управління тощо).

- Формування та прийняття рішень шляхом нарад та обговорень.
- Реалізація рішень (визначення виконавців, вироблення вказівок та моніторинг виконання прийнятих рішень).
- Ретроспективний аналіз (оцінка) якості та ефективності виконання прийнятих рішень.

При здійсненні процесу ситуаційного управління на кожному етапі в АОССУ використовуються різні типи агентів. Конвергентні агенти на різних рівнях виконують функції диспетчеризації, координації, забезпечення безпеки, пошуку та надання доступу до сервісів тощо. Тобто вирішення задач різних стадій та етапів СУ виконується відповідними групами конвергентних агентів в АОССУ з урахуванням особливостей кожного етапу. Конвергентні агенти виступають як активні елементи, що забезпечують інтеграцію та адаптацію сервісів для вирішення конкретних задач ситуаційного управління. Взаємодія між агентами визначається моделлю управління, прийнятою в ССУ. Моделі агентів для кожного етапу СУ повинні враховувати аспекти комунікації та координації – формалізацію опису ситуації, паралельне управління процесами, механізми оркестрування та хореографії, обмін повідомленнями тощо.

3.3 Трансформація знань у багаторівневій ССУ

Однією з основних задач при створенні інтелектуальних систем є моделювання процесів мислення, пізнання та навчання. Оскільки розумова діяльність пов'язується з людиною, то й створення штучних інтелектуальних систем іде шляхом моделювання та імітації аналогічних процесів у людині. Визначний прогрес у дослідженнях процесів і створенні теорій, пов'язаних із розумовою діяльністю людини, було досягнуто впродовж останніх ста років, хоча ці питання цікавили ще античних філософів, зокрема, Платона і Арістотеля. Здатність до мислення, пізнання, абстрагування, формування системи знань та використання їх у

цілеспрямованій діяльності для зміни та пристосування оточуючого середовища стала найважливішою конкурентною перевагою людини у процесі еволюції.

Знання представлені взаємопов'язаними практичною і теоретичною складовими, через які реалізуються основні функції знань: пізнавальна (гносеологічна), онтологічна (інтерпретаційна, сутнісна), прогностична, твірна (синтетична). Платон вважав, що людина вже вміщує в собі увесь можливий обсяг знань і лише шлях інтелектуальних обчислень та суджень відділяє людину в розумово-незрілому стані від людини, що володіє значними обсягами інформації. Зокрема, Платон визначає процес виділення знання на основі постійного міжособистісного контакту (комунікацій, контактів, транзакцій). Наступним кроком у теорії пізнання була ідея І. Канта про три рівні створення знання: відчуття, інтерпретація відчуттів (підведення під окремі категорії усього здобутого простими даними чуттів), інтелектуальні (логічні) судження на основі гіпотез і ситуацій. Значний склад у феноменологічні аспекти пізнання було зроблено Е. Гуссерлем, зокрема, ним була створена теорія знаку. За Гуссерлем пізнання носить характер інформаційних актів, що дозволяє визначити ноєму, – відоме ядро, яке разом із множиною змінних характеристик утворює різноманітні модифікації даного знання. На основі теорії Е. Гуссерля П. Прехтль зробив припущення, що істинним може бути те, що є цінним у житті. Цим феноменом можна пояснити обмеженість, відносність і нескінченність пізнання. Важливим доповненням до феноменології пізнання є ідея логіцизму, сформульована Ф.Л.Г. Фреге. Зокрема, ним була визначена відмінність між змістом і значенням і запропонований семантичний трикутник, що пов'язував знак, зміст і значення. В системі уявлень Ф.Л.Г. Фреге значення – це іменована предметна область, а зміст – певний аспект розгляду цієї предметної області.

Базовим інструментом представлення, накопичення, передачі та використання знань є мова, в тому числі її формальна семіотична модель. Важливі у семіотиці результати були отримані Ф. де Соссюром та Ч.С. Пірсом. Зокрема, Ф. де Соссюром було введено дихотомічну концепцію знаку – визначник-визначальне. Мова за Ф. де Соссюром являє собою систему значимостей. На відміну від значення значимість

знаку виникає з його відношень із іншими знаками мови. Відповідно виділяють два види значимостей: синтагматичні відношення і асоціативні відношення. В розвиток ідей Ф. де Соссюра Ч.С. Пірс визначив предмет семіотики як вчення про природу і основні різновиди знакових процесів та створив базову семіотичну класифікацію знаків: знаки-образи (icon), знаки-індекси (index), знаки-символи (symbol). Сучасна семіотика була сформована Ч.У. Моррісом, який систематизував семіотику і виділив основні її розділи: синтактику, семантику і прагматику [251]. Семіотику відносять до метанаук.

Поняття синтаксису застосовується в різних контекстах (предметних сферах). Синтаксис у семіотиці вивчає сукупність відношень між знаками. Прагматика у межах семіотики досліджує відношення між знаковими системами та їх користувачами. При створенні штучних інтелектуальних систем правильний вибір або створення знакової системи дозволяє структурувати процес комунікацій і надати йому адекватності (змістовності) й передбачуваності реакцій учасників комунікацій на різні повідомлення (знаки). Знакова система забезпечує однозначність інтерпретації (розпізнавання) і трактування знаків-повідомлень, якими обмінюються учасники комунікацій.

Семантика в семіотиці вивчає зміст мовних конструкцій виходячи із відношень між визначниками мови – словами, словосполученнями, знаками і символами. Предмет семантики визначається її предметною сферою. Для створення штучних інтелектуальних систем важлива формальна семантика, що надає формальні математичні засоби для опису семантичних моделей та інтерпретації формальних і природних мов. Формальна семантика визначає різні підходи до опису семантики формальних систем, найбільш поширеним з яких є модельно-теоретична семантика. Модельно-теоретична семантика ґрунтується на теорії істинності А. Тарського [252] і полягає в тому, що зміст частин твердження задається різними способами рекурсивного визначення групи функцій інтерпретації на попередньо задані математичні множини. Крім того, А. Тарським був запропонований механізм вирішення семантичних парадоксів на основі використання метамови, причому для кожної метамови може бути визначена метмова наступного, вищого рівня.

Множина метамов для усунення семантичних парадоксів мов нижчих рівнів дістала назву «сходи Тарського».

При побудові штучних інтелектуальних систем використовуються також положення основних складових сучасної математики – теорії доведення, теорії моделей, аксіоматичної теорії множин, теорії рекурсивних обчислень, причому теорія доведення представляє синтаксичний аспект, а теорія моделей – семантичний.

Таким чином, при створенні й функціонуванні інтелектуальних систем, орієнтованих на знання, слід враховувати описані вище особливості використання знакових систем, які є характерними для природних інтелектуальних систем.

Крім описаних особливостей пізнання і трансформації інформації із зовнішнього середовища, в інтелектуальних системах не менш важливим є активний аспект – цілеспрямований вплив штучних систем на оточуюче середовище у процесі свого функціонування з урахуванням поточного стану «розвитку» («навченості») системи. Актуальними у цьому контексті стають процедури визначення поточного стану інтелектуальної системи та визначення адекватності її реакції на зовнішні збудники. Отже, процес функціонування інтелектуальної системи може бути описаний такою послідовністю [57]: реальність – сприйняття – мова – інтерпретація (пояснення) – пізнання (розширення) – знання (систематизація, модель) – трансформація – навчання (створення моделі) – застосування моделі – вироблення навичок – свідоме/несвідоме відтворення моделі.

Під час функціонування ССУ її агенти використовують знання, які відповідають контексту ситуації. Знання агента є фрагментом області знань для цільової проблеми СУ. Знання проблемної області СУ є елементом моделі СУ. Управління знаннями ССУ є одним із ключових факторів ефективного СУ. Отже, знання є визначальним ресурсом, а управління знаннями є важливим компонентом функціонування ССУ. Модель предметної області знань визначається архітектурною моделлю ССУ та її контекстом взаємодії з оточуючим середовищем. Специфікою управління знаннями при СУ є наявність двох областей знань. Перша область пов'язана з підтримкою функціонування ССУ, а друга – безпосередньо із проблемною сферою СУ, на яку направлено вплив ССУ.

Модель знань організації відображає специфіку кожного з її компонентів і характер взаємозв'язків між ними. Модель знань ССУ представляє собою окрему область у системі знань організацій. Таким чином, побудова моделі знань із урахуванням специфічних особливостей діяльності ССУ потребує додаткових досліджень та уточнення. Особливе місце в архітектурі організації (підприємства) займають корпоративні знання і технології управління знаннями. Корпоративні знання є частиною архітектури організації і визначають його місію, цілі, завдання, процеси діяльності (бізнес-процеси), організаційну структуру і розглядаються в контексті взаємодії з навколишнім середовищем. Таку систему знань організації будемо називати архітектурою знань. Управління знаннями ґрунтується на архітектурі підприємства, його місії тощо. Елементи організаційної структури забезпечують виконання місії через композицію мотивацій, знань (здібностей, навичок), ресурсів (або обмежень) і спроможностей.

З урахуванням змінюваності та проактивності знань, а також їх розподілу на явні (формалізовані) та неявні (суб'єктивні, неформалізовані) знання, носіями яких можуть бути окремі агенти, групи агентів або організації, Ікуджіро Нонакою була запропонована модель трансформації корпоративних знань між рівнями носіїв знань [253], [254] [255]. Така модель представлена на рис. 3.4. Архітектура ССУ забезпечує реалізацію процесу СУ як композицію мотивацій, знань, спроможностей, ресурсів та обмежень.

Модель трансформації знань І. Нонаки отримала назву СЕКІ-модель відповідно до назв механізмів трансформації неявних знань у явні і, навпаки, на різних рівнях корпоративної архітектури. Механізми інтерналізації забезпечують підтримку засвоєння знань окремими індивідами (суб'єктами). Механізми соціалізації забезпечують формування колективних (групових) знань стосовно цільової області зацікавленості групи людей, зокрема, в рамках малих експертних груп. Механізми екстерналізації забезпечують поширення знань цільової області, сформованої групою експертів у середині організації. Механізми композиції (комбінації, конвергенції) забезпечують застосування знань цільової області, які

усвідомлені і засвоєні персоналом організації, при здійсненні організацією цілеспрямованої діяльності.

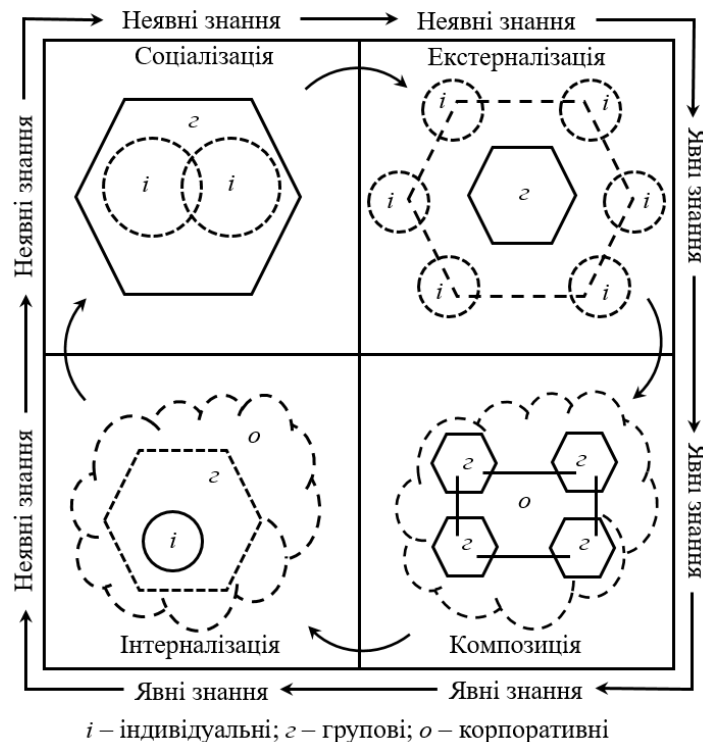


Рисунок 3.4 – СЕКІ-модель трансформації корпоративних знань у кіберматичних системах

Така модель дозволяє виділити формалізовану і неформалізовану складові знань і пов'язати їх із природними або штучними (технічними) носіями, які на основі конвергенції компонентів цих знань здатні формувати цілеспрямовану ситуаційну поведінку у процесі СУ.

Система ситуаційного управління є специфічним типом організації. Тому виникає проблема визначення структури формальної моделі, її представлення, впровадження та використання у практиці організації з урахуванням особливостей створення та функціонування ССУ з використанням сучасних методів формалізації.

Основними рисами організаційної моделі є механізм координації, ключова частина організації, проєктні параметри (тип децентралізації), ситуаційні фактори (мотивації).

Формальна модель діяльності W ССУ базується на визначенні моделі категорії M_K , параметрів організаційної моделі M_O , архітектурної моделі M_A , процесної (функціональної) моделі M_F , логічної моделі (включаючи модель модальності) M_L :

$$W = \langle M_K; M_O; M_A; M_F; M_L \rangle. \quad (3.8)$$

Категорійна модель визначається на основі місії, цілей і завдань функціонування ССУ з використанням груп класифікаційних параметрів. Перша (загальна) група містить параметри цільової місії, предметну область, масштаб і визначеність завдань. Організаційна модель визначає аспекти контролю за функціонуванням ССУ, зокрема, підпорядкування, персонал, методи обробки ситуаційної інформації та часові обмеження для прийняття рішень. Архітектурна модель визначає конструктивні особливості ССУ та містять спосіб розгортання, універсальність, кількість і тип фізичних розташувань тощо. Параметри четвертої групи визначають інженерно-технологічні аспекти функціонування ССУ з переліком технічного обладнання, номенклатурою засобів моделювання ситуацій, використанням технологій, рівня безпеки, рівня автоматизації оцінки ситуацій тощо.

Параметри організаційної моделі ССМ визначаються головним координаційним механізмом, типом ядра (основної частини) організації, загальними проєктними параметрами, ситуаційними факторами (мотиваціями).

Архітектурна модель визначається обраним архітектурним зразком (моделлю, каркасом) для моделювання архітектури програмно-насичених систем [44]. Зокрема, модель Захмана (Zachman Framework) для архітектури інформаційних систем [229], розроблена Інститутом типової архітектури Захмана (Zachman Institute for Framework Architecture, ZIFA), включає шість аспектів опису архітектурного представлення системи:

- мотиваційні наміри, цілі організації та основні правила, за якими вона функціонує і які визначають її діяльність (A_M);
- персонал, підрозділи та інші елементи організаційної структури, відношення між ними, розподіл повноважень та відповідальності в рамках організації (A_S);

- сутності і дані, які обробляються в організації, перелік інструментів, які використовуються для цього (A_D);
- виконувані організацією та її різними підрозділами функції й операції з даними, а також процеси, що їх реалізують (A_f);
- просторовий розподіл елементів і зв'язків між віддаленими частинами в межах розподіленої мережі організації (A_G);
- часові характеристики і обмеження на функціонування організації, події, важливі для її скоординованої діяльності, що відповідають реалізації часових циклів процесів в організації (A_R).

Таким чином, архітектурна модель має такий вигляд:

$$M_A = \langle A_M, A_S, A_D, A_f, A_G, A_R \rangle .$$

Модель модальностей ССУ заснована на логічних моделях, що враховують відповідні модальності [20], [39]. Модальна логіка та їх розширення використовуються для формалізації тверджень слабо формалізованих систем, включаючи природні мови. Алетична логіка оперує основними модальностями необхідності, можливості і ймовірності. Деонтична логіка оперує модальністю зобов'язання, допустимістю, заборонами щодо норм і нормативних понять. Аксиологічна логіка оперує модальностями цінностей. Абсолютна аксіологічна логіка оперує модальностями: добре, нейтрально (байдуже), погано. Відносна аксіологічна логіка оперує модальностями: краще, рівноцінно, гірше. Епістемічна логіка може описувати стан знань агентів у системах агентів, де знання (незнання) означає, що група агентів має (не має) певні знання і для цієї групи можуть бути правдиві деякі твердження. Темпоральна (часова) логіка, що оперує модальностями істини тверджень минулого, сьогодення, майбутнього. У логічній моделі ССУ також використовує просторову модальність (там, тут, ніде), докастичну модальність (визначає набір довірчих тверджень), гібридну модальність (використовує додаткові номінальні символи, які справедливі тільки в одному з можливих семантичних (сміслових) світів).

Логічна модель модальностей формується на шкалі Кріпке з відповідною семантикою Кріпке [256]. Використання моделей Кріпке для моделювання систем може ґрунтуватися на логіці дерева обчислень (computation tree logic, CTL) [257]. Додаткові засоби для моделювання надає КНМЛ [258].

Реалізація формальної моделі знань ССУ у форматі онтологічних концептів моделі знань представляється як концепти (поняття) моделі категорій, структурні та архітектурні моделі. Відношення між концептами представлені у вигляді відношень між концептами онтології. Функції інтерпретації, задані на концептах і/або відношеннях онтології, представляються виразами (термами) дескрипційної (описової) логіки [259], еквівалентної висловлюванням модальної логічної моделі. Для встановлення еквівалентності з виразами дескрипційної логіки модальна логічна модель формалізується як мультимодальна логіка з n незалежними модальностями. Тоді пропозиційні змінні кожної модальності інтерпретуються як атомарні концепти дескрипційної логіки. Булеві функції кон'юнкції, диз'юнкції і заперечення інтерпретуються як перетин, об'єднання і доповнення концептів, модальні вирази необхідності і можливості інтерпретуються кванторами універсальності і існування. Отже, логічна модель представлена як

$$M_L = \langle L_M, L_A, L_C, L_U, L_R \rangle,$$

де L_M – множина модальностей, L_A – множина аксіом логічної системи, L_C – множина логічних умов і обмежень, L_U – множина правил побудови логічних виразів (термів), L_R – множина відношень між логічними термами.

Модель організаційної підтримки ССУ є основою для реалізації процесів діяльності при ситуаційному управлінні в мінливих умовах з адаптацією до конкретних проблем предметної області підтримки прийняття рішень. Відповідно до специфіки функціонування найбільш адекватною моделлю організаційної конфігурації ССУ є адхократична (ad-hoc) модель, описана в [255].

Модель організаційної підтримки базується на конкретній організаційній структурі. Формально модель організаційної структури описується кортежем

$$M_O = \langle O_E, O_K, O_C, O_D, O_S, O_R \rangle,$$

де O_E – множина структурних елементів організації, O_K – множина основних механізмів координації, O_C – множина типів ядра (основної частини) організації, O_D – множина основних конструктивних параметрів, O_S – множина ситуаційних факторів (мотивацій), O_R – відношення між множинами O_E, O_K, O_C, O_D, O_S .

Конкретні параметри моделі організаційної підтримки визначаються категорією ССУ. Класифікація ССУ може ґрунтуватися на різних критеріях класифікації. Зокрема, такими характерними критеріями класифікації ССУ можуть бути цільове призначення, склад засобів ситуаційного моделювання, масштаб завдань, метод розгортання, визначеність рівня завдань, метод обробки ситуаційної інформації, ступінь універсальності, вид підпорядкування та управління, рівень автоматизації оцінки ситуації, розповсюдження (кількість приміщень), кількість і структура персоналу, необхідний для прийняття рішень час, використовувані технології, рівень захисту, проблемна область застосування. Деякі з цих класифікаційних ознак є базовими (вирішальними), а решта є додатковими або уточнюючими. Категорія ССУ визначається на основі її місії, цілей і завдань діяльності відповідно до моделі функціонування організації [19].

Пропонується така ієрархія класифікаційних критеріїв. Перший (загальний) рівень включає критерії цільового напрямку, проблемної області, масштабу та визначеності завдань. Критерії другого рівня визначають адміністративні аспекти діяльності ССУ, зокрема, підпорядкування, кадрове забезпечення, обробку ситуаційної інформації, обмеження часу прийняття рішень. Критерії третього рівня визначають конкретний вид ССУ і включають в себе спосіб розміщення, універсальність, кількість приміщень. Критерії четвертого рівня визначають техніко-технологічні аспекти діяльності ССУ і включають перелік технічного обладнання, структуру інструментів ситуаційного моделювання, використовувану технологію, рівень безпеки та рівень автоматизації оцінки ситуації. Ієрархічна категорійна модель представлена кортежем виду

$$M_K = \langle K_G, K_U, K_B, K_T, K_R \rangle,$$

де K_G – множина категорій загальних ознак, K_U – множина параметрів аспектів управління, K_B – множина специфічних параметрів проектування, K_T – множина технічних та технологічних параметрів, K_R – відношення між множинами K_G, K_U, K_B, K_T .

Відповідно до аспектів функціонування конвергентної ССУ, її модель може бути представлена кортежем

$$M_F = \langle F_L, F_A, F_N, F_{Cnt}, F_P, F_R \rangle,$$

де F_L – множина формальних комунікаційних потоків, заснованих на повноваженнях і відповідальності доменів, F_A – множина регульованих потоків діяльності, F_N – множина неформальних комунікаційних потоків, F_{Cnt} – множина потоків процесів прийняття рішень, F_P – множина робочих груп, в яких циркулюють потоки різних типів, F_R – відношення між множинами $F_L, F_A, F_N, F_{Cnt}, F_P$.

Діяльність організації може бути представлена як композиція функцій окремих компонентів, які реалізуються через відповідні послуги. Основними видами композиції послуг є оркестровка і хореографія. Формальний опис оркестровки сервісів на основі запропонованої моделі може бути реалізований за допомогою BPEL [260] або YAWL [261]. Хореографію сервісів можна формалізувати за допомогою Chor [262], BPEL4Chor [263], WS-CDL [264].

Використання YAWL [261] дозволяє будувати моделі процесів прийняття рішень, використовуючи уніфікований інтерфейс для доступу до переліку сервісів організації з залученням персоналу та на основі стандартів веб-сервісів, що підтримуються типовими патернами (зразками, шаблонами) робочих процесів [265]. Цей інтерфейс дозволяє розробникам створювати власний робочий перелік сервісів для підтримки завдань персоналу відповідно до їхніх потреб. Крім того, система YAWL поставляється зі списком сервісів за замовчуванням, які підтримують різні типи розподілу та обробки завдань персоналу.

Спільну діяльність персоналу можна розглядати з точки зору агентно-орієнтованого підходу з використанням моделі поведінки агентів [78]. Поведінка агентів персоналу ґрунтується на їх знаннях і має певний сенс (ситуаційна семантика). Отже, організаційна підтримка діяльності агентів персоналу в аспекті ситуаційної семантики може бути формалізована з використанням відповідної моделі знань. Отримана в результаті узагальнена поведінкова модель ситуаційного агента представляється кортежем

$$A_b = \langle T, P, C, M, D, W \rangle,$$

де T – засоби опису ситуації, P – засоби комунікативного управління у змінних середовищах зв'язку, C – засоби механізмів координації, M – засоби обміну повідомленнями між агентами, D – засоби опису дій, W – формальна модель функціонування ССУ (1) і визначає контекст компонентів T, P, C, M, D .

Типові патерни робочих процесів (заплановані процедури) підтримуються сервісами ССУ [50]. Інформаційні технології в ССУ впроваджуються на основі ієрархії процедур інфраструктурного (регламентного), організаційного (адміністративного) та предметного (змістовного, спеціального) рівнів. Зокрема, регламентні процедури (рівень інфраструктури) повинні забезпечувати такі функції [266]:

- автентифікація та авторизація учасників наради з будь-якого робочого місця ССУ;
- реєстрація учасників наради;
- централізоване управління мікрофонами учасників наради;
- контроль відображення інформації колективними та індивідуальними екранами;
- процедури реалізації різних моделей голосування учасників наради;
- контроль синхронного перекладу;
- аудіозв'язок між учасниками наради та їх консультантами;
- контроль презентаційних матеріалів із будь-якого робочого місця;

– інформаційне забезпечення нарад (дані про учасників, порядок денний та деякі інші сервісні операції).

Організаційні процедури (адміністративний рівень) реалізують такі функції [267]:

- заповнення бази даних експертів ССУ;
- складання списку учасників наради (тобто пошук експертів за відповідною спеціалізацією та іншими критеріями);
- розробка порядку денного наради;
- інформування потенційних учасників наради;
- забезпечення наради всією необхідною інформацією (формулювання проблеми, кінцевого списку учасників, пунктів порядку денного, документів тощо);
- збереження результатів наради в базі знань ССУ для використання при підготовці та проведенні наступних нарад, а також для розрахунку рейтингу експертів.

Процедури рівня предметної (проблемної, змістовної) області повинні передбачати засоби моделювання, аналізу, класифікації процесів цієї предметної області з метою прийняття обґрунтованих, відповідних ситуацій управлінських рішень та їх оцінки.

Ефективна реалізація ієрархії технологій для ССУ може бути здійснена на основі сервісно-орієнтованих інформаційних систем. Служби вищих рівнів можуть застосовувати більш низькі рівні послуг і організовувати внутрішню взаємодію відповідно до логіки прийняття рішень у ССУ. Координація цієї взаємодії може ґрунтуватися на механізмах оркестрування та хореографії сервісів [198].

Особливістю функціонування конвергентної ССУ є поєднання регулярних регламентних процедур із процедурами, що визначаються поточними завданнями ситуаційного управління, розв'язуваними за допомогою ССУ. Таким чином, цільова модель реалізації процесів діяльності (бізнес-процесів) може змінюватися. Змінюваність цільових моделей ситуаційного управління визначає складність визначення стандартизованих процедур обробки різних ситуацій, що виникають на практиці.

Слід зазначити, що в реальній організації неможливо досягти повної формалізації процесів через вплив на діяльність організації невизначеності станів навколишнього середовища та інших випадкових факторів. Але ізоляція фрагментів діяльності, які піддаються формалізації, дозволяє спростити й підвищити ефективність роботи всієї організації. Компоненти визначеної моделі ССУ мають слабо структурований і неформальний характер. Адекватним для них є використання технологій ситуаційного управління, орієнтованих на знання.

3.4 Модель інтероперабельності багаторівневої ССУ

Автономність агентів та їх розподіленість у середовищі конвергентної ССУ обумовлює використання для побудови систем еталонної моделі систем відкритої розподіленої обробки (EMCBPO, Reference Model of Open Distributed Processing (RM-ODP) Systems,) визначеної стандартом ISO/IEC 10746 [269]. Зокрема, шаблон RM-ODP стандарту ISO/IEC 10746 пропонує п'ять загальних та взаємодоповнюючих точок зору на систему та її середовище:

- точка зору на організацію, яка фокусується на призначенні, обов'язках та політиці системи. Він описує вимоги бізнесу та шляхи їх задоволення;
- точка зору інформації, яка зосереджена на семантиці інформації та виконаній обробці інформації. Він описує інформацію, якою керує система, та структуру і тип вмісту допоміжних даних;
- обчислювальна точка зору, яка дозволяє розподілити через функціональне розкладання в системі на об'єкти, які взаємодіють з інтерфейсами. Він описує функціональність, що надається системою, та її функціональне розкладання;
- інженерна точка зору, яка фокусується на механізмах та функціях, необхідних для підтримки розподілених взаємодій між об'єктами в системі. Він описує розподіл обробки, що виконується системою для управління інформацією, та забезпечення функціональності;

– технологічна точка зору, яка фокусується на виборі технології системи. У стандартному шаблоні описані технології, обрані для забезпечення обробки, функціональності та подання інформації.

Одним із ключових механізмів ефективного функціонування конвергентної ССУ є забезпечення інтероперабельності між її компонентами. Відповідно до визначення інтероперабельності у стандарті ISO/IEC 24765 [270] та європейської моделі інтероперабельності (European Interoperability Framework, EIF), прийнятого в ЄС, базовим рівнем інтероперабельності є технічна інтероперабельність (рис. 3.5) [271]. У контексті стандарту ISO/IEC 10746 поняття інтероперабельності означає здатність об'єктів до взаємодії (співпраці), тобто здатність взаємно передавати інформацію для обміну подіями, пропозиціями, запитами, результатами, зобов'язаннями та потоками. Ефективна інтероперабельність може бути забезпечена на основі оптимізаційного [272], знання-орієнтованого [273] та загально-системного підходів [274], [275].

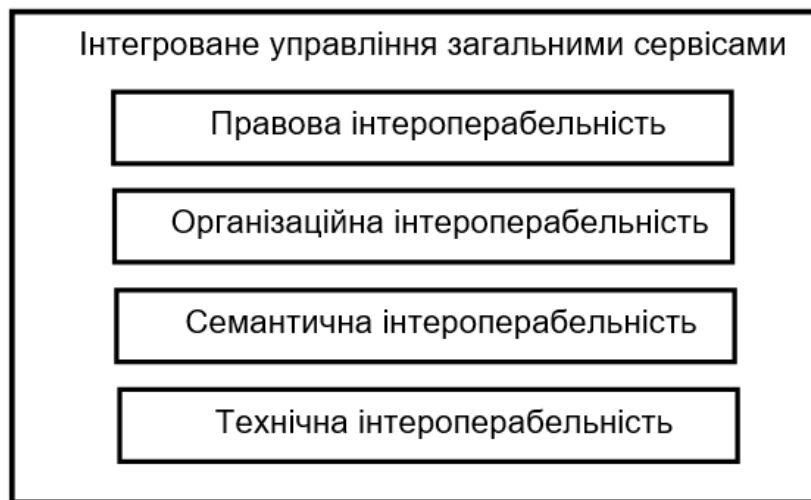


Рисунок 3.5 – Чотири рівні інтероперабельності

На основі EIF була розроблена Європейська еталонна архітектура інтероперабельності (EIRA) [276].

У контексті корпоративного приватного чи гібридного середовища хмарних обчислень технологія ситуаційного управління має бути реалізована у вигляді набору сервісів і надаватися у вигляді інтегрованого адаптованого до розв'язуваної

задачі ситуаційного управління набором сервісів, які організуються на основі ієрархії процедур інфраструктурного (регламентного), організаційного (адміністративного) та предметного (змістовного) рівнів. Для реалізації сервісів потрібні певні ресурси, статистика використання яких залежить від моделі профілю навантаження.

У контексті хмарних обчислень для надання сервісу в технології ситуаційного управління потрібно для кожного етапу цієї технології визначити необхідну множину сервісів, потреби в ресурсах для реалізації сервісів, організувати безконфліктне «еластичне» виділення ресурсів і забезпечити коректну «безшовну» стиковку між етапами технології. Підбір сервісу може здійснюватися на основі параметричної формалізації цільової функції задачі ситуаційного управління.

При побудові корпоративних хмарних технологій, у тому числі технологій для ССУ, кожен етап технології описується шаблоном, що включає функціональні вимоги, вимоги до вхідних і вихідних даних, опису інтерфейсів і методів доступу до хмарних сервісів. При реалізації технології ситуаційного управління в середовищі хмарних обчислень на кожному етапі використовуються специфічні для даного етапу прикладні хмарні сервіси.

На етапі аналізу проблеми, що вимагає ухвалення рішення, використовуються сервіси обчислень (статистична обробка даних), баз даних і знань, обробки експертних оцінок і методи визначення рейтингів.

На етапі підготовки інформації, необхідної для ситуаційного управління, використовуються сервіси баз даних і знань, електронного документообігу, забезпечення безпеки даних, аналізу та верифікації даних. На цьому етапі може виникнути необхідність використання глобальних хмарних сервісів, наприклад, зовнішніх джерел даних, що містять інформацію з даної проблеми.

На етапі підбору та підготовки колективу для здійснення ситуаційного управління знадобляться сервіси HRM (Human Resource Management), баз даних і знань, телекомунікацій, комп'ютерного навчання, обміну повідомленнями, створення і управління віртуальними організаціями.

На етапі підготовки та налаштування інформаційного середовища ситуаційного управління потрібно використовувати сервіси CRM, управління віртуальними організаціями, баз даних і знань, обчислювальні (для попереднього моделювання ситуацій із проблеми прийняття рішення).

На етапі реалізації процедур діяльності відповідно до цілей ситуаційного управління використовуються сервіси управління колективною роботою користувачів, обчислювальні (моделювання та аналіз даних), верифікації даних, баз даних і знань, електронного документообігу.

На етапі документального та нормативного забезпечення прийнятих управлінських рішень використовуються сервіси електронного документообігу, баз даних і знань.

На етапі організації реалізації прийнятих управлінських рішень використовуються сервіси електронного документообігу, телекомунікацій, баз даних і знань, динамічної оптимізації.

На етапі аналізу результативності та якості ситуаційного управління використовуються методи аналізу та оптимізації даних, обчислювальні (статистичний аналіз і моделювання), баз даних і знань, електронного документообігу.

Інтеграція сервісів у рамках єдиного хмарного середовища (вертикальна інтеграція) забезпечує коректну «безшовну» інтеграцію етапів технології прийняття рішення (горизонтальна інтеграція).

Створення корпоративної технології як системи сервісів полягає у побудові та конфігуруванні середовища хмарних обчислень відповідно до параметричних описів технології. Параметричні описи доцільно представляти у вигляді шаблонів та/або метаописів, які зберігаються у відповідних сховищах даних і знань. Хмарне середовище надає розширені можливості для підтримки діяльності ССУ протягом всього її життєвого циклу:

- на етапі розробки та впровадження забезпечується інтеграція середовища та інформаційних ресурсів підприємства і оптимізація управління ними, створення

платформи для обробки корпоративної інформації, реалізація та каталогізація сервісів обробки даних;

- на етапі навчання користувачам надається перелік доступних сервісів і технологій, методів доступу і способів їх налаштування для вирішення завдань;
- на етапі експлуатації використовуються шаблони, які описують технології обробки корпоративної інформації та залучені для цього сервіси із хмарного середовища;
- на етапі модернізації забезпечується узгоджена зміна хмарних сервісів відповідно до вимог користувачів з урахуванням можливостей та обмежень корпоративного середовища;
- на етапі обслуговування здійснюються централізована верифікація і тестування корпоративного хмарного середовища, платформи і сервісів із точки зору узгодженості корпоративних політик, бізнес-процесів і використовуваних технологій;
- виведення з експлуатації на рівні додатків і технологій може здійснюватися простим видаленням сервісів, шаблонів і підтримуючих їх віртуальних ресурсів, а на рівні середовища і платформи – заміною застарілих сервісів і компонентів новими з передачею їм підтримки функцій відповідних хмарних сервісів.

Таким чином, вертикальна і горизонтальна інтеграція корпоративних хмарних сервісів забезпечує прозорість, гнучкість, надійність та ефективність функціонування інформаційної системи підприємства.

3.5 Формальна модель конвергенції технологій ССУ

Побудова і використання ССУ у різних сферах застосування вимагає розробки методів і засобів конвергенції інформаційних технологій для здійснення адекватного ситуаційного управління у цільовій проблемній області. Конвергенція визначається як глибока інтеграція знань, засобів та усіх раціональних видів людської діяльності

для досягнення спільної мети, можливостей отримання відповідей на нові питання щодо зміни відповідної фізичної або соціальної екосистеми.

Розробка конвергентних систем, у першу чергу, вимагає оцінки їх доцільності. Коли ж рішення про необхідність розробки такої системи прийнято, слід формалізувати опис предметної області і провести декомпозицію задачі створення ССУ на підзадачі.

Для здійснення процесів СУ використовуються різні набори (cases) інформаційних, комунікаційних, організаційних технологій (ІКОТ), необхідні для вирішення проблем СУ у цільовій предметній сфері. Різноманіття ІКОТ, змінюваність проблем СУ та вимог щодо їх вирішення потребує задачі швидкої адаптації ССУ для розв'язання актуальних задач. Використання стандартних технологій проєктування ІКС може бути неприйнятним через обмеженість ресурсів та необхідність оперативної інтеграції (конвергенції) запитаних ІКОТ у рамках конкретної реалізації (конфігурації) ССУ для вирішення актуальних проблем СУ.

Функціональні спроможності та властивості окремої конфігурації ССУ визначаються її архітектурою. Системна архітектура визначається як концептуальна модель, яка визначає структуру, поведінку та множинність видів (проекцій) системи. Для представлення кіберматичних систем [171], до яких можна віднести ССУ, використовуються різні види системних архітектур: програмна, організаційна (підприємства), технологічна, інформаційна, процесна тощо. Опис архітектури є формальним описом і поданням системи [13], організованої таким чином, щоб підтримувати висновки (умовивід) про структури та поведінку системи. Загальний опис архітектури системи складають три типи описів: структурні, поведінкові та організаційні.

Програмна архітектура визначається як процес визначення структурованого рішення, яке відповідає всім технічним і експлуатаційним вимогам, одночасно оптимізуючи загальні атрибути якості такі, як продуктивність, безпека і керованість. Вона включає в себе ряд рішень, заснованих на широкому діапазоні факторів, і кожне з цих рішень може мати значний вплив на якість, продуктивність, ремонтпридатність і загальний успіх програми [277], [278].

Технологічна архітектура визначає набір принципів і стандартів (індустріальних стандартів; стандартів, пов'язаних із продуктами; конфігурацій), які забезпечують керівництво щодо вибору і використання таких технологій як апаратні платформи, операційні системи, системи управління базами даних, засоби розробки, мови програмування, програмне забезпечення (ПЗ) проміжного шару, сервіси електронної пошти, каталоги, системи безпеки, мережева інфраструктура тощо. Технологічну архітектуру визначають мережева архітектура та архітектура платформ. Мережеву архітектуру утворюють обчислювальні мережі, використовувані комунікаційні протоколи, сервіси та системи адресації в мережах, а також методики забезпечення безперебійної роботи мереж у форс-мажорних умовах. Архітектура платформ включає апаратні засоби обчислювальної техніки: сервери, робочі станції, пристрої зберігання даних та інше комп'ютерне обладнання, операційні й керуючі системи, утиліти та офісні програмні системи, методики забезпечення безперебійної роботи апаратури (головним чином, серверів) і баз даних у форс-мажорних обставинах. Основне призначення технологічної архітектури – це забезпечення надійних ІТ-сервісів, що надаються в рамках всього підприємства в цілому і координованих централізовано, як правило, підрозділами інформаційних технологій.

Однією з вимог, що висуваються до ССУ, є адаптивність до конкретних задач ситуаційного управління. Така адаптивність досягається шляхом конвергенції потрібних технологій у рамках ССУ для реалізації процесів ситуаційного управління відповідно до визначених умов. Основою для конвергенції технологій є визначення вимог стосовно очікуваних результатів кожного з етапів ситуаційного управління і створення агрегованого набору технологій із загальними спроможностями, що забезпечують виконання цих вимог. Формалізація системного опису ССУ дозволяє побудувати формальну модель конвергенції технологій у складі цільової ССУ. Така формалізація може бути проведена на основі архітектурного підходу до опису ССУ. На основі формалізації можуть бути пов'язані між собою вимоги до конвергентної ССУ, потреби для їх задоволення та спроможності інформаційних технологій, об'єднаних у конвергентній архітектурі.

Створення ССУ пов'язане з поєднанням її компонентів у цілісну систему, призначену для вирішення певного класу проблем ситуаційного управління. Цілісність системи досягається шляхом конвергенції її складових. Цільове призначення ССУ здійснюється через цілісну інформаційну технологію ситуаційного управління, побудовану шляхом конвергенції необхідних інформаційних технологій.

Особливістю інформаційної технології є те що її входом і виходом являється інформація, яка відрізняється лише за категорією, призначенням, структурою та змістом. На основі категорії та призначення вхідної й вихідної інформації інформаційних технологій можна побудувати їх класифікацію. Зокрема, за призначенням можна виділити інформаційні технології:

- реєстрації та обробки первинних сигналів;
- структурування інформації;
- очищення інформації;
- добування, перетворення та завантаження інформації;
- аналізу інформації;
- медіації (візуалізація, озвучування, роз'яснення) інформації;
- семантизації інформації;
- формалізації семантики;
- оцінки корисності інформації;
- побудови конструктивних інформаційних моделей.

Традиційні технологічні підходи до створення ІКС потребують додаткових витрат ресурсів (людських, часових, фінансових) для конвергенції технологій у рамках цільової ССУ. Для вирішення цієї проблеми використовуються різні методології модельно-орієнтованої розробки (model-driven development, MDD) систем [279], зокрема, такі, як предметно-орієнтована розробка (domain-driven development, DDD) [280], [281], розробка, керована вимогами [282], розробка, керована функціональністю (feature driven development, FDD) [283], та ін. Крім того, одним із напрямів підвищення ефективності розробки ІКС є можливість багаторазового використання готових програмно-технологічних компонентів у

нових розробках. Цей напрям одержав назву "складальне програмування" [284]. Однак такий підхід висуває жорсткі вимоги до функціональної універсальності та стандартизації архітектури інформаційних технологій, а також забезпечення їх портабельності в гетерогенному інформаційному середовищі. Крім того, участь у створенні спеціалізованих ІКС, зокрема, ССУ, приймають участь не лише ІТ-спеціалісти, а й фахівці із предметних сфер застосування ІКС. У зв'язку з цим, доцільно створення засобів автоматизації із проблемно-орієнтованої розробки ІКС на основі вимог і знань фахівців із предметної області [22].

Формальний опис предметної області СУ, для якої створюється проблемно-орієнтована ССУ, є ієрархією концептів (понять) і функціональних перетворень, якими будуть оперувати користувачі. Формальний опис предметної області має також містити узагальнений опис моделі процесів СУ. Композиція технологій СУ повинна здійснюватися з урахуванням архітектури ССУ.

Таким чином, основну задачу композиції ІКТ у рамках конвергентної архітектури ССУ можна визначити як встановлення відповідності між формальним описом прикладної області та засобами інформаційних технологій [11]:

$$F: \langle A, D \rangle \rightarrow \langle B, C, R \rangle, \quad (3.9)$$

де F – функція відображення моделі A предметної області й моделі вимог D до проблемно-орієнтованої ССУ на множину технологічних засобів (алфавіт) СУ B і множину функцій управління C цими технологічними засобами на множині R допустимих ситуацій у предметній галузі.

Множину B технологічних засобів СУ, в залежності від їх семантичного змісту та функціональних спроможностей, можна розділити на підмножини, які визначають класи технологічних засобів СУ:

$$B = \bigcup_{i=1}^n B^{(i)},$$

де n – визначає кількість технологічних засобів. В основу розбивки множини B на підмножини може бути покладена ознака функціонального призначення, тобто технологічна орієнтація функції, що покладається на певний клас $B^{(i)}$ технологічних засобів.

За функціональним призначенням можна виділити такі технологічні засоби СУ:

- засоби візуалізації;
- засоби отримання та збереження інформації;
- засоби фільтрації та очищення даних;
- засоби аналітичної обробки;
- засоби персоналізації;
- засоби інформаційної безпеки;
- системні засоби;
- конфігураційні засоби тощо.

У свою чергу всі технологічні засоби можна згрупувати за категоріями застосування:

- організаційна;
- системна;
- предметна;
- допоміжна.

Множина C функцій управління технологічними засобами включає в себе допустимі операції маніпулювання цими засобами на множині R допустимих ситуацій.

Таким чином, кінцеву конфігурацію технологій у рамках окремої реалізації ССУ можна представити як процес P конвергенції технологій у проблемно-орієнтованій ССУ:

$$P = \langle \tilde{B}, \tilde{C}, \tilde{R}, \alpha \rangle, \quad (3.10)$$

де $\tilde{B} \subset B, \tilde{C} \subset C, \tilde{R} \subset R, \alpha$ – відношення лінійного порядку на множині $\tilde{C}$.

Вихідними даними для процесу P (3.10) конвергенції технологій у проблемно-орієнтованій ССУ є узгоджена модель знань

$$\Pi = \langle \tilde{A}, \tilde{D}, \beta \rangle, \quad (3.11)$$

де $\tilde{A} \subset A, \tilde{D} \subset D, \beta$ – функція інтерпретації вимог $\tilde{D}$ до ССУ у контексті моделі знань предметної області $\tilde{A}$.

Процес P конвергенції може бути розділений на декілька етапів, що відповідають ступеню наближення конфігурації технологій у ССУ для задоволення вимог $\tilde{D}$. Важливим при реалізації процесу P конвергенції є забезпечення поступовості, тобто збереження результатів проміжних конфігурацій конвергентної архітектури незалежно від ступеня задоволення вимог. Отже, кожен етап процесу P повинен передбачати збереження його результатів.

Першим етапом функціонального відображення F (3.9) є логіко-семантичний (онтологічний) аналіз вимог до архітектури технологій ССУ. У процесі відображення отримуємо онтологічне подання моделі вимог з конкретизованими атрибутами, що відповідають моделі предметної області. Для ССУ таке відображення буде мати вид

$$F_1: \langle A, D \rangle \rightarrow \langle A_s, D_o, D_s, D_c, D_r, S \rangle, \quad (3.12)$$

де $D = D_o \oplus D_s \oplus D_c \oplus D_r$, F_1 – функція онтологічного відображення вимог на предметну область, D_o – множина вимог до організаційних технологій, D_s – множина системних вимог, D_c – множина вимог до технологій предметної області, D_r – множина вимог до допоміжних технологій; S – множина відношень між концептами предметної області A , A_s – онтологічне представлення моделі знань предметної області.

Наступним етапом відображення (3.9) є функціональна декомпозиція F_2 , у процесі якої здійснюється розбиття логіко-семантичного представлення, отриманого в результаті відображення F_1 (3.12) на функціонально однотипні групи, відповідні класам сервісів проблемної області, системної підтримки, функцій організаційної підтримки, допоміжних функцій (наприклад, функцій розширення та інтеграції ССУ):

$$G^{(i)} = \langle A_S^{(i)}, D_O^{(i)}, D_S^{(i)}, D_C^{(i)}, D_r^{(i)}, S^{(i)} \rangle.$$

Тут $\bigcap_{i=1}^n G^{(i)} = 0$, де n – перевірена (верифікована) та підтверджена (валідована)

кількість категорій конвергентних технологічних сервісів.

Таким чином, відображення F_2 можна визначити як

$$F_2 : \langle A_S, D_O, D_S, D_C, D_r, S \rangle \rightarrow \left\langle \bigcup_{i=1}^n G^{(i)}, \Gamma \right\rangle, \quad (3.13)$$

де Γ – функція фільтрації (верифікації та валідації) логіко-семантичної моделі за категоріями сервісів. Наступним етапом відображення (3.9) є предметна інтерпретація результатів відображення (3.13):

$$F_3 : \left\langle \bigcup_{i=1}^m G^{(i)}, \Gamma \right\rangle \rightarrow \langle B, C, R \rangle, \quad (3.14)$$

де $m \leq n$ – число функціональних груп, що визначають кінцеву конфігурацію технологій ССУ. Якщо кортеж $\langle B, C, R \rangle$ відповідає вимогам до сервісів технологій конвергентної архітектури ССУ, то кінцева конфігурація технологій ССУ може бути представлена у вигляді (3.10). Якщо ж кортеж $\langle B, C, R \rangle$ являє собою узагальнену конвергентну модель технологій ССУ (наприклад, опис у вигляді моделей

формальної нотації), то необхідно провести перетворення узагальненої моделі в екземпляр (варіант) її конкретної фізичної реалізації:

$$F_4 : P \rightarrow P'$$

Таким чином, відображення (3.9) можна представити у вигляді суперпозиції відображень

$$F = F_1 \circ F_2 \circ F_3 \circ F_4,$$

де $\circ$ – знак суперпозиції, причому, якщо $P = P'$, то F_3 – тотожне відображення.

3.6 Ситуаційне управління інформаційною безпекою на основі конвергенції моделей знань

Управління безпекою є важливою частиною ситуаційної діяльності. Підтримка прийняття рішень у сфері безпеки – це ситуаційна діяльність, пов'язана з багатоаспектним аналізом загроз, ризиків та вразливостей для функціонування організації. Таким чином, управління безпекою потребує підтримки депозитарію знань, зокрема, у формі бази знань.

Одним із способів використання бази знань в управлінні безпекою є онтологічний підхід до безпеки, який встановлює концептуальні відношення між суб'єктами, що представляють інформацію і дають систематизоване представлення з метою ідентифікації, аналізу та проведення заходів щодо організації безпеки. Багато міжнародних та інших стандартів безпеки й захисту визначають правила оцінки ризиків та підготовки профілів захисту і цілей безпеки.

Не існує уніфікованих рекомендацій щодо створення систем безпеки та захисту підприємств. Кожен випадок унікальний і конкретний. Отже, процес

створення систем безпеки та захисту для підприємства буде більш ефективним із використанням бази знань. Питання структури бази знань є головним і стосується врахування різноманітності різних факторів. У роботах з онтології безпеки [285] та [286] використовується підхід до управління безпекою, орієнтований на активи. Системна точка зору на інформаційну безпеку вимагає комплексного врахування різних активів організації та їх атрибутів безпеки [287], [288], [289]. Такий системний підхід може бути реалізований на основі конвергенції різних моделей [15], [26].

Кожен тип організації потребує адекватного способу захисту. Система ситуативного управління є специфічним типом організації і включає систему (системи) підтримки прийняття рішень як її складову. Існує багато міжнародних стандартів, що стосуються розробки, створення та впровадження систем управління безпекою. Але всі вони викладають лише загальні рекомендації щодо різних аспектів організації безпеки й захисту системи. Створення робочої системи управління безпекою з використанням допоміжних інструментів і, зокрема, бази знань. Розробка адекватної моделі бази знань для управління безпекою системи є головним питанням даної роботи.

Загальним підходом до управління безпекою, зазначеним у стандартах, є підхід, орієнтований на ризик. Отже, основною метою управління безпекою є мінімізація ризиків. Існує багато визначень ризику. Зокрема, стандарт ISO / IEC 27005 визначає ризик як «вплив невизначеності на цілі», а примітка 6 до цього визначення говорить: «ризик інформаційної безпеки пов'язаний з потенціалом, що загрози будуть використовувати вразливості інформаційного активу або групи інформаційних активів і, отже, заподіяти шкоду організації» [290]. Більше того, у стандартному контексті ISO/IEC 27005 «уразливості можуть бути пов'язані з властивостями активу, які можуть бути використані певним чином або з ціллю, відмінною від тієї, що передбачалася під час придбання чи виготовлення активу» [290], або простіше вразливість – це слабкість активу або групи активів, які можуть бути використані однією або кількома загрозами, але загроза, яка не має відповідної вразливості, не може спричинити ризик. І врешті-решт, «оцінка ризику визначає

вартість інформаційних активів, застосовні загрози та уразливості, які існують (або могли б існувати), існуючі засоби контролю та їх вплив на виявлений ризик, потенційні наслідки та, нарешті, пріоритети отриманих ризиків і класифікує їх за критеріями оцінки ризику, встановленими у контексті встановлення» [290]. Отже, залежності між ризиками, загрозами, вразливостями та інформаційними активами можуть бути представлені діаграмою, зображеною на рис. 3.6.



Рисунок 3.6 – Конвергенція компонентів інформаційних ризиків

Оцінка ризику є основою управління безпекою та використання адекватного набору моделей безпеки [291]. Інформаційні активи є частинами ІТ-архітектури. Отже, оцінюючи інформаційні активи, ми розглядаємо ці активи як екземпляр (варіант) ІТ-архітектури. Модель описує фактори ризику, пов'язані з загрозами, вразливістю та ІТ-архітектурою [15].

Конвергентна модель знань управління інформаційною безпекою реалізується на основі моделей знань компонентів ризиків (рис. 3.6). Онтологічна модель знань предметної сфери дозволяє описати основні концепції (положення) предметної області й визначає відносини між ними. Процес побудови онтологій включає:

- класи та їх властивості (classes, properties);
- властивості кожної концепції, що описує різні функціональні можливості і атрибути концепції (слоти (slots), іноді звані ролі);

– обмеження по слотах (також відомих як аспекти / грані (slot facets), іноді звані обмеження ролей). Онтологія разом із безліччю індивідуальних екземплярів класів складають базу знань.

Онтологія є формальним явним описом концептів (понять, класів) цільової предметної області, властивостей кожного концепту, атрибутів концепту (слотів, ролей) і обмежень, накладених на слоти (фасетів, обмеженнями ролей).

Фрагмент часткової онтології архітектурних складових безпеки системи представлений на рис. 3.7.

Приклад запиту до БЗ архітектури безпеки наведено на рис. 3.8.

Фрагмент часткової онтології загроз безпеки системи представлений на рис. 3.9.

Приклад запиту до БЗ загроз безпеки наведено на рис. 3.10.

Фрагмент часткової онтології обробки ризиків безпеки системи представлений на рис. 3.11.

Приклад запиту до БЗ обробки ризиків безпеки наведено на рис. 3.12.

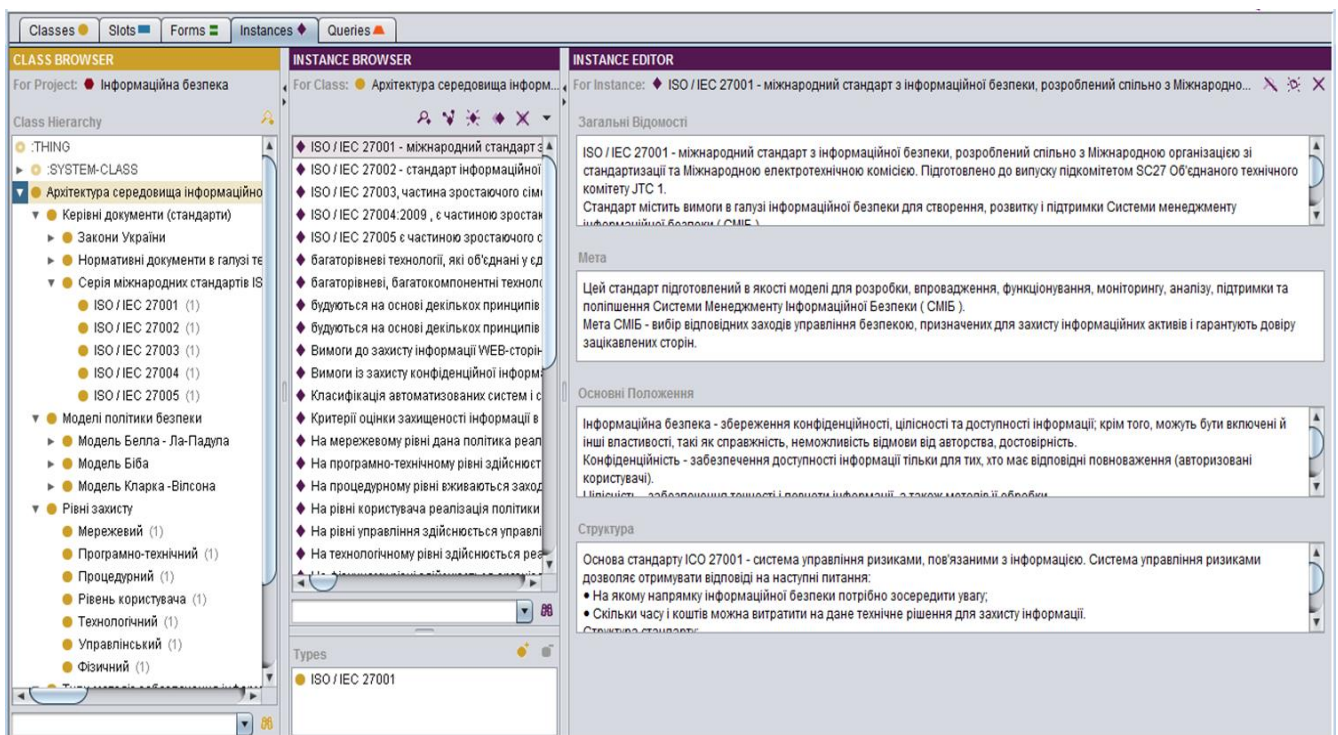


Рисунок 3.7 – Онтологія архітектури безпеки

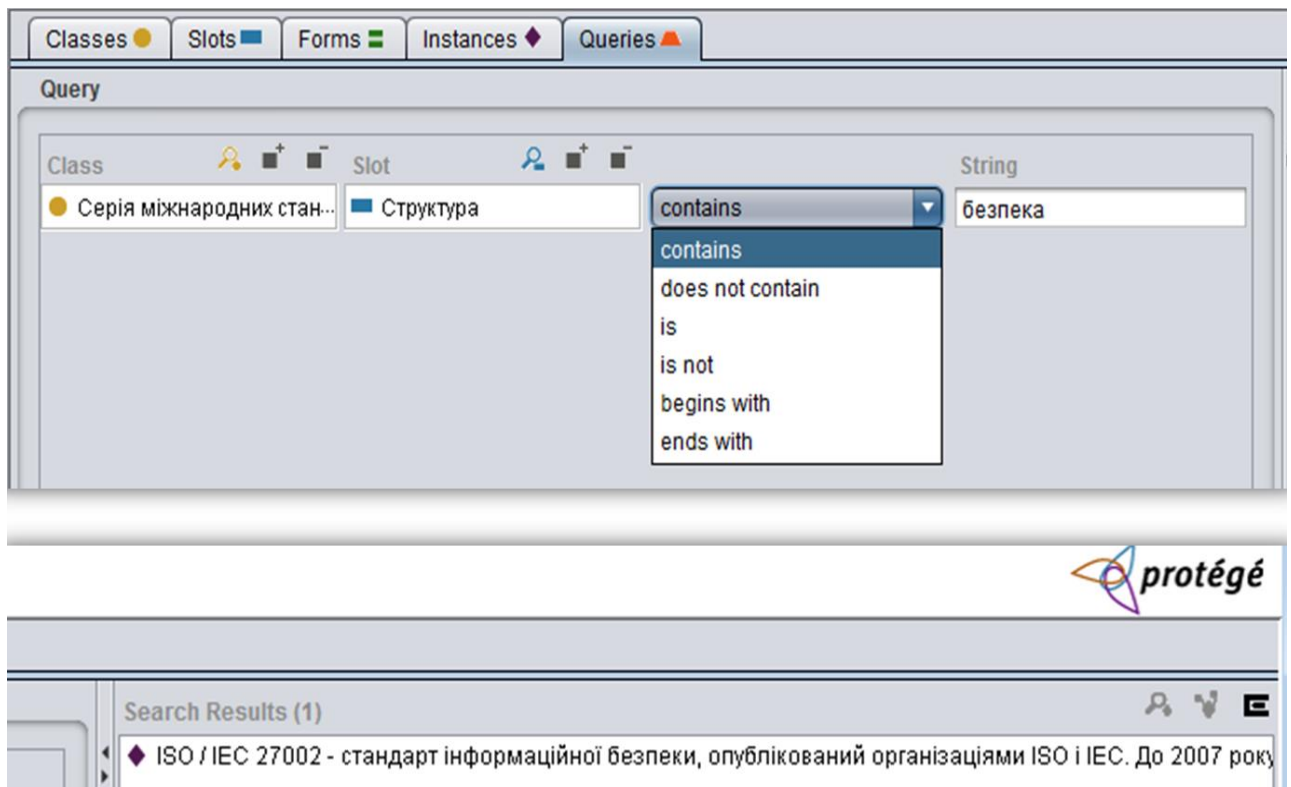


Рисунок 3.8 – Приклад запиту до БЗ архітектури безпеки

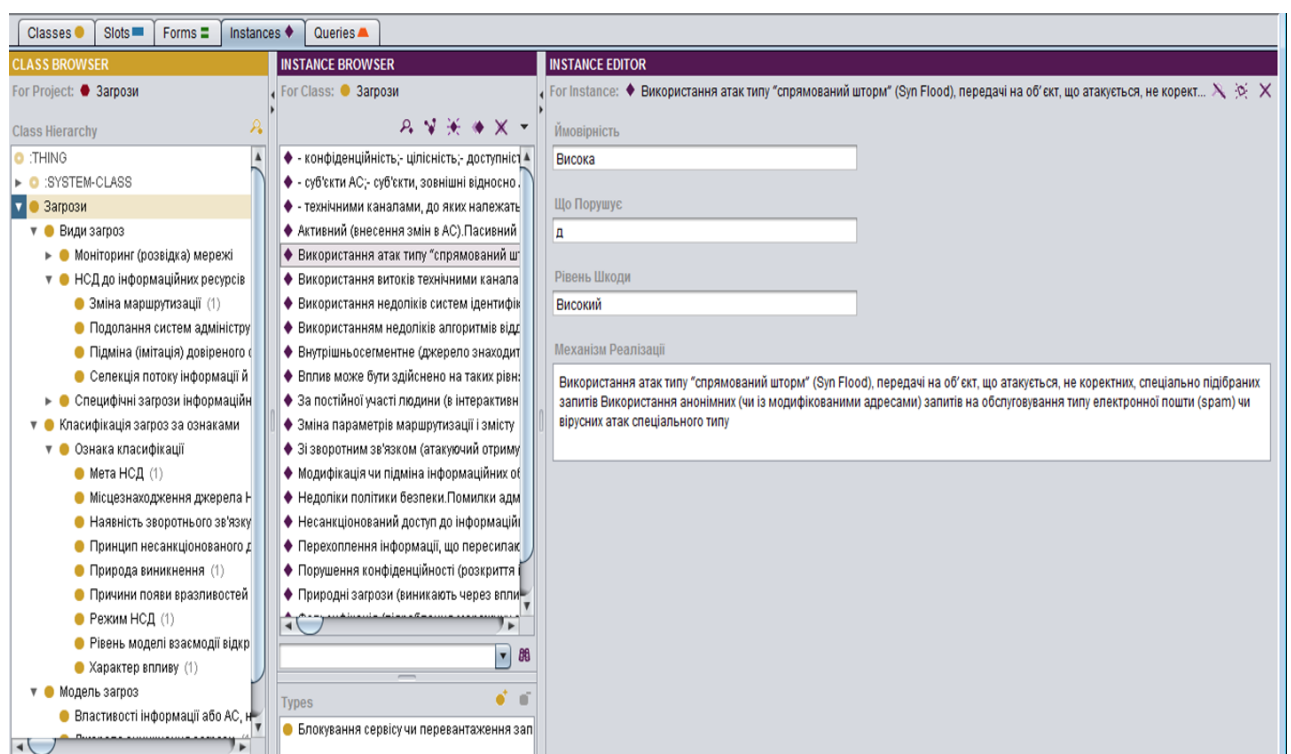


Рисунок 3.9 – Онтологія загроз безпеки

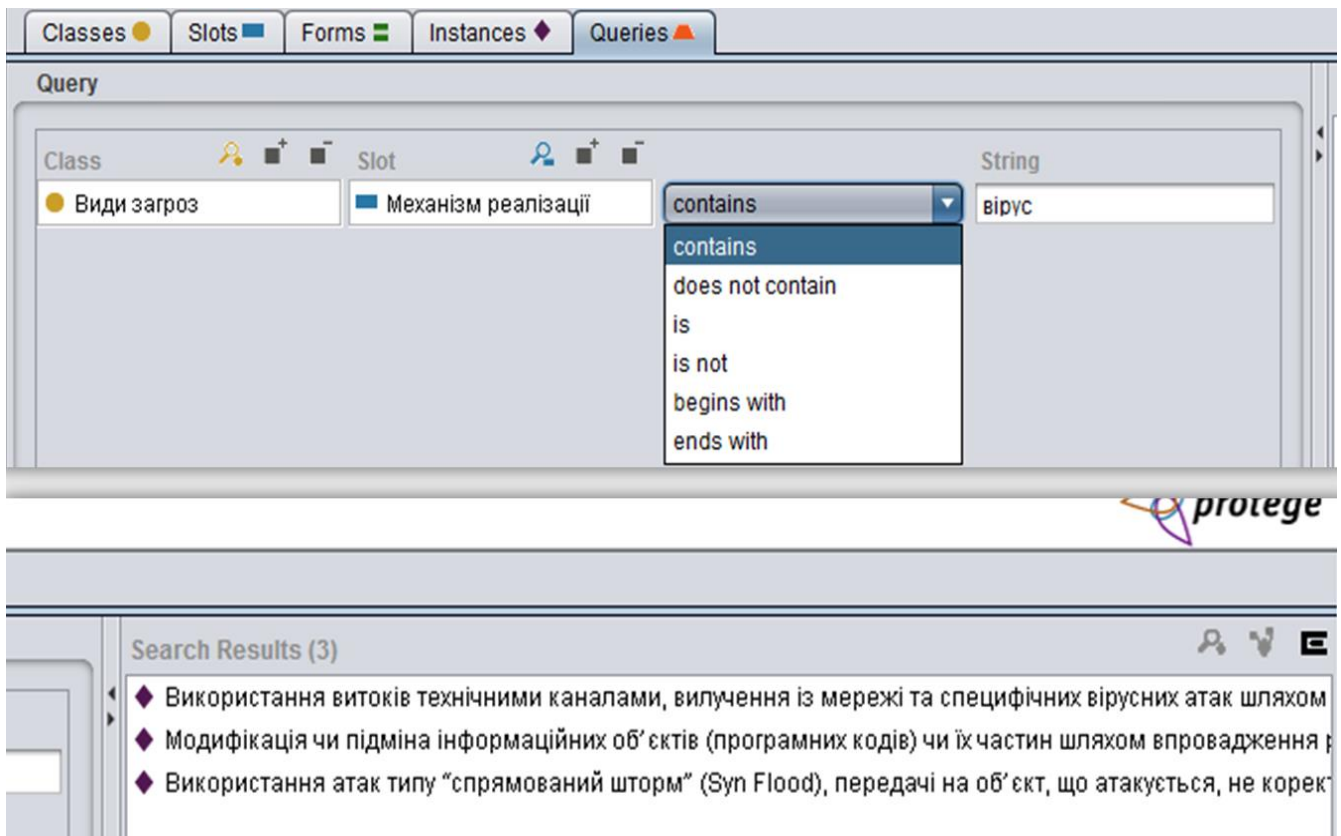


Рисунок 3.10 – Приклад запиту до БЗ загроз безпеки

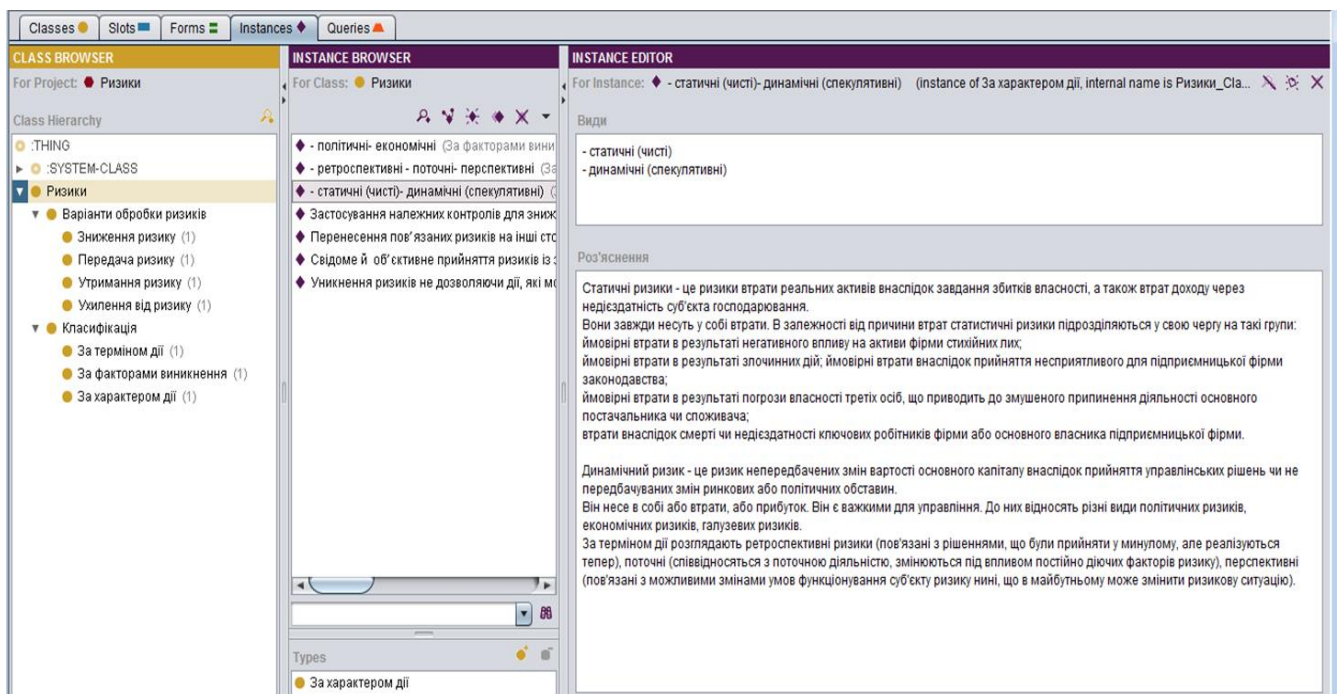


Рисунок 3.11 – Онтологія управління ризиками безпеки

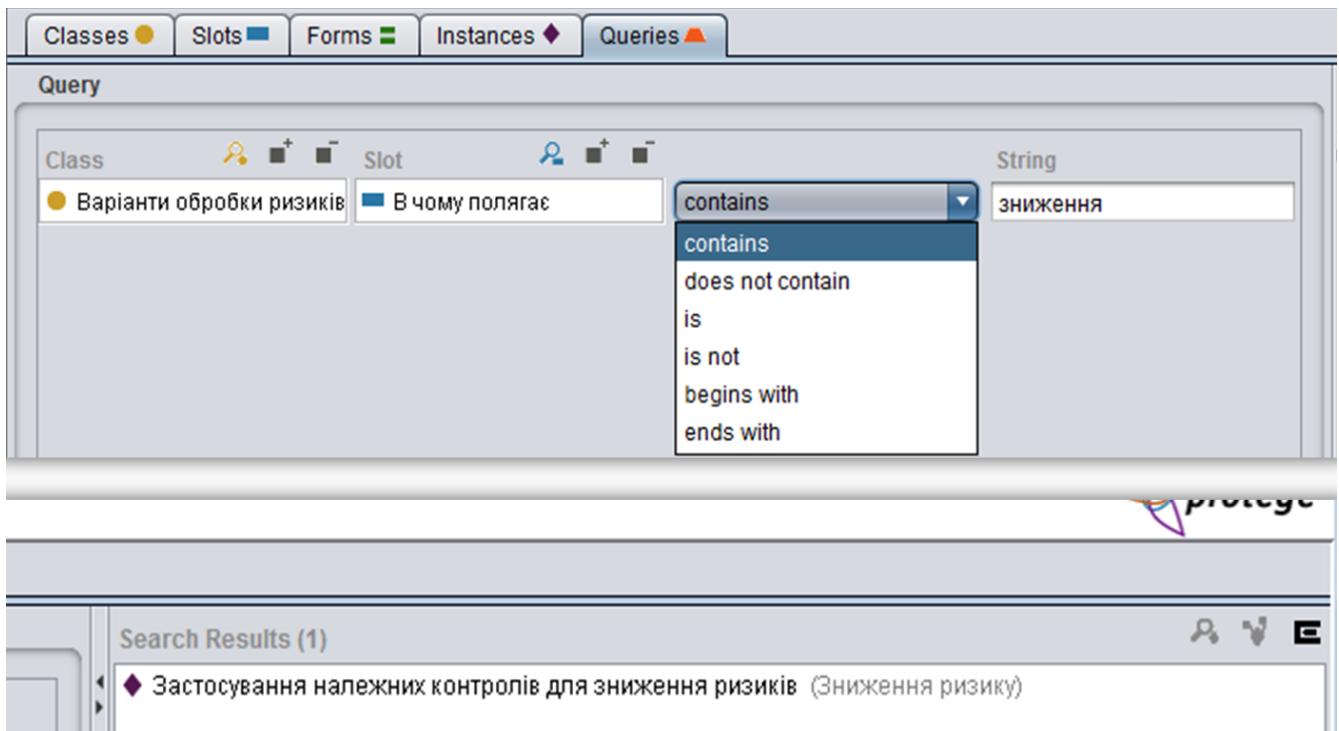


Рисунок 3.12 – Приклад запиту до БЗ загроз безпеки

3.8 Висновки до розділу 3

Встановлення відповідності між формальним описом предметної області і засобами інформаційних технологій дозволяє застосувати орієнтований на знання підхід до розробки ССУ. Запропонований підхід забезпечує формування репозиторію та конвергенцію засобів інформаційних технологій для вирішення цільових проблем ситуаційного управління шляхом композиції етапів онтологічного аналізу вимог, функціональної декомпозиції, предметної інтерпретації та фізичної реалізації. Використання запропонованої моделі процесу проєктування дозволить забезпечити формалізований синтез ССУ для цільової предметної області шляхом конвергенції потрібних ІКТ на основі моделей знань.

Складні системи, зокрема ССУ, мають багаторівневу структуру, з розподілом функцій і сервісів системи за рівнями у відповідності з їх призначенням. Для формалізації архітектурного опису багаторівневої системи запропоновано

чотирикомпонентну модель, яка включає множину функціональних рівнів системи, множину зовнішніх інтерфейсів рівнів системи, множину функцій загальносистемного управління та функцію цільового призначення системи. Також визначена чотирикомпонентна модель кожного рівня, що включає множину функціональних сервісів рівня, множину внутрішніх інтерфейсів рівня, множину функцій управління внутрішніми компонентами та множину функцій адміністрування рівнем. Для представлення семантики рівнів та системи в цілому запропонована багаторівнева модель онтології. Онтології є основою для опису семантики рівня та представлення знань агентів на кожному рівні системи. Цільові групи агентів утворюють «ансамблі» агентів у складі мультиагентної ССУ. Для уніфікації опису мультиагентної ССУ запропонована модель ситуаційного агента, що характеризується раціональною поведінкою, яка формується на основі внутрішніх динамічних знань агента та процесів трансформації інформації в ньому. Раціональна поведінка таких агентів формується на основі лінійних темпоральних логік та мов координації агентів.

Знання як динамічна складова конвергентних ССУ формуються на основі знань попереднього досвіду та використанні логічних методів пізнання перцептивного циклу. Враховуючи природну конвергентність сукупних знань ССУ та знань окремих ситуаційних агентів, що також є поєднанням явних і неявних знань, доцільним є використання моделі трансформації корпоративних знань І.Нонаки.

Запропонована модель організації діяльності ССУ поєднує моделі категорій, моделі організаційної взаємодії (оркестрування та хореографії), архітектурні, функціональні та логічні моделі. Зокрема, логічні моделі забезпечують формування і підтримку моделей предметних знань агентів на основі визначення відношень між концептами (поняттями) і категоріями предметної сфери, що визначають властивості організаційних структур і функцій, які встановлюють правила інтерпретації конвергентних моделей. На основі моделі організації діяльності ССУ запропонована узагальнена поведінкова модель агента.

Формальна модель інтероперабельності (взаємодії) ситуаційних агентів у конвергентній ССУ базується на еталонній моделі систем відкритої розподіленої обробки (RM-ODP), яка передбачає багатоаспектний розгляд таких систем із організаційної, інформаційної, обчислювальної, інженерно-координаційної, технологічної точок зору. Еталонна архітектура інтероперабельності включає чотири рівні інтероперабельності: техніну, семантичну, організаційну, правову.

Запропоновані моделі інтеграції формальних моделей конвергентної ССУ на основі ситуаційної семантики з моделями, що описують аспекти організаційного забезпечення цього процесу. З урахуванням ключових компонентів реалізації процесу діяльності в ССУ були запропоновані методи їх підтримки на основі моделей знань.

Встановлення відповідності між формальним описом предметної області і засобами інформаційних технологій дозволило застосувати знання-орієнтований підхід до розробки конвергентної ССУ. Запропонований підхід забезпечує формування репозиторію та конвергенцію компонентів ССУ для вирішення цільових проблем ситуаційного управління шляхом композиції етапів онтологічного аналізу вимог, функціональної декомпозиції, предметної інтерпретації та фізичної реалізації. Використання запропонованої моделі процесу проєктування дозволяє забезпечити формалізований синтез ССУ для цільової предметної області шляхом конвергенції необхідних компонентів на основі агентного підходу.

Основні результати розділу опубліковані автором у працях [6, 8, 11, 13-17, 19, 20, 22, 26, 39, 43, 47–54, 57, 74, 76, 78-80].

РОЗДІЛ 4

МЕТОДИ СТРУКТУРНОЇ ОПТИМІЗАЦІЇ КОНВЕРГЕНТНИХ АРХІТЕКТУР ССУ

4.1 Загальна постановка задачі оптимізації конвергентної ССУ

Масштаб і функціональність ССУ визначаються множиною вимог, які можуть визначати кількість користувачів, класи цільових проблем, робоче навантаження, потрібну продуктивність, об'єм та вміст баз даних і знань, системні вимоги тощо. У реальності необхідно враховувати також можливу територіальну віддаленість користувачів, наявну інфраструктуру та її технічні характеристики, вимоги, які стосуються результативності та ефективності СУ та інші специфічні для кожної окремої системи обмеження.

Задача створення архітектури конвергентної системи пов'язана з задоволенням вимог до її функціональності з забезпеченням ефективності реалізації. Вимоги до сучасних ІКС, у тому числі до ССУ, висуваються до сервісів відповідних типів спроможностей:

- 1) комп'ютерної платформи;
- 2) базового програмного забезпечення;
- 3) спеціального програмного забезпечення;
- 4) комунікацій;
- 5) об'ємів пам'яті;
- 6) контенту;
- 7) методичного забезпечення;
- 8) організаційної підтримки.

Вимоги виступають як обмежуючі фактори при реалізації ССУ потрібної функціональності й складають множину функцій-обмежень $R(r_i)$, де r_i – квантифіковані (кардинальні) оцінки параметрів вимог, $i = 1, \dots, n$.

Наявні альтернативні можливості для задоволення вимог визначаються варіантами комбінацій сумісних компонентів при реалізації ССУ. Допустимість

використання компонентів для задоволення відповідних вимог може бути визначена матрицею кваліфікацій $V = [v_{ij}]$ розмірності $m \times n$, де v_{ij} – ваговий коефіцієнт оцінки ефективності використання компонента j для задоволення вимоги i . У найпростішому випадку (бінарному оцінюванні), якщо вимога i може бути задоволена компонентом j , то $v_{ij} = 1$, в іншому випадку $v_{ij} = 0$. Специфіка даної задачі полягає у тому, що для її вирішення проблематично використовувати методи прямого пошуку. Це пов'язано з тим, що допустимі значення і обмеження задаються не на неперервних функціях, а на множинах допустимих значень. Тому для вирішення такої задачі слід використовувати методи дискретної оптимізації. Отже, задача оцінки ефективності архітектури конвергентної ССУ може бути сформульована як задача дискретного програмування [45], [77].

4.2 Однокритерійна оптимізація конвергентної ССУ

Однокритерійна задача визначення оптимального поєднання компонентів із заданою ефективністю використання c_j кожного компонента у складі конвергентної архітектури ССУ формулюється як пошук екстремуму цільової функції оцінки ефективності архітектури системи в цілому $f(c)$, де $c = (c_1, \dots, c_m)$, яка задовольняє множині вимог функцій-обмежень $R(r_i)$.

Пряма задача оцінки ефективності архітектури ССУ формулюється як задача мінімізації цільової функції витрат, пов'язаних із використанням композиції компонент у конвергентній системі [12]

$$Z = \min f(c), \quad (4.1)$$

яка повинна задовольняти системі квантифікованих вимог-обмежень

$$\sum_{i=1}^n R(r_i) \geq H$$

Двоїста задача побудови ефективної архітектури визначається позитивним ефектом (максимальною функціональністю) від впровадження відповідної композиції компонент системи і формулюється як задача максимізації функціональних характеристик системи

$$H = \max R(r), \quad (4.2)$$

яка повинна задовольняти обмеженням вартості забезпечення цих характеристик:

$$\sum_{j=1}^m f(c_j) \leq Z$$

У обох випадках використовуються співставні показники ефективності кожної з компонент, які приймають участь в оцінюванні, і при необхідності частковий показник ефективності компонента приводиться до прийнятої основи оцінювання.

Задача оцінки ефективності архітектури ССУ в наведеному формулюванні геометрично інтерпретується як задача пошуку оптимальної кількості вершин політопа (n -мірного багатогранника) [156]:

$$M(V, R) = \{x | Vx = R, x \geq 0\}, \quad (4.3)$$

де $x = (x_{11}, x_{12}, \dots, x_{mn})$.

Якщо на r_j і c_i накласти умови

$$r_j = \sum_{i=1}^m v_{ij} x_{ij}, j = \overline{1, n}, \quad (4.4)$$

$$c_i = \sum_{j=1}^n x_{ij}, i = \overline{1, m}, \quad (4.5)$$

$$x_{ij} \geq 0 \text{ і } x_{ij} = 0 \text{ при } v_{ij} = 0, i = \overline{1, m}, j = \overline{1, n}, \quad (4.6)$$

то задача (1) прийме такий вигляд;

$$f(c) = f(x) = \sum_{i=1}^m f_i \left(\sum_{j=1}^n x_{ij} \right) \rightarrow \min, \quad (4.7)$$

на багатограннику $M(V, R)$, заданому обмеженнями (4.4)–(4.6).

Вектор рішення $x = (x_{11}, x_{12}, \dots, x_{m1}, \dots, x_{mn})$ буде вершиною багатогранника $M(V, R)$ лише тоді, коли в кожному j -му стовпчику матриці $x = [x_{ij}]$ розмірності $m \times n$ міститься один позитивний компонент, а решта компонентів дорівнює нулю. Отже, задача (4.7) стане еквівалентною задачі пошуку оптимальної комбінації

$$\min \{ f(x_p) \mid p \in O_f \}, \quad (4.8)$$

$$\text{де } O_f = \{ p = 1(p_1, p_2, \dots, p_n) \mid v_{pj} > 0, j = 1, \dots, n \}.$$

Для розв'язання задачі (4.8) використовується алгоритм упорядкування виборок з множини O_f за значенням міноранти

$$g(x_p) = \sum_{j=1}^n d_{p_j j} \leq f(x_p), \quad (4.9)$$

де

$$d_{ij} = \frac{r_j f_i \left(\sum_{j \in J_i} \frac{r_j}{v_{ij}} \right)}{v_{ij} \sum_{j \in J_i} \frac{r_j}{v_{ij}}}, \quad J_i = \{j | v_{ij} > 0\}. \quad (4.10)$$

Використання запропонованої моделі оцінки ефективності архітектури системи вимагає попередньої квантифікації висунутих вимог і оцінки показників ефективності застосування альтернативних компонент для побудови ССУ.

Запропонований підхід може бути використаний при визначенні композиції складових елементів для побудови конвергентної ССУ. Такий підхід може бути застосований для створення будь-яких складних інформаційних систем, які будуються шляхом інтеграції компонент різної функціональності, для кожної з яких існують альтернативні варіанти реалізації.

4.3 Багатокритерійна оптимізація конвергентної ССУ

Багатокритерійна задача прийняття рішення стосовно оптимальної архітектури ССУ на основі показників ефективності компонентів полягає у виборі оптимального набору компонентів із множини доступних альтернатив. Множина вибору альтернатив матиме вигляд

$$\Omega = \{X^1(\varphi_1^1, \varphi_2^1, \dots, \varphi_m^1), X^2(\varphi_1^2, \varphi_2^2, \dots, \varphi_m^2), \dots, X^n(\varphi_1^n, \varphi_2^n, \dots, \varphi_m^n)\}, \quad (4.11)$$

де X^i – альтернатива, $i = 1, \dots, n$, φ_j^k – значення j -го критерію для альтернативи X^k , $j = 1, \dots, m$.

Враховуючи неоднозначність різних альтернатив (комбінацій компонентів) при різних умовах використання і, відповідно, відмінних вимог та їх важливості,

слід оцінювати альтернативи, виходячи із балансу критеріїв, які характеризують стан (ефективність) системи, коли значення кожного окремого критерію, що характеризує стан системи, не може бути покращено без погіршення стану інших критеріїв [70]. Такий підхід відповідає принципу Парето, який проголошує, що будь-які зміни, які не наносять шкоди і покращують деякі показники, є позитивними. Тоді ефективність за Парето – це ситуація, коли вичерпані всі можливості покращення одних показників без погіршення інших. Таким чином, множину конфігурацій системи, оптимальних за Парето, називають «множиною Парето», або «множиною альтернатив, оптимальною за Парето», або «множиною оптимальних альтернатив».

Багатокритерійний вибір за Парето визначається функціоналом оцінки ефективності реалізації різних альтернатив системи [293]

$$C(\Omega) = \{x \in \Omega \mid \forall i \in \{1, \dots, m\} [x_i \geq y_i]\}, \quad (4.12)$$

де Ω – множина альтернатив, $X = (x_1, x_2, \dots, x_m)$ – векторна оцінка альтернативи $x \in \Omega$.

Такий вибір дає декомпозицію вихідної багатокритерійної задачі: спочатку визначається область компромісних рішень (множина Парето) X^0 , а потім за допомогою деякого математичного методу остаточний вибір X^* здійснюється вже з X^0 .

Таким доцільним математичним методом є турнірний механізм. Вхідною інформацією для турнірного механізму є множина альтернатив Ω та множина відношень переваги R , яка є результатом вибору за Парето.

Найкраща альтернатива, згідно з цим механізмом, визначається з оцінки ефективності [21]

$$C_T = \{x \mid x \in X, x = \arg \max f(x)\}, \quad (4.13)$$

де

$$f(x) = \sum f_R(x, y), \quad (4.14)$$

де

$$f_R(x) = \begin{cases} 1, & xRy, y\bar{R}x \\ 0, & x\bar{R}y, yRx \\ 1/2, & xRy, yRx \text{ або } x\bar{R}y, y\bar{R}x \end{cases}. \quad (4.15)$$

Тобто спочатку з отриманих відношень R і (4.15) будуюмо таблицю значень $f_R(x)$, потім отримуємо суму $f(x)$ для кожної альтернативи і альтернативи, яким відповідатиме найбільше значення цієї суми, будуть оптимальними.

Перехід від матриці кваліфікацій V і ефективностей використання C до числових значень критеріїв Φ здійснюється як

$$\varphi_i = \frac{\zeta_i}{\sum_{j=1}^n c_j}, \quad (4.16)$$

де φ_i – нормована оцінка i -ої компоненти;

ζ_i – ненормована оцінка i -ої компоненти:

$$\zeta_i = \sum_{j=1}^n (v_{ij} \cdot c_j). \quad (4.17)$$

Загальний показник ефективності $\Phi(X^i)$ альтернативи X^i можна визначити як суму

$$\Phi^i = \sum_{j=1}^n \varphi_j^i . \quad (4.18)$$

Порівнюючи кінцеві оцінки Φ^{ai} , можна зробити остаточний вибір оптимальних конфігурацій компонентів системи електронного навчання.

4.4 Приклад застосування методу багатокритерійної оптимізації

Нехай існують дві групи вимог (обмежень) до системи ситуаційного управління: важливість (рейтинг) платформи і вартість компонентів системи. Оцінки варіантів вибору платформ і вартості компонентів наведені в табл. 4.1 і 4.2. Коефіцієнти важливості $\mathbf{r}$ нормовані до одиниці.

Таблиця 4.1 – Оцінки варіантів вибору платформ реалізації

Група вимог	№	Варіант реалізації	Коефіцієнт важливості ($\mathbf{r}$)
Платформа	1	Windows NT 4	0,7
	2	Windows 2000 Server	0,7
	3	Windows XP	0,9
	4	Windows 2003 Server	0,8
	5	Solaris	0,2
	6	Linux	0,5
	7	Unix	0,4

В табл. 4.2 представлені групи компонентів та їх варіанти реалізації з оцінками їх вартості в абсолютній і відносній нормованій шкалі $r \in [0,1]$.

Таблиця 4.2 – Варіанти вибору компонентів системи

Група компонентів	№	Варіант реалізації	Ціна (у.о)	Нормована оцінка (σ)
HTTP-Server	1	Apache	0	1
	2	IIS	100	0,9
LDAP-каталог	3	Avtive Directory	50	0,95
	4	OpenLDAP	0	1
Java Server	5	WebSphere	200	0,8
	6	WebSphere CE	0	1
DataBase	7	Oracle	800	0,2
	8	Microsoft SQL Server	600	0,4

Отже, з табл. 4.1 маємо множину оцінок важливості вимог $R(r_i)$, $i=1..7$, а з табл. 4.2 – множину оцінок вартості компонентів C_j , $j=1..8$.

Побудуємо матрицю кваліфікацій V , де представлена оцінка використання компоненти j для задоволення вимоги i . Якщо вимога i може бути задоволена компонентом j , то $v_{ij}=1$, в іншому випадку $v_{ij}=0$. В табл. 4.3 наведено матрицю кваліфікацій разом із нормованою оцінкою ефективності використання компоненти за критерієм K_2 , яка розраховується за формулою (4.16).

Таблиця 4.3 – Матриця кваліфікацій

Група компонент	Компонента	Умови та їх коефіцієнти r, v							Сума добутків кваліфікації на коефіцієнт (ς)	Нормована оцінка (σ)
		Windows NT 4	Windows 2000 Server	Windows XP	Windows 2003 Server	Solaris	Linux	Unix		
		0,7	0,7	0,9	0,8	0,2	0,5	0,4		
HTTP-Server	Apache	1	1	1	1	1	1	1	4,2	1
	IIS	1	1	1	1	0	0	0	3,1	0,74
LDAP-каталог	Avtive Directory	1	1	1	1	0	0	0	3,1	0,74
	OpenLDAP	1	1	1	1	0	1	1	4	0,95
Java Server	WebSphere	1	1	1	1	0	1	1	4	0,95
	WebSphere CE	1	0	1	1	0	1	1	3,3	0,79
Data Base	Oracle	1	1	1	1	1	1	1	4,2	1
	Microsoft SQL Server	1	1	1	1	0	0	0	3,1	0,74

Табл. 4.3 представляє двовимірну проєкцію тривимірної таблиці кваліфікованих оцінок альтернатив.

В нашій задачі ефективність визначається позитивним ефектом від застосування відповідних компонент. Отже, задача формулюється як задача максимізації:

$$Z = \max f(c),$$

де $f(c)$ – сумарна ефективність використання компоненти.

Ця задача вирішується як багатокритерійна задача прийняття рішення [21]. В табл. 4.4 наведено комбінації альтернатив разом із значеннями їх критеріїв K_1 і K_2 . Значення критерію K_1 є сумою нормованих оцінок σ від оцінки вартості компонент,

які входять до альтернативи. Коефіцієнт K_2 є сумою нормованих оцінок ефективності використання компонент, які входять до даної альтернативи.

Таблиця 4.4 – Перелік альтернатив

№	Альтернатива	K_1	K_2
1	{OpenLDAP, Apache, WebSphere, Oracle}	2,95	3,69
2	{OpenLDAP, Apache, WebSphere, Microsoft SQL Server}	3,15	2,69
3	{OpenLDAP, Apache, WebSphere Community Edition, Oracle}	3,15	3,53
4	{OpenLDAP, Apache, WebSphere Community Edition, Microsoft SQL Server}	3,35	2,53
5	{OpenLDAP, IIS, WebSphere, Oracle}	3	3,9
6	{OpenLDAP, IIS, WebSphere, Microsoft SQL Server}	3,2	2,9
7	{OpenLDAP, IIS, WebSphere Community Edition, Oracle}	3,2	3,74
8	{OpenLDAP, IIS, WebSphere Community Edition, Microsoft SQL Server}	3,4	2,74
9	{ActiveDirectory, Apache, WebSphere, Oracle }	2,85	3,43
10	{ActiveDirectory, Apache, WebSphere, Microsoft SQL Server}	3,05	2,43
11	{ActiveDirectory, Apache, WebSphere Community Edition, Oracle}	3,05	3,27
12	{ActiveDirectory, Apache, WebSphere Community Edition, Microsoft SQL Server}	3,25	2,27
13	{ActiveDirectory, IIS, WebSphere, Oracle }	2,9	3,64
14	{ActiveDirectory, IIS, WebSphere, Microsoft SQL Server}	3,1	2,64

Продовження таблиці 4.4

15	{ActiveDirectory, IIS, WebSphere Community Edition, Oracle}	3,1	3,48
16	{ActiveDirectory, IIS, WebSphere Community Edition, Microsoft SQL Server}	3,3	2,48

Використовуючи відношення Парето, отримуємо табл. 4.5 переваг альтернатив.

Таблиця 4.5 – Таблиця переваг альтернатив

R°	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
2	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	0	1	1	1	0	0	1	1	0
4	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	1
5	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
6	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
7	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	0
8	0	1	0	1	0	0	0	1	0	1	0	1	0	1	0	1
9	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
13	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
14	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
15	0	0	0	0	0	0	0	0	1	1	1	0	0	1	1	0
16	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1

З табл. 4.5 переваг отримуємо $R = \{<1,1>, <1,9>, <1,13>, <2,2>, <2,10>, <2,14>, <3,2>, <3,3>, <3,9>, <3,10>, <3,11>, <3,14>, <3,15>, <4,4>, <4,10>, <4,12>, <4,16>, <5,1>, <5,5>, <5,9>, <5,13>, <6,2>, <6,6>, <6,10>, <6,14>, <7,1>, <7,2>, <7,3>, <7,6>, <7,7>, <7,9>, <7,10>, <7,11>, <7,13>, <7,14>, <7,15>, <8,2>, <8,4>, <8,8>, <8,10>, <8,12>, <8,14>, <8,16>, <9,9>, <10,10>, <11,10>, <11,11>, <12,12>, <13,9>, <13,13>, <14,10>, <14,14>, <15,9>, <15,10>, <15,11>, <15,14>, <15,14>, <16,10>, <16,12>, <16,16>\}$.

За допомогою турнірного механізму (табл. 4.6) знаходимо значення функції вибору для кожної пари альтернатив.

Таблиця 4.6 – Турнірний механізм

Пара альтерна- тив			f_R	Пара альтерна- тив			f_R	Пара альтерна- тив			f_R	Пара альтерна- тив			f_R
1	1	0,5		2	1	0,5		3	1	0,5		4	1	0,5	
1	2	0,5		2	2	0,5		3	2	1		4	2	0,5	
1	3	0,5		2	3	0		3	3	0,5		4	3	0,5	
1	4	0,5		2	4	0,5		3	4	0,5		4	4	0,5	
1	5	0		2	5	0,5		3	5	0,5		4	5	0,5	
1	6	0,5		2	6	0		3	6	0,5		4	6	0,5	
1	7	0		2	7	0		3	7	0		4	7	0,5	
1	8	0,5		2	8	0		3	8	0,5		4	8	0	
1	9	1		2	9	0,5		3	9	1		4	9	0,5	
1	10	0,5		2	10	1		3	10	1		4	10	1	
1	11	0,5		2	11	0,5		3	11	1		4	11	0,5	
1	12	0,5		2	12	0,5		3	12	0,5		4	12	1	
1	13	1		2	13	0,5		3	13	0,5		4	13	0,5	
1	14	0,5		2	14	1		3	14	1		4	14	0,5	
1	15	0,5		2	15	0,5		3	15	1		4	15	0,5	

Продовження таблиці 4.6

1	16	0,5		2	16	0,5		3	16	0,5		4	16	1
5	1	1		6	1	0,5		7	1	1		8	1	0,5
5	2	0,5		6	2	1		7	2	1		8	2	1
5	3	0,5		6	3	0,5		7	3	1		8	3	0,5
5	4	0,5		6	4	0,5		7	4	0,5		8	4	1
5	5	0,5		6	5	0,5		7	5	0,5		8	5	0,5
5	6	0,5		6	6	0,5		7	6	1		8	6	0,5
5	7	0,5		6	7	0		7	7	0,5		8	7	0,5
5	8	0,5		6	8	0,5		7	8	0,5		8	8	0,5
5	9	1		6	9	0,5		7	9	1		8	9	0,5
5	10	0,5		6	10	1		7	10	1		8	10	1
5	11	0,5		6	11	0,5		7	11	1		8	11	0,5
5	12	0,5		6	12	0,5		7	12	0,5		8	12	1
5	13	1		6	13	0,5		7	13	1		8	13	0,5
5	14	0,5		6	14	1		7	14	1		8	14	1
5	15	0,5		6	15	0,5		7	15	1		8	15	0,5
5	16	0,5		6	16	0,5		7	16	0,5		8	16	1

Оптимальна альтернатива вираховується як

$$C_T = \{x \mid x \in X, x = \arg \max f(x)\},$$

де $f(x) = \sum f_R(x, y)$.

В табл. 4.7 наведено значення $f(x)$ для кожної альтернативи.

Таблиця 4.7 – Результати турнірного механізму

Альтернатива	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$f(x)$	8	7	11	9	9,5	9	13	11	5	3	7	6,5	7	5,5	9	8

Як видно з табл. 4.7, найефективнішою є альтернатива № 7. У табл. 4.8 показано набір компонент, який відповідає цій альтернативі, а також їх сумарна вартість.

Таблиця 4.8 – Оптимальна альтернатива

№	Альтернатива	K_1	K_2	Вартість
7	{OpenLDAP, IIS, WebSphere Community Edition, Oracle }	3,2	3,74	900 у.о.

Даний приклад показує логіку розв’язання задачі пошуку оптимальної множини компонентів.

4.5 Висновки до розділу 4

При побудові конвергентних ССУ на різних етапах процесу ситуаційного управління застосовуються різні набори комп’ютерних компонентів. Різноманіття альтернатив вибору варіантів висуває вимогу вибору найбільш ефективного набору компонентів для розв’язання задач конкретних етапів.

Розв’язання задачі структурної оптимізації конвергентних ССУ забезпечує обґрунтування ефективності архітектурних рішень, що задовольняють конкретним вимогам цільової області використання ССУ. Задача оптимізації конвергентних ССУ розглядається у двох аспектах. Перший аспект – це розв’язання задачі однокритеріальної оптимізації для визначення оптимального набору компонентів системи, що задовольняє інтегрованому критерію ефективності. В якості критеріїв

ефективності обираються функціональність системи та вартість забезпечення цієї функціональності. Отже, Задача визначення оптимальних конфігурацій компонентів розв'язується як пряма оптимізаційна задача максимізації функціональності та двоїста до неї задача мінімізації витрат забезпечення визначеної функціональності.. Для прямої задачі знаходиться максимум цільової функції інтегрального критерію, що визначає функціональність конвергентної ССУ за умов обмежень вартості забезпечення заданої функціональності. Для двоїстої задачі визначається мінімальна вартість забезпечення функціональності не гірше, ніж обумовлено вимогами до системи. Розв'язання однокритеріальної задачі оптимізації дає рішення, щодо компонентного складу та вартості реалізації конвергентної ССУ. Отриманий розв'язок може бути кінцевим у випадку відсутності альтернатив наборів компонентів, що відповідають цьому розв'язку, або попереднім, якщо існують альтернативні набори компонентів.

Для вибору оптимального набору компонентів з декількох альтернатив застосовується багатокритеріальна оптимізація за принципом Парето. Принцип Парето враховує неоднозначність різних альтернатив (комбінацій компонентів) при різних умовах використання і, відповідно, змінних вимогах та їх важливості. За принципом Парето альтернативи оцінюються, виходячи із балансу критеріїв, які характеризують стан (ефективність) системи, коли значення кожного окремого критерію, що характеризує стан системи, не може бути покращено без погіршення стану інших критеріїв. Множину конфігурацій системи, оптимальних за Парето, називають «множиною Парето», або «множиною альтернатив, оптимальною за Парето», або «множиною оптимальних альтернатив». Для формування множини альтернатив, оптимальною за Парето застосовувався турнірний механізм. На прикладі формування складу комп'ютерних компонентів систуаційного центру було показано застосування турнірного механізму.

Запропоновані методи структурної оптимізації можуть бути використані при визначенні композиції складових елементів для побудови конвергентної ССУ. Однокритерійний підхід може бути застосований для створення будь-яких складних

інформаційних систем, які будуються шляхом інтеграції компонент різної функціональності, для кожної з яких існують альтернативні варіанти реалізації.

Багатокритеріальна модель оцінки альтернатив конфігурацій ССУ і методика застосування моделі при оптимізації таких систем дозволяє реалізувати об'єктивний механізм вибору оптимальних альтернатив конфігурацій в умовах часткової невизначеності або зміни системних вимог.

Основні результати розділу опубліковані автором у працях [12, 21, 45, 70, 77].

РОЗДІЛ 5

МОДЕЛІ ДИНАМІЧНОЇ ОПТИМІЗАЦІЇ КОНВЕРГЕНТНИХ ССУ

5.1 Модель діяльності конвергентної ССУ

ССУ відносяться до категорії складних систем через притаманні складним системам такі властивості: відкритість, казуальність, непередбачуваність поведінки в умовах змін (навіть незначних) оточуючого середовища, здатність до самоорганізації, ієрархічність та поліструктурність підрівнів організації системогенезу, здатність до самовідтворення в умовах ієрархічного системогенезу. При системогенезі ССУ враховуються складність структури, ситуативний характер зв'язків між компонентами, ситуативність сценаріїв поведінки за різних умов, велика кількість параметрів і змінних, неповнота і недетермінованість джерел інформації, різноманітність інформаційно-ймовірнісної природи зовнішнього середовища тощо.

Обмежені можливості експериментального дослідження альтернативних варіантів при прийнятті рішень у рамках ССУ вимагають використання адекватних методів моделювання складних процесів, що дозволяє представити в належній формі процеси функціонування складних систем та системи систем. Це також надає можливість описати процеси у цільових системах з використанням математичних моделей для отримання результатів імітаційних експериментів із такими моделями для оцінки параметрів і стану досліджуваних об'єктів. Більше того, на різних етапах створення та застосування прикладних методів для представлення всього різноманіття підсистем метод математичного моделювання та обчислювального експерименту має конкретні специфічні цілі. Ефективність методів моделювання залежить від професійності та аналітичних спроможностей експертів та осіб, які приймають рішення при застосуванні модельного підходу.

Поведінка агентів у МАС описується теорією основних дій (ТОД, Basic Action Theory, BAT) [294], яка формалізує дії агентів. ТОД, як правило, описує дію як поведінку, викликану агентом у певній ситуації. Бажання і переконання агента

визначають його мотивації, наміри та поведінку. Основними поняттями в обчисленні ситуації є ситуації, дії та флюенти. Дії – це те, що змінює динамічний світ від однієї ситуації до іншої під час функціонування агента. Флюенти – це контекстно залежні від ситуації функції, що використовуються для опису впливу дій. Областю визначення флюента є множина усіх можливих ситуацій. Існує два види флюентів: реляційні і функціональні. Щоб описати динаміку домена в ситуаційному численні, потрібно визначити множину дій, доступних для виконання агентами, і множину флюентів, необхідних для опису змін, які відбудуться у світі внаслідок цих дій. Таким чином, ТОД описує ситуаційний аспект діяльності агента.

Мови числення дій та опису дій є засобом подання подій та контекстно керованої поведінки агентів. Мови опису дій є формалізованими мовами для опису наслідків та можливості виконання дій. Найвідомішою мовою дій є PDDL (Planning Domain Definition Language, мова визначення планування домена) та його агент-орієнтована версія MA-PDDL. PDDL [295], [296] на основі SRIPS (Stanford Research Institute Problem Solver) концептуально та в математичній формалізації подається як четвірка $\langle P, O, I, G \rangle$, де P є множиною умов, O являє собою множину операторів (тобто, дії) (кожен оператор є власне четвіркою $\langle \alpha, \beta, \gamma, \delta \rangle$, кожен елемент є множиною умов, де множина α визначає умови, які повинні виконуватись для виконання дії, множина β – умови, які не повинні виконуватись перед виконанням дії, γ – умови, які виконуються після виконання дії, δ – умови, які не виконуються після виконання дії), I є початковим станом, заданим як множина умов, які спочатку є істинними, G – це специфікація цільового стану (подається як задана пара $\langle N, M \rangle$, що вказує, які умови є істинними та хибними, відповідно, для того, щоб стан вважався цільовим). Мови дій містять опис залежних від подій аспектів поведінки агентів. Як наслідок, інтегрована поведінкова модель ситуаційного агента представляється як кортеж [78]

$$A_b = \langle T, P, C, M, D \rangle, \quad (5.1)$$

де T – множина засобів опису ситуації, P – множина засобів управління комунікаціями в змінних середовищах комунікацій, C – множина засобів механізмів координації, M – засоби обміну повідомленнями між агентами, D – множина засобів опису дій.

Модель координованої діяльності створюється для забезпечення належного функціонування системи ситуаційного управління відповідно до її організаційної структури, яка підтримується відповідними ситуаційними агентами. Конвергентні агенти організовані в ієрархії відповідно до рівня абстракції та спеціалізації їх функцій, що забезпечують доступ до послуг у процесі ситуаційного управління. Координація функціонування системи здійснюється конвергентними ситуаційними агентами, на основі керування послугами нижчого рівня агентами вищого порядку та координування послуг на одному рівні процесу управління у разі ситуаційного управління під час вирішення конкретних проблем. Тому, відповідно до контексту ситуаційного управління, поведінка ССУ може бути описана як динаміка руху з точки деякого стану керованої системи у фазовому просторі під впливом ансамблю агентів управління ССУ.

5.2 Метод аналізу динамічної стійкості конвергентної ССУ

ССУ в загальному випадку є нерівноважною відкритою нелінійною динамічною системою. Такі системи називаються дисипативними. Внаслідок самоорганізації у дисипативних системах утворюються стійкі структури, які існують завдяки витрачання (дисипації) системою енергії. Підтримка таких складних систем пов'язана зі зростанням ентропії в середині системи, яка компенсується негативним потоком ентропії з оточуючого середовища. Дисипативні процеси, пов'язані з обміном енергії в середині дисипативної системи та з зовнішнім середовищем, проходять у нерівноважних умовах і наближають систему до стану рівноваги. Поведінка ансамблю агентів може бути інтерпретована в контексті керованої

дисипативної системи, що знаходиться під впливом ентропійних сил. Таким чином, можна стверджувати, що у загальному випадку ССУ має стохастичну природу.

Оскільки керована агентами ССУ розглядається як нерівноважна динамічна стохастична система, то головною метою ансамблю агентів є збереження керованої стохастичної системи у стійкому (стабільному) та передбачуваному стані. Така динамічна стійкість досягається шляхом використання принципу стійкості за Ляпуновим.

За аналогією з термодинамічними системами, рівноважний стан системи визначається мінімумом вільної енергії Гельмгольца (ентропійної енергії). У класичному розумінні нерівноважна динамічна стохастична система як модель фізичної системи, що спрямовується на досягнення максимального значення ентропії [297], еволюціонує через послідовність досягнення макроскопічних станів із вищими рівнями ентропії. Такий процес характеризується формалізмом дії ентропійних сил. Для канонічного ансамблю ентропійна сила F пов'язана з розділом на множину макростанів $\{X\}$ через температуру замкненого простору T , ентропія $S(X)$ пов'язана з макростаном X , X_0 , є поточним макростаном.

Конвергентні агенти ССУ характеризуються певним рівнем інтелекту. Формально інтелект агентів як ентропійну силу F канонічного ансамблю агентів можна визначити таким чином:

$$F(X_0, \tau) = T_C \nabla_X S_C(X, \tau)|_{X_0}, \quad (5.2)$$

де сила F (вибір, реалізація управлінських рішень на основі моделі поведінки (5.1) конвергентного агента) діє так, щоб максимізувати майбутню свободу дій або залишатись із відкритими варіантами управління з певною цілеспрямованістю (знаннями та обізнаністю) T_c , із множиною (різноманіттям) можливих доступних варіантів майбутнього (майбутні ситуації, цільові точки в фазовому просторі) S_c , із певним майбутнім часовим горизонтом (рівнем) τ . $X = \{x_i | i = 1, 2 \dots n\}$ – це множина стохастичних параметрів. Отже, можна стверджувати, що інтелект прагне уникати

будь-яких пасток. Також T_c можна інтерпретувати як параметризацію рівня, на якому шляхи в гіпотетичному динамічному ансамблі всіх можливих фіксованих тривалостей шляхів переходять один в інший, аналогічно переходам між конфігураційними мікростанами ідеального ланцюга, що має відповідну ентропійну пружність, тобто силу, знання та обізнаність агентів.

Метою ситуаційного управління є уникнення можливих кризових ситуацій та утримування керованих параметрів стохастичних дисипативних систем у заданих межах «нормального» стійкого функціонування. Специфічні характеристики та спроможності ансамблю конвергентних агентів визначаються статистичними властивостями керованої системи. Функціонування системи ситуаційного управління ґрунтується на координації її компонентів. Модель координованої діяльності створюється для забезпечення належного функціонування системи ситуаційного управління відповідно до її організаційної структури, яка підтримується відповідними конвергентними агентами. Конвергентні агенти організовані в ієрархії відповідно до рівня абстракції та спеціалізації їх рольових функцій в архітектурі ССУ і забезпечують доступ до сервісів, які підтримують процеси ситуаційного управління. Координація функціонування агентів здійснюється шляхом керування послугами нижчого рівня агентами вищого порядку та координацією сервісів одного рівня у процесі ситуаційного управління при вирішенні конкретних проблем.

Впровадження ситуаційного моделювання (СМ) можна описати моделями, заснованими на проєктному підході та процесах його реалізації. Управління життєвим циклом проєкту є важливою складовою цього підходу. Питання управління життєвим циклом проєктів надається стандартом ISO/IEC/IEEE 16326:2019 – Systems and Software Engineering – Life Cycle Processes – Project Management. Крім того, стандарт ISO 21500-2012 – Guidance on project management є першим стандартом у низці стандартів управління проєктами, встановлює принципи й процедури, в яких представлені найкращі практики управління проєктами, з урахуванням узагальнених знань із управління вимогами та якістю, викладеними у РМВОК.

Життєвий цикл СУ як цільова діяльність реалізується через послідовність процесів: усвідомлення ситуації $\rightarrow$ оцінка ситуації, визначення мети СУ, розробка альтернативних планів $\rightarrow$ підбір плану для досягнення цілей СУ, організація процедур прийняття рішень $\rightarrow$ реалізація плану СУ, моніторинг та коригування плану реалізації СУ $\rightarrow$ оцінка результатів СУ, збереження результатів проведених заходів СУ.

Ситуаційне моделювання в ССУ реалізується як проєктна діяльність, яка вимагає задоволення умов стійкості за Ляпуновим керованої динамічної стохастичної системи [298]. ССУ як динамічна стохастична система керується ансамблем агентів із інтелектом F (5.2). Модель динамічної стохастичної системи характеризується набором стохастичних параметрів $X = \{x_i \mid i = 1, 2 \dots n\}$. Під час ситуаційного управління або життєвого циклу системи комплекс параметрів X еволюціонує як послідовність множин, отриманих у результаті керуючого впливу на керовану систему в цілому. Визначимо різноманітність можливих доступних тенденцій S_C у майбутньому як $\nabla_X S_C(X, \tau) = dx_i/dt$. Агрегована модель динамічної стохастичної системи зі зсувними в часі аргументами має такий вигляд:

$$dx_i/dt = (N_i - x_i) \cdot (a_{i0} + \sum_{j=1}^n a_{ij} \cdot (N_j - x_j)) , \quad (5.3)$$

де x_i – i -й параметр моделі, N_i – максимально можливі значення i -го параметра, a_{ij} – коефіцієнти, які визначаються зі статистики об'єкта, n – кількість параметрів, $i = 1, 2 \dots n$. У загальному випадку a_{ij} може бути деякими функціями від часу, N_i може варіюватися залежно від досліджуваної проблеми (тактичне або стратегічне планування, дослідження критичних ситуацій, прогнозування певних рівнів розвитку, наприклад, бажаних рівнів, параметрів тощо).

За певних умов система може затримуватися або просуватися в часі відповідно до індивідуальних параметрів, наприклад, через те, що в реальній системі кожен елемент керування являє собою сигнал із певним кінцевим рівнем

розповсюдження і обробки інформації, що також займає деякий час. Тому, беручи до уваги ці особливості, можна вводити затримки і випередження для кожного параметра керованої системи. Очевидно, що, у свою чергу, часові затримки та випередження можуть бути також функціями часу, що ускладнює завдання керування. Враховуючи перше наближення щодо постійних часових затримок (зсувів часу), можна подати агреговану модель досліджуваного об'єкта у такому вигляді:

$$\begin{bmatrix} dz_1/dt \\ dz_2/dt \\ \dots \\ dz_{M-1}/dt \\ dz_M/dt \end{bmatrix} = \begin{bmatrix} z_1(t - \tau_{10}) \\ z_2(t - \tau_{20}) \\ \dots \\ z_{M-1}(t - \tau_{(M-1)0}) \\ z_M(t - \tau_M) \end{bmatrix} \cdot \begin{bmatrix} b_{10} + \sum_{j=1}^M b_{1j} \cdot z_j \cdot (t - \tau_{1j}) \\ b_{20} + \sum_{j=1}^M b_{2j} \cdot z_j \cdot (t - \tau_{2j}) \\ \dots \\ b_{(M-1)0} + \sum_{j=1}^M b_{(M-1)j} \cdot z_j \cdot (t - \tau_{(M-1)j}) \\ b_{M0} + \sum_{j=1}^M b_{Mj} \cdot z_j \cdot (t - \tau_{Mj}) \end{bmatrix}, \quad (5.4)$$

де z_i – відхилення i -го параметра x_i агрегованої моделі системи, τ_{ij} – постійні зсуви часу (затримки) з діапазону майбутнього часового горизонту τ (5.2) для відповідних змінних, b_{ij} – коефіцієнт для відповідного z_i (відхилення i -го параметра), який визначається зі статистики динамічної поведінки об'єктів, M – кількість параметрів, $i = 1, 2 \dots M$. Попереднє значення може бути введено так само, як зміна знаку на протилежний.

Математична модель (5.4) описує еволюцію, наприклад, динамічного нестійкого в часі об'єкта залежно від його попередньої історії функціонування. Система рівнянь (5.4) може бути перетворена та спрощена в обмеженому околі значення поточного моменту часу t , якщо застосовувати розклад у ряд Тейлора для функцій $z_i(t - \tau_{ij})$. Для цього використовуємо теорему Ельсгольца, враховуючи лише члени першого порядку ряду Тейлора відносно малих величин:

$$\tau_{ij} : z_i(t - \tau_{ij}) \approx z_i(t) - \tau_{ij} \dot{z}_i, \quad (\dot{z} \equiv dz / dt)$$

Відповідно до теореми Ельсгольца, членам вищого порядку притаманна менша точність при розкладанні в ряд. Тому всі змінні в системі будуть апроксимуватися функціями лише однієї змінної часу t . Якщо розглянути задачу оптимального керування [299], [300] для динамічного нестійкого в часі об'єкта за даними критеріями, то можна подати Гамільтоніан системи (5.4) у вигляді

$$H(z, p) = \sum_{i=1}^M p_i z_i \sum_{j=0}^M b_{ij} z_j, \quad (5.5)$$

де $z_0 = 1$, p_i у виразі (5.5) визначаються з системи рівнянь

$$dp_i/dt = -(p_i \sum_{j=0}^M b_{ij} z_j + \sum_{j=1}^M p_j z_j b_{ji}) \quad (5.6)$$

Тоді проблема оптимального управління динамічного нестійкого в часі об'єкта полягає в обчисленні параметрів $u_m(t)$ ($m=1, 2, \dots, M$) на основі таких рівнянь:

$$dz_i/dt = f_i(z_1, \dots, z_M, u_1, \dots, u_M), \quad (5.7)$$

де $u_i(t)$ в (5.7) можна вважати позитивними напрямними функціями дискретного часу (або $b_{ij}(t)$ у (5.6), відповідно).

Функціональна якість (інтелект агентів F , визначений рівнянням (5.2)) може бути представлена як

$$F = \int_0^{\tau} f_0(z_i, u_m) dt \quad (5.8)$$

або

$$\dot{z}_0 = f_o(z_i, u_m), z_o(0) = 0. \quad (5.9)$$

Оптимальне ситуаційне управління, згідно з (5.8), (5.9), відбувається за значеннями $u_m(t)$, що задовольняють (5.8), (5.9) та умові $H(z_i, p_i, u_m) = 0$. Зауважимо, що принцип максимуму Понтрягіна дає лише необхідну, але не достатню умову для оптимального управління протягом ситуаційного управління системою. Тому можуть бути прийняті інші альтернативні рішення для вирішення проблеми або рішення може бути відсутнє взагалі.

Розробка, прийняття та підтримка ефективних управлінських рішень відбуваються на основі конкретних ситуаційних параметрів та показників керованої системи. Ситуаційне управління як технологія інформаційної та модельної підтримки прийняття рішень у процесі управління складними об'єктами забезпечує збір, обробку, зберігання, організацію доступу та інтерпретацію даних. Ця технологія спрямована на аналіз та прогнозування економічних, екологічних, соціальних та інших проблем, зокрема, розробку планів стратегічного розвитку на основі статистично узагальнених теоретичних та емпіричних підходів, виявлення, оцінку та застосування нових спроможностей керованої системи.

Підтримка функцій моделювання агентами агентно-орієнтованої системи дозволяє формувати адекватну поведінку у процесі ситуаційного управління в умовах змін оточуючого середовища.

5.3 Метод забезпечення технічної інтероперабельності конвергентної ССУ

При впровадженні інформаційно-аналітичного процесу обробки інформації під час підготовки альтернатив та прийняття рішень у конвергентній ССУ виникає задача управління послідовністю потоків обробки інформації таким чином, щоб між

агентами конвергентної ССУ, пов'язаних із однією і тією ж проблемою, забезпечувався необхідний рівень інтегрованості в процесах ситуаційного управління з мінімальними витратами часу.

Розглянемо розподілену агентну ССУ. Нехай M – множина агентів агентної ССУ, n – кількість видів ресурсів сервісів (у нашому випадку $|M| > n$, тобто ресурси сервісу можуть надаватись одному або декільком агентам), c_k – оцінка інформаційних потреб k -го ($k \in M$) агента, R_i – оцінка обсягу i -го виду ресурсу у розподіленій ССУ, P_{ij} – потік перерозподілу запитів ресурсів між i -им та j -им видами ресурсів ($i, j = 1, 2, \dots, n$), q – максимально допустимий потік перерозподілу між двома ресурсами.

Введемо змінну

$$x_{ki} = \begin{cases} 1, \text{ якщо } k \text{ – а задача обслуговується } i \text{ – м агентом, } i = 1, \dots, n \\ 0, \text{ в іншому випадку} \end{cases}.$$

Формалізована модель фактичного використання H_i i -го ресурсу ($i = 1, 2, \dots, n$) у розподіленій ССУ в процесі використання агентами має такий вигляд [3], [4]:

$$H_i = \max_i \sum_{k \in M} c_k x_{ki} \rightarrow \min, \quad (5.10)$$

$$\sum_{k \in M} c_k x_{ki} \leq R_i, i = 1, 2, \dots, n, \quad (5.11)$$

$$\frac{1}{M} \sum_{i=1}^n x_{ki} \leq 1, k \in M, \quad (5.12)$$

$$P_{ij}(x_{ki}, x_{kj}) \leq q; k \in M; i \neq j; i, j = 1, 2, \dots, n. \quad (5.13)$$

Умова (5.11) виражає обмеження на обсяг i -го ресурсу, умова (5.12) – це вимога обов'язкового обслуговування всіх агентів та належність агента до групи обслуговування окремим локальним сервісом, умова (5.13) регулює потоки перерозподілу ресурсів між агентами розподіленої ССУ.

Задача представлена співвідношеннями (5.10)–(5.13) – це задача булевого цілочисельного програмування. Для розв'язання задачі (5.10)–(5.13) запропоновано алгоритм, який реалізує метод пошуку з локальною оптимізацією, що дозволяє отримати розв'язок задачі (5.10)–(5.13) з прийнятною точністю для практичної роботи розподіленої агентної ССУ.

Вихідні умови для розв'язання задачі (5.10)–(5.13) формуються у вигляді матриці A . Рядки матриці являють собою оцінку ресурсних потреб агентів, які використовують сервіси конвергентної ССУ. Стовпці матриці A – це фактичне використання ресурсів конвергентної ССУ, тобто H_i ($i = 1, 2, \dots, n$).

Алгоритм розв'язання задачі обслуговування агентів у ССУ включає такі п'ять кроків.

Крок 1. Якщо всі стовпці матриці A переглянуті, переходимо до кроку 5. У іншому випадку переходимо до наступного кроку. При аналізі в матриці A шукаємо стовпчик із максимальною сумою рядків, тобто з максимальним використання конкретного ресурсу. Нехай таким стовпцем буде стовпець i . Наступним кроком є закріплення області з трьох стовпців $i, i+1, i-1$ ($i = 1, 2, \dots, n$). Якщо $i+1$ або $i-1$ не існують (для кінцевих елементів), тоді колонка та її окіл будуть усічені.

Крок 2. У i -му стовпці знаходимо ненульовий елемент $c_k x_{ki}$ і перевіряється можливість зсуву цього елемента вправо або вліво відповідно до умов (5.11)–(5.13). Якщо такі зсуви можливі, то вони виконуються. Для запам'ятовування переміщень елементів матриці A на наступному кроці алгоритму (з метою визначення можливих переміщень на наступному кроці) вводимо допоміжну матрицю B , елементами якої є числа з множини $(-1, 0, 1)$. Початкові значення елементів матриці рівні 0. При переміщенні елемента $c_k x_{ki}$ до розміщення елемента $c_k x_{ki-1}$ елемент e_{ki} матриці B приймає значення -1 , при переміщенні $c_k x_{ki}$ до розміщення $c_k x_{ki+1}$ елемент e_{ki}

приймає значення $+1$, інакше значення елемента e_{ki} залишається рівним 0 . Якщо наступні ітерації алгоритму потребують зсуву деякого елемента $-c_k x_{ki-1}$ матриці A на місце елемента $c_k x_{ki}$ і елемент $-e_{ki}$ матриці B отримує значення -1 , що також вимагає переміщення елемента $-c_k x_{ki+1}$ матриці A до розміщення елемента $c_k x_{ki}$, а елемент $-e_{ki}$ матриці B стає рівним $+1$, то ці переміщення не відбуваються і переходять до ряду $k+1$ матриці A .

Крок 3. На цьому кроці перевіряється, який із трьох варіантів (початковий та два отримані) дає найкращий результат за критерієм (5.10). Якщо зміщення було можливе лише вліво або лише вправо, то обидва варіанти (початковий і отриманий) підлягають верифікації. Якщо на m -й ітерації виявиться, що $H_i^m = H_i^{m-1}$, то процес завершується і H^m приймається як наближений розв'язок для стовпця i . Відповідні зміни вносяться в матрицю A , матрицю B і вектор $H = \{H_i\}$ ($i = 1, 2, \dots, n$). Таким чином, i -й стовпець виключається з розгляду і переходимо до кроку 1 алгоритму. В іншому випадку виконується крок 4.

Крок 4. Переходимо до рядка $k+1$ матриці A та кроку 2 алгоритму. Якщо всі рядки були переглянуті, то процес закінчується, значення H_i^k приймається як наближене рішення для задачі з розв'язаним стовпцем i . Відповідні зміни вносяться в матрицю A , матрицю B і вектор $H = \{H_i\}$ ($i = 1, 2, \dots, n$). Таким чином, i -й стовпець виключається з розгляду і виконується крок 1 алгоритму.

Крок 5. Після перегляду всіх стовпців матриці A розв'язок задачі (5.10)–(5.13) завершується. Отримано нову розмітку для агентів, що використовують ресурс окремого сервісу.

Вищенаведений алгоритм розв'язку задачі (5.10)–(5.13) обслуговування агентів у конвергентній ССУ є скінченним завдяки обмеженій кількості сервісів у ній.

Результати алгоритму перерозподілу використання ресурсів між 9 сервісами конвергентної ССУ для перших 5 ітерацій алгоритму представлені на рис. 5.1. Номер графіка на діаграмі відповідає номеру ітерації. Горизонтальна вісь показує

номер сервісу у розподіленій конвергентній ССУ, уздовж вертикальної осі – об'єм номінального об'єму ресурсу для розв'язання конкретної задачі в процесі ситуаційного управління, яку слід обробити протягом регламентованого проміжку часу.

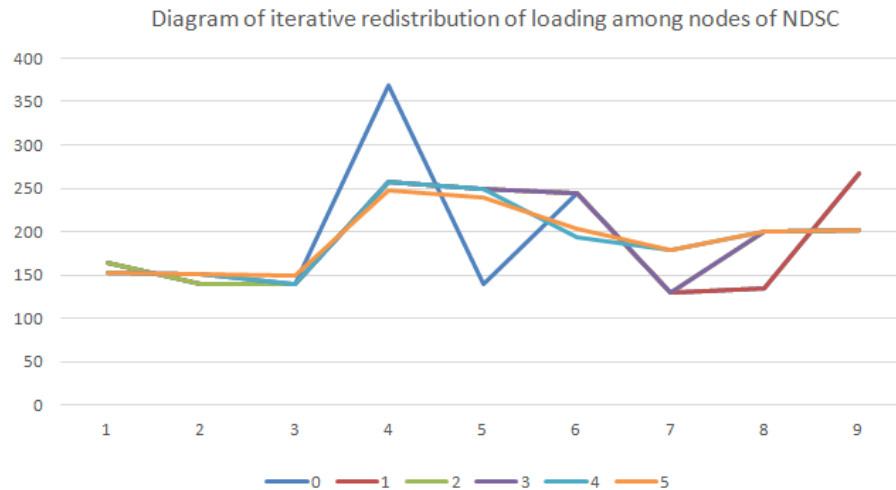


Рисунок 5.1 – Результати для 5 ітерацій перерозподілу ресурсів
для розподіленої ССУ з 9 сервісами

Таким чином, алгоритм розв'язку задачі (5.10)–(5.13) оптимізації споживання ресурсів для ССУ на основі методу булевого програмування дозволяє якісно підвищити рівень забезпечення сервісами агентів у розподіленій ССУ і тим самим підвищити ефективність її функціонування. Наведений приклад є прототипом моделі обслуговування агентів у розподіленій конвергентній ССУ.

В процесі обробки інформації при використанні відповідних сервісів виникає задача управління послідовністю обробки інформаційних та керуючих потоків, завдяки чому комплекс сервісів забезпечує максимальну ефективність ресурсного забезпечення агентів ССУ з мінімальними витратами часу [301]. Така задача є двоїстою до прямої задачі, алгоритм розв'язку якої було наведено вище. Запит на обслуговування визначає потребу в ресурсах кожного окремого сервісу. Ця задача формулюється таким чином.

Розглянемо спільні розподілені інформаційні ресурси конвергентної ССУ, які надаються n сервісами $A_1, A_2, \dots, A_n$. Система сервісів забезпечує виконання запиту

обслуговування для агентів ССУ. Запит вважається обслуговуваним, якщо він обслуговується кожним локальним сервісом послідовно, починаючи з сервісу A_1 . Припустимо, що час обслуговування запиту агента i -м сервісом є незалежною випадковою величиною з функцією розподілу $F^i(t)$ та щільністю розподілу $\varphi_i(t) = F^i(t)$. Запит обслуговується сервісом A_1 одразу після його надходження. На момент надходження запиту сервісу A_2 можливі два варіанта:

1) запит агента був обслужений сервісом A_1 раніше ніж залученим до обслуговування сервісом A_2 і агент очікує звільнення ресурсу сервісу A_2 . За очікування обслуговування запиту накладається певний штраф;

2) запит агента ще знаходиться в процесі обслуговування сервісом A_1 коли надсилається запит на обслуговування сервісом A_2 , тобто сервіс A_2 очікує завершення обслуговування запиту сервісом A_1 . У такому випадку за затримку сервісу A_2 накладається штраф. У цьому випадку здійснюється перерозподіл ресурсів між іншими сервісами.

Очевидно, що штраф f_i є випадковою величиною. Необхідно визначити такі моменти $x_1, x_2, \dots, x_n$ підключення сервісів $A_2, A_3, \dots, A_n$, коли математичне сподівання загального штрафу, що дорівнює сумі накладених штрафів через невідповідність моментів x_i підключення i -го сервісу моментам η_{i-1} припинення обслуговування запиту в i -м сервісом, буде мінімальним.

Нехай α – питома вартість очікування виконання запиту агента, а β – питома вартість простою сервісу. Тоді штрафні санкції пов'язані з розбіжністю між моментами x_i надання ресурсу сервісу для обслуговування запиту та моментами η_{i-1} завершення обслуговування запиту агента $(i-1)$ -м сервісом визначається як

$$f_1(\eta_1, x_2) = \begin{cases} \alpha(x_2 - \eta_1), & \eta_1 \leq x_2 \\ \beta(\eta_1 - x_2), & \eta_1 \geq x_2 \end{cases}, \quad (5.14)$$

де $\eta_1 = \tau_1$,

$$f_2(\eta_2, x_3) = \begin{cases} \alpha(x_3 - \eta_2), \eta_2 \leq x_3 \\ \beta(\eta_2 - x_3), \eta_2 \geq x_3 \end{cases}, \quad (5.15)$$

$$\text{де } \eta_2 = \max(\eta_1, x_2) + \tau_2,$$

$$f_{n-1}(\eta_{n-1}, x_n) = \begin{cases} \alpha(x_n - \eta_{n-1}), \eta_{n-1} \leq x_n \\ \beta(\eta_{n-1} - x_n), \eta_{n-1} \geq x_n \end{cases}, \quad (5.16)$$

$$\text{де } \eta_{n-1} = \max(\eta_{n-2}, x_{n-1}) + \tau_{n-1},$$

$$Mf_1(\eta_1, x_2) = (\alpha + \beta) \int_0^{x_2} F(u_1) du_1 + \beta(M_{\tau_1} - x_2), \quad (5.17)$$

$$\begin{aligned} Mf_2(\eta_2, x_3) = & (\alpha + \beta) \int_{x_2}^{x_3} \left(\int_{x_2}^{u_3} F(u_1) \varphi_2(u_2 - u_1) du_1 + \beta(M_{\tau_1} - x_2) \right) du_2 + \\ & + \beta \left(M_{\tau_1} + M_{\tau_2} + \int_0^{x_2} F(u_1) du_1 - x_3 \right) \end{aligned}, \quad (5.18)$$

$$\begin{aligned} Mf_3(\eta_3, x_4) = & (\alpha + \beta) \int_{x_3}^{x_4} \left(\int_{x_3}^{u_3} \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) \varphi_3(u_3 - u_2) du_2 \right) du_3 + \\ & + \beta \left(M_{\tau_1} + M_{\tau_2} + M_{\tau_3} + \int_0^{x_2} F(u_1) du_1 + \int_{x_2}^{x_3} \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) du_2 - x_4 \right), \end{aligned} \quad (5.19)$$

$$\begin{aligned} Mf_{n-1}(\eta_{n-1}, x_n) = & (\alpha + \beta) \int_{x_{n-1}}^{x_n} \left(\dots \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) \dots \right) du_{n-1} + \\ & + \beta \left(M_{\tau_1} + \dots + M_{\tau_n} + \int_0^{x_2} F(u_1) du_1 + \int_{x_{n-2}}^{x_{n-1}} \left(\dots \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) \dots \right) du_{n-2} - x_n \right). \end{aligned} \quad (5.20)$$

Загальний штраф визначається як

$$F(x) = \sum_{i=2} Mf_{i-1}(\eta_{i-1}, x_i) \quad (5.21)$$

Використання класичного підходу для мінімізації функції $F(x)$ вимагає обчислення множинних інтегралів і значень випадкового розподілу $F_i(t)$ за відповідними законами, що не завжди можливо. Однак ці труднощі можна усунути, використовуючи методи стохастичного програмування, з точки зору яких переформулюємо задачу. Задача у новій постановці полягає у пошуку мінімуму функції

$$F(x) = Mf(\eta, x) \quad (5.22)$$

за умов

$$x_{i+1} \geq x_i \geq 0, \quad i = 2, 3, \dots, n, \quad (5.23)$$

де

$$f(\eta, x) = \alpha \sum_{i \in I} (x_i - \eta_{i-1}) + \beta \sum_{j \in J} (\eta_{j-1} - x_j), \quad (5.24)$$

причому

$$\eta_i = \max(\eta_{i-1}, x_i) + \tau_i, \quad i = 2, 3, \dots, n.$$

Зауважимо, що функція (5.24) опукла відносно x_i для фіксованих η_{i-1} .

Задачі (5.22)–(5.24) є особливим випадком задачі стохастичного програмування. Для їх розв'язання може бути використаний стохастичний квазіградієнтний метод проєкції, який описується процедурою

$$x_i^{s+1} = \pi_x(x^s - \rho_s W_i^s), s = 0, 1, \dots, j; i = 2, 3, \dots, n, \quad (5.25)$$

де s – номер ітерації, X – область проєкції, що задовольняє обмеженням (5.23), $\pi_x(\bullet)$ – оператор, що відображає кожну точку $x \in R^n$ у точку $\pi_x(x) \in X$ так, що $\|x - \pi_x(x)\|^2 \leq \|y - x\|^2$ для будь-якого $y \in X$, ρ_s – коефіцієнт (множник) кроку, W_i^s – випадковий вектор такий, що $M(W_i^s | x_i^0, x_i^1, \dots, x_i^s) = F_x^A(x_i^s)$, $s = 0, 1, \dots, j; i = 2, \dots, n$, де $F_x^A(x_i^s)$ – градієнт (узагальнений градієнт) функції $F(x)$. Послідовність $\{x_i^s\}$, $s = 0, 1, \dots, j$, визначена процедурою (5.16), збігається до розв'язку задач (5.22), (5.23).

Обчислювальна схема вирішення задачі формулюється таким чином. Нехай x^s отримано на s -й ітерації (початкове значення x^0 задається). Потрібно зробити наступні кроки для отримання $(s+1)$ -го наближення.

Крок 1. Спостерігаємо реалізацію x_i^s значення x_i .

Крок 2. Обчислюємо вектор W_i^s .

Крок 3. В результаті виконання операції проєкції на область $X = \{x_i : x_i \geq x_{i-1} \geq 0\}$ наступне наближення визначається як

$$x_i^{s+1} = \max(0, \max(x_{i-1}^s, x_i^s - \rho_s W_i^s)), s = 0, 1, \dots, j; i = 2, 3, \dots, n.$$

Наведений вище алгоритм легко реалізувати. Він не залежить від типу закону розподілу випадкового часу обслуговування запитів агентів.

Використання значення згладженого середнього

$$Z^{s+1} = Z^s + 1/W^s (f(\eta^s, x^s) - Z^s)$$

дає оцінку математичного сподівання випадкової функції $f(\eta^s, x^s)$.

5.4 Висновки до розділу 5

Динамічна оптимізація конвергентних ССУ розглядається у двох аспектах. Перший аспект – це забезпечення динамічної стійкості функціонування ССУ як дисипитивної системи. Другий аспект – це забезпечення оптимальної інтероперабельності між агентами в рамках цільової ССУ.

Динамічна стійкість конвергентної ССУ, що розглядається як дисипитивна система забезпечується шляхом функціонування ансамблю агентів, здатних отримувати інформацію із зовнішнього середовища і, таким чином, збільшувати загальний рівень ентропії системи, що надає системі більшу свободу дій. Запропоновано модель та метод для дослідження стійкості динамічної стохастичної системи на основі принципу стійкості за Ляпуновим, представлену у формі системи звичайних диференціальних рівнянь. Запропоновано та досліджено агентно-орієнтований підхід для включення моделі стійкості за Ляпуновим у процес ситуаційного управління динамічною стохастичною системою як цільової проєктної діяльності.

Запропоновано методи розв'язання задачі оптимізації технічної інтероперабельності в конвергентній ССУ дозволяють забезпечити ефективне завантаження та мінімізацію часу обслуговування агентів у розподілених обчислювальних системах шляхом розв'язання прямої та двоїстої задач математичного програмування.

Для розв'язання прямої задачі обґрунтовано використання методу булевого програмування та запропоновано алгоритм його реалізації. Для розв'язання двоїстої задачі обґрунтовано використання методу стохастичного програмування та запропоновано алгоритм його реалізації. Алгоритм, визначений для управління використанням ресурсів сервісів, є ефективним інструментом управління агентними ССУ з метою забезпечення ефективного розподілу ресурсів сервісів між агентами розподіленої ССУ з мінімізацією часу обробки інформації.

Розроблені моделі інтеграції поведінкових та координаційних аспектів агентів на основі знань можуть бути використані при розробці систем та технологій ситуаційного управління у різних цільових сферах ситуаційного управління. Запропоновані моделі підтримки технічної інтеоперабельності та алгоритми управління середовищем конвергентної ССУ забезпечують можливості ефективного управління спроможностями складних ситуаційних систем.

Основні результати розділу опубліковані автором у працях [3, 4, 5, 24, 27, 29, 78].

РОЗДІЛ 6

ЗАСОБИ ПІДТРИМКИ ПОБУДОВИ КОНВЕРГЕНТНИХ СИСТЕМ

6.1 Засоби оптимізації конфігурацій конвергентних систем

Одним із важливих етапів побудови ССУ є підготовка персоналу і користувачів до використання ССУ. Така підготовка проводиться з використанням конвергентних систем електронного навчання (e-learning). Основою технологій електронного навчання є складні програмно-технічні комплекси, які забезпечують організаційну підтримку, широку функціональність, гнучкість і масштабованість середовищ електронного навчання. Різноманітність цілей, задач, очікуваних результатів і економічних факторів застосування систем електронного навчання вимагає створення і супроводження архітектури системи електронного навчання, яка буде адекватною її цільовому призначенню. Тому сучасний підхід до побудови систем електронного навчання ґрунтується на модульному підході, що забезпечує задоволення найрізноманітніших вимог до систем такого типу.

Популярною системою електронного навчання являється система IBM Lotus Learning Management System (LMS) [302], яка є гнучким і масштабованим продуктом. Шляхом розділення можливостей між різними компонентами та включення різноманітних зовнішніх додатків LMS дозволяє користувачам різними способами конфігурувати систему електронного навчання в залежності від конкретних потреб користувача.

Однією з найголовніших переваг IBM Lotus LMS є те, що вона дозволяє отримати переваги існуючих додатків, таких як каталог LDAP або реляційна СУБД, так само, як і від уже працюючих інших продуктів IBM та інших виробників. Комбінування різноманітних технологій робить LMS придатною для задоволення найрізноманітніших потреб навчання.

Організація, яка здійснює розгортання і супроводження систем електронного навчання, потребує інформаційної підтримки своєї діяльності, зокрема, зберігати параметри розгорнутих систем, мати інструмент для обґрунтування необхідної

обчислювальної потужності системи та її структури в залежності від вимог до системи. Ефективна інформаційна підтримка діяльності може здійснюватись із використанням системи адміністрування проєктів дистанційного навчання, яка б дозволяла вести декілька проєктів, зберігати всі перед- та постінсталяційні дані, розраховувати необхідні параметри та забезпечити вирішення типових проблем на основі покрокового виконання інструкцій. Вона також забезпечує двосторонній зв'язок із клієнтами – власниками встановлених систем електронного навчання, які можуть отримати всі необхідні установчі дані про свій проєкт у системі адміністрування проєктів дистанційного навчання, переглядаючи їх та змінюючи за необхідністю.

Мінімальна конфігурація системи електронного навчання включає такі компоненти:

- сервер LMS;
- сервер LDAP (Lightweight Directory Access Protocol – мережева служба управління доступом до каталогів);
- сервер контенту;
- сервер доставки (Delivery Server – DS);
- сервер баз даних;
- сервер FTP (файловий сервер).

Система електронного навчання може включати додаткові компоненти, які розширюють її функціональність:

- сервер електронної пошти (SMTP-сервер);
- сервер обговорень;
- сервер конференцій;
- пошуковий сервер;
- сервер інтерактивної віртуальної аудиторії (virtual classroom);
- авторські засоби розробки навчальних курсів;
- утиліти імпорту стандартних навчальних курсів.

При встановленні масштабу системи електронного корпоративного навчання слід визначити:

- загальну кількість слухачів, зареєстрованих у корпоративному каталозі LDAP;
- загальну кількість слухачів, яку можна визначити у каталозі LDAP;
- кількість одночасно працюючих користувачів;
- середню кількість відвідувань на день;
- середню відвідуваність на годину;
- чи є дана конфігурація об'єктом постійного завантаження, чи вона призначена для підтримки неочікуваних сплесків інтенсивності використання;
- відношення пікового робочого навантаження до середнього навантаження;
- максимальний прийнятний час відгуку (який сприймається через браузер);
- очікуваний час реакції інтерфейсу користувача під час навігації (переміщення) (зазвичай 20 с);
- очікуваний час роздумування, пов'язаний із зчитуванням сторінок контенту (зазвичай 3 хв);
- кількість сторінок контенту, що зчитуються слухачами за час відвідування;
- параметри апаратної платформи або інформація про існуючу апаратну конфігурацію: модель платформи, кількість процесорів, швидкість процесорів, RAM, ОС;
- вимоги до СУБД або параметри існуючого сервера баз даних, до якого буде під'єднано LMS: модель платформи, кількість процесорів, швидкість процесорів, RAM, ОС;
- вимоги до LDAP або інформація про встановлений LDAP-сервер, до якого буде під'єднано LMS: модель платформи, кількість процесорів, швидкість процесорів, RAM, ОС;
- максимальне завантаження процесора, яке потрібно передбачити для роботи сервера додатків (50%, 60%, 70%, 80%);
- кількість курсів у системі;
- деякі деталі, що стосуються контенту: розмір, середовище (html, Windows, netG) тощо;

- чи розміщені сервери доставки (DS) централізовано разом із LMM (Learning Management Module – модуль управління процесом навчання), чи географічно розподілені;

- доступні мережеві з'єднання між сайтами користувачів;

- яким чином користувачі приєднані до мережі: LAN чи dial-up (швидкість модемів);

- звітні елементи (наприклад, щомісячні звіти всіх менеджерів).

Фактори, що впливають на ефективність реалізації системи електронного навчання, визначені у [21].

Усі конфігураційні параметри встановлюваної (або вже встановленої) системи електронного навчання співставляються з певним проєктом дистанційного навчання. Він включає в себе як загальні дані всієї системи (замовник, контактні особи, адміністратор, опис), так і дані по кожному з компонентів (Каталог LDAP, Сервер Доставки, LMM тощо).

З накопиченням встановлених систем дистанційного навчання виникає необхідність автоматизації адміністрування цих проєктів із можливістю швидкого й розподіленого доступу до параметрів проєктів, а також із здатністю динамічного перерахування параметрів, що залежать від введених.

Першочерговим завданням системи автоматизованого управління проєктами дистанційного навчання є збереження даних кожного із проєктів, забезпечення користувачам розподіленого доступу до них та керування усім цим процесом.

У системі автоматизованого управління проєктами необхідно виділити два типи користувачів: адміністратори та звичайні користувачі.

До прав адміністратора входять такі функції:

- створення нового проєкту;

- повне редагування даних існуючого проєкту;

- розрахунок орієнтовної конфігурації системи від введених передінсталяційних даних;

- реєстрація нового користувача;

- управління доступом користувачів до проєктів.

У звичайного користувача системи може бути право перегляду даних своєї системи, власником якої він є, та з дозволу адміністратора – редагування певних параметрів. Також усім користувачам має бути відкрито доступ до програми покрокового вирішення типових проблем.

Після створення нового проєкту і заповнення його попередніми даними необхідно приступити до налагодження складу компонентів. Певний набір компонентів встановлюється автоматично після створення проєкту і видаленню не підлягає. Інші компоненти додаються і видаляються користувачем. Також налагоджуються зв'язки між ними.

Кожна компонента має свій перелік даних. Практично всі дані повинні мати можливість редагуватись користувачем із відповідними правами. Дані компоненти поділяються на передінсталяційні та постінсталяційні. Передінсталяційні дані є результатом збору інформації про майбутню систему, а постінсталяційні отримуються вже після її встановлення.

Відповідно до введених даних проєкту, попередньої конфігурації компонентів та прогнозованих вимог, розраховуються вимоги до потужності апаратних засобів і видається рекомендація щодо способу розподілення системи.

З виникненням певних проблем у процесі встановлення та користування системою дистанційного навчання можна звернутись до програми вирішення типових проблем із покроковими інструкціями і діагностикою коректності даних проєкту.

Усі події, пов'язані зі зміною будь-яких параметрів, повинні відслідковуватись і записуватись у журнал подій, доступ до якого буде відкрито для адміністраторів.

По кожному проєкту повинна бути надана можливість отримання повної інформації у зручному для читання форматі.

Функціонально система адміністрування складається з таких основних модулів:

- модуль ідентифікації й аутентифікації, призначений для визначення користувача і його прав доступу;

- модуль користувачів, призначений для перегляду та внесення змін в їхні дані та дозволи (доступний лише адміністраторам);
- модуль проєктів, призначений для перегляду та редагування даних проєкту та його компонентів. Також цей модуль призначений для математичних розрахунків проєкту;
- модуль керування правами користувачів;
- журнал подій.

Ієрархію модулів системи оптимізації конвергентної системи показано на рис. 6.1 [66].

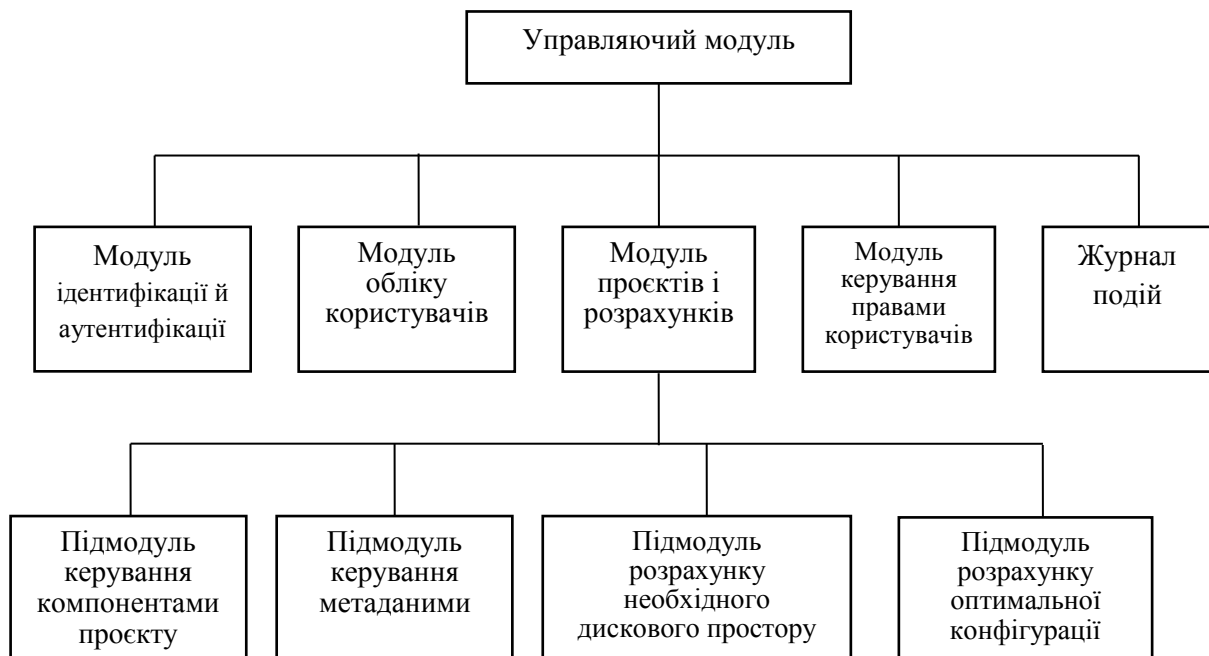


Рисунок 6.1 – Структура системи оптимізації конвергентної системи

Використання системи адміністрування проєктами електронного навчання дозволяє забезпечити ефективне керування та супроводження проєктів, отримувати оперативну інформацію про стан і конфігурацію проєкту, обґрунтувати вибір архітектурних рішень та безперервність зв'язків із клієнтами. Внутрішні засоби опису метаданих проєктів забезпечують гнучкість описів різних систем і адаптацію конкретних проєктів під вимоги користувачів.

Модель даних засобу оптимізації конвергентної системи наведено на рис. 6.2.

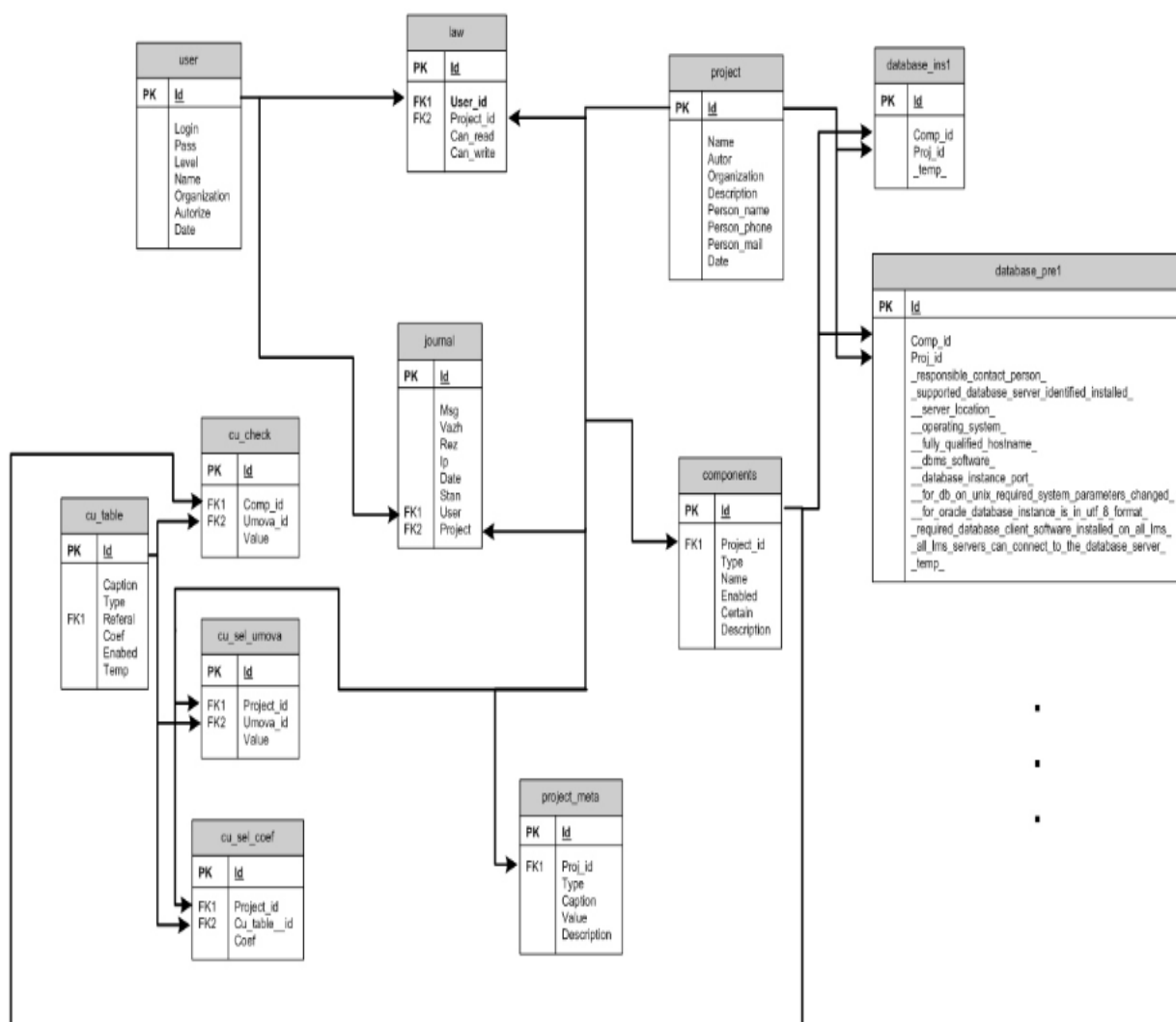


Рисунок 6.2 – Модель даних засобу оптимізації конвергентної системи

Спрощений алгоритм роботи засобу оптимізації конвергентної системи наведено на рис. 6.3.

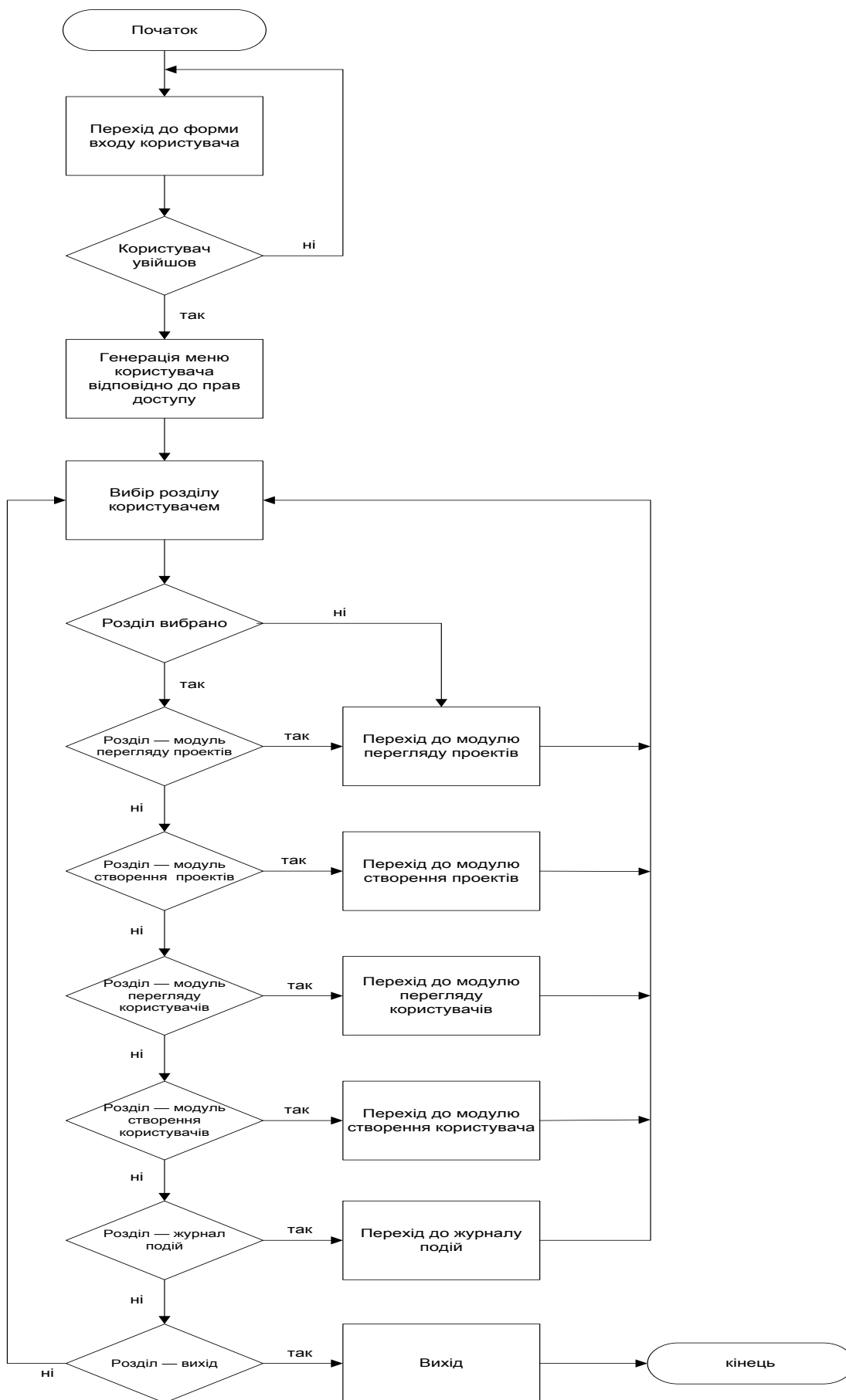


Рисунок 6.3 – Спрощений алгоритм роботи засобу оптимізації
конвергентної системи

Форма введення даних для розрахунку оптимальних наборів компонент наведена на рис. 6.4.

Розрахунки для проекту «Композиція» (id=14)

	Windows NT 4	Windows 2000 Server	Windows XP	Windows 2003 Server	Linux	Unix	0\$ (free)	<100 \$	<1000 \$	<5000 \$	<10000 \$	<100000 \$	Сума	Оцінка
HTTP-Server	1.2	1.5	1	0.7	1	1	1	2	6	7	8	10		
Apache	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	12	0.63
IIS	☒	☒	☒	☒	☐	☐	☐	☐	☐	☒	☒	☒	7	0.37
LDAP-каталог														
ActiveDirectory	☒	☒	☒	☒	☐	☐	☐	☐	☐	☒	☒	☒	7	0.37
OpenLDAP	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	12	0.63
Java Server														
WebSphere	☒	☒	☒	☒	☒	☒	☐	☐	☐	☒	☒	☒	9	0.33
WebSphere Community Edition	☐	☐	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	10	0.37
Oracle Active Server	☒	☒	☒	☒	☒	☒	☐	☐	☐	☐	☒	☒	8	0.30
Data Base														
Oracle	☒	☒	☒	☒	☒	☒	☐	☐	☐	☐	☒	☒	8	0.30
Microsoft SQL Server	☒	☒	☒	☒	☐	☐	☐	☐	☐	☐	☒	☒	6	0.22
IBM DB2	☒	☒	☒	☒	☐	☐	☐	☐	☐	☒	☒	☒	7	0.26
Informix	☒	☒	☒	☒	☐	☐	☐	☐	☐	☐	☒	☒	6	0.22
Web Browser														
Internet Explorer	☒	☒	☒	☒	☐	☐	☒	☒	☒	☒	☒	☒	10	0.24
Mozilla / Firefox	☒	☒	☒	☒	☒	☐	☒	☒	☒	☒	☒	☒	11	0.26
Opera	☒	☒	☒	☒	☒	☐	☒	☒	☒	☒	☒	☒	11	0.26
Netscape Navigator	☒	☒	☒	☒	☒	☐	☐	☒	☒	☒	☒	☒	10	0.24

Рисунок 6.4 – Форма введення даних для розрахунку оптимальних наборів компонент

Приклади реалізації компонентів програмного застосунку для підтримки проектів конвергенції комп'ютерних систем наведено у додатках В, Д.

6.2 Архітектурні та інформаційні моделі ситуаційної діяльності в ССУ

Однією з форм реалізації систем ситуаційного управління є ситуаційні центри (СЦ). Узгоджене прийняття та впровадження рішень у різних сферах та на різних рівнях державного управління потребує створення мережі СЦ державного управління. Мережа СЦ забезпечить єдність та сумісність різноманітних підходів,

моделей, інформаційного забезпечення та великих баз даних, державних реєстрів різних галузевих центрів підготовки, підтримки та прийняття важливих державних рішень. Вона забезпечує їхню стратегічну направленість, комплексність і одночасно надає можливості для оперативного коригування діяльності, зокрема, завантаженості вузлів [4]. Створення мережі СЦ для швидкого реагування на зміни ситуації в різних сферах та на різних рівнях державного й військового управління вимагає створення інформаційно-організаційних сервісів для підтримки діяльності таких СЦ. Сімейства подібних сервісів розробляються з урахуванням досвіду побудови організаційних систем. Цей досвід узагальнено у відповідних стандартах та в теорії управління організаційними системами з урахуванням класифікацій систем у контексті використання СЦ. Вибір та використання моделей організаційних систем та їх структур базується на моделях знань відповідної предметної області.

Формування управлінських рішень, які приймаються шляхом колективного обговорення і проводиться у формі нарад, є одним із ключових етапів ситуаційного управління в СЦ. Одним із основних етапів ситуаційного управління є формування управлінських рішень, які приймаються індивідуально або шляхом колективного обговорення та прийняття рішень відповідно до правил, затверджених в організації. Колективне обговорення питань ситуаційного управління здійснюється шляхом проведення наради. Тому актуальною задачею є створення програмних сервісів підтримки процедур прийняття рішень у СЦ шляхом проведення нарад.

СЦ – це автоматизована система ситуаційного управління, яка об'єднує новітні інформаційні технології, людські можливості та теорію прийняття рішень, а також дозволяє приймати колективні рішення, аналізувати та прогнозувати їх результати. СЦ функціонує як організаційно-технічний комплекс, основою якого є інформаційні технології, що підтримують управлінські рішення на основі комплексного моніторингу факторів впливу на розвиток процесів, що відбуваються в об'єкті управління. СЦ є важливим компонентом забезпечення адекватного оперативного управління складними системами різного масштабу та призначення. Різноманіття проблемних областей (доменів) діяльності та завдань, пов'язаних із

узгодженим СУ цих доменів, вимагає побудови мережі ситуаційних центрів. Складовими елементами СЦ як організаційно-технічних комплексів є архітектура, технологія, методи організації та управління. На основі аналізу архітектурних рішень для системи ситуаційного управління у роботі [303] була запропонована сервіс-орієнтована архітектура, яка представлена на рис. 6.2. Сучасний підхід до створення ССУ, зокрема СЦ, базується на композиції відповідних сервісів.

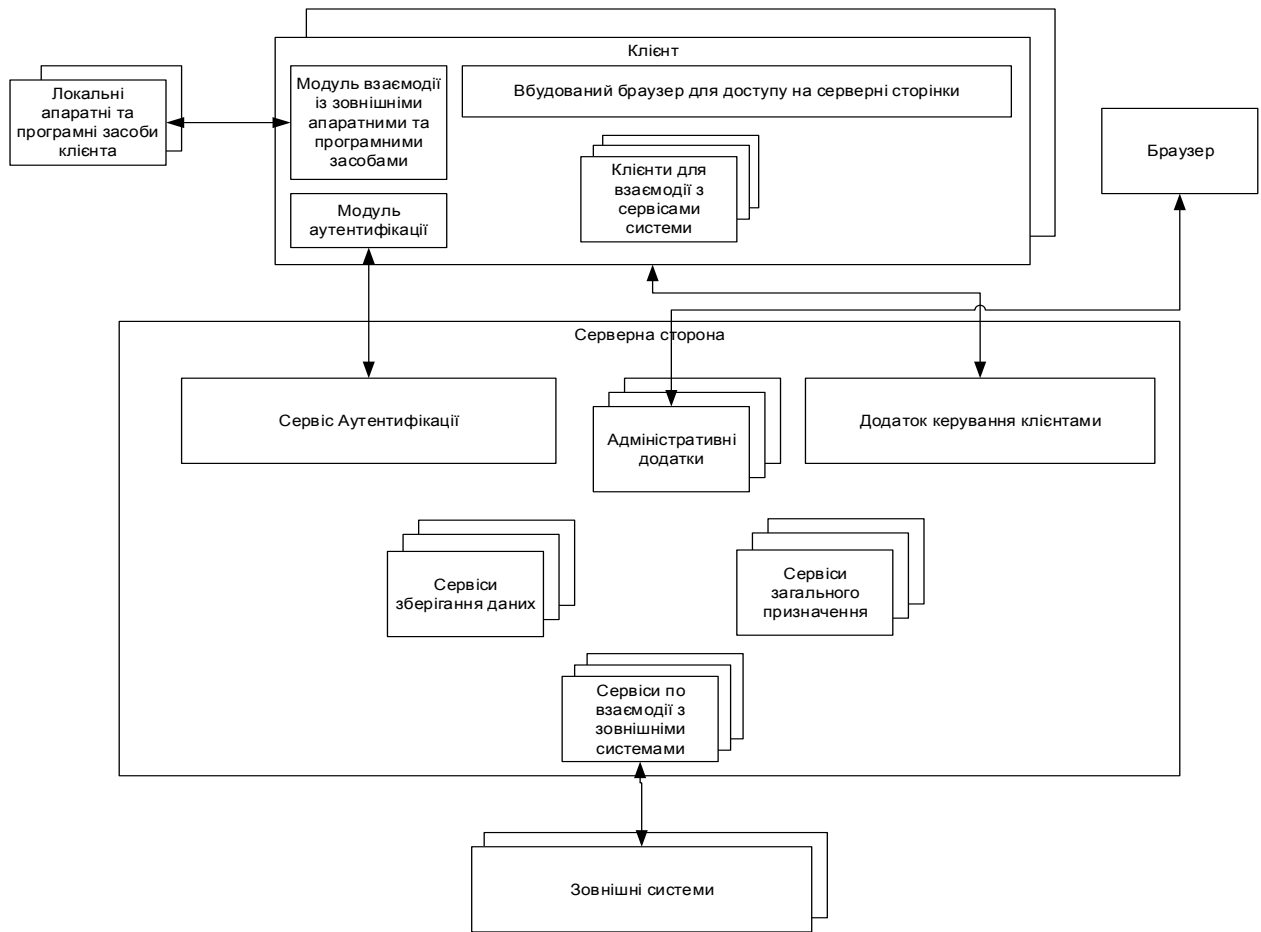


Рисунок 6.5 – Сервіс-орієнтована архітектура системи ситуаційного управління

Автоматичне документування процесу проведення наради здійснюється з використанням сервісу підтримки проведення нарад. Організаційно-інформаційний сервіс для системи підтримки прийняття рішень в процесі ситуаційного управління шляхом проведення нарад (ІС «Meeting»), який розроблено в ІПММС НАН України, забезпечує інформаційну підтримку прийняття рішень у СЦ та передачу цих рішень до організаційної системи управління для подальшої підтримки їх реалізації та

контролю. Метою створення та використання сервісу ІС «Meeting» є підвищення ефективності прийняття рішень шляхом проведення нарад на різних етапах ситуаційного управління на базі застосування прогресивних та сучасних організаційних, інформаційних, інформаційно-аналітичних підходів, моделювання та технічних засобів у СЦ. Сервіс ІС «Meeting» забезпечує налаштування, обслуговування та проведення нарад у СЦ.

Життєвий цикл ситуаційного управління як цілеспрямованої діяльності реалізується через послідовність таких процесів: усвідомлення ситуації → оцінка ситуації → визначення цілей СУ → розробка та відбір плану для досягнення цілей СУ → організація процедур прийняття рішень → виконання плану СУ → моніторинг виконання плану СУ → оцінка та збереження результатів виконання СУ [13], [28], [36]. Прийняття рішень при реалізації ситуаційного управління можна описати моделями, які базуються на проектному підході та процесах його реалізації. Важливим елементом цього підходу є управління життєвим циклом проекту. Аналіз сучасного досвіду формалізації процесів проектної діяльності свідчить, що у світі накопичено великий обсяг теоретичних знань та практики, розроблено низку стандартів щодо вирішення завдань проектного менеджменту. Знання, практика та досвід управління проектами різних типів підсумовуються і поширюються всесвітньою неприбутковою організацією професійного управління проектами – Інститутом управління проектами (PMI) у формі публікації під назвою «Керування знаннями у сфері управління проектами» (PMBOK) [253]. Посібник PMBOK також містить визначення основних концепцій управління проектами та їх змісту. Еволюція посібника PMBOK, або просто PMBOK, відображається у її виданнях. Останнє, шосте видання PMBOK вийшло в 2017 році. Досвід ефективного вирішення проблем різного характеру для отримання нової якості продукції або послуг у процесі проектної діяльності зосереджений у збірці знань PMBOK, пов'язаний із стандартами та практикою PMBOK. Питання управління життєвим циклом проектів розглядаються у стандарті ISO/IEC/IEEE 16326:2009 – Системи та програмне забезпечення – Процеси життєвого циклу – Управління проектами [305]. Крім того, стандарт ISO 21500:2012 – Керівництво по управлінню проектами [306] є

першим стандартом ряду стандартів управління проектами і встановлює принципи та процедури, в яких представлені найкращі практики управління проектами з урахуванням вимог РМВОК та стандартів управління якістю.

Практично будь-який вид діяльності, пов'язаний із ситуаційним управлінням має на меті підтримку параметрів системи в певних межах або приведення цих параметрів до норми і має характер проєктної діяльності. Посібник РМВОК дає таке визначення проєкту: «Проєкт є тимчасовою діяльністю, метою якої є створення унікального продукту, послуги або результату. Тимчасовий характер проєкту вказує на певний час початку і закінчення. Закінчення настає тоді, коли цілі проєкту досягнуті або коли проєкт припиняється через те, що його цілі не будуть або не можуть бути досягнуті, або ж коли у проєкті більше немає потреби» [304]. Таким чином, проєктний підхід із певними обмеженнями та умовами може застосовуватися до ситуаційного управління. Аналіз показує, що структура ситуаційного управління та проєктної діяльності є подібними. По суті, ситуаційне управління є специфічним видом проєктної діяльності. Модель проєктної діяльності дає конкретні рекомендації щодо організації процесів ситуаційного управління на основних етапах планування, реалізації, моніторингу, управління та контролю. Незважаючи на багато спільних рис у проєктній діяльності та ситуаційному управлінні, основні відмінності між ними знаходяться на початковій та заключній стадіях цих процесів. Так, ініціація проєкту пов'язана з мотивацією, яка базується на отриманні певної нової якості в результаті проєкту. Початок процесів ситуаційного управління пов'язаний, насамперед, з реалізацією ситуації, яка потребує певної реакції. Заключний етап діяльності проєкту передбачає отримання нового продукту або послуги. Завершення процесу ситуаційного управління полягає у досягненні цільового стану керованої системи. Така послідовність процесів життєвого циклу та модель СУ, що базується на проєктному підході, реалізується у сервісі ІТ «Meeting».

6.2.1. Моделі організації роботи груп у ССУ

Методологія ситуаційного управління в режимі колективного прийняття рішень іноді вимагає проведення додаткових консультацій між зацікавленими особами, особливо в ході проведення важливої наради, з метою обговорення додаткових аргументів та врахування нових обставин для обґрунтування альтернативних рішень. Тому, для кращої обізнаності, зацікавленим особам та експертам бажано бути ознайомленими із пропонованими варіантами рішень і довідковими матеріалами (протоколами, аудіо- та відеофайлами тощо). Таким чином, учасники наради можуть заздалегідь прослухати доповіді, переглянути ілюстративні матеріали, включити пояснення до них із використанням режиму відтворення. Матеріали звіту можуть мати різний рівень деталізації – від стислих анотацій та змісту майбутнього звіту до його детального викладу. Попередня підготовка доповіді й матеріалів до неї може бути корисною також самим доповідачам для узгодження із прийнятими процедурами та правилами наради, підготовки записів відтермінованих доповідей, які можуть бути використані за відсутністю доповідача на нараді.

Існують різні групи учасників з особливими ролями (агенти), що беруть участь у процесі підготовки та прийняття рішень у процесі ситуаційного управління. Кожен учасник наради (агент), за участю якого приймається рішення, пов'язаний із відповідним актором на діаграмі прецедентів UML. Модель варіантів використання сервісу IC «Meeting» підтримки проведення нарад у СЦ наведено на рис. 6.3.

Сервіс IC «Meeting» для забезпечення та обслуговування проведення нарад у СЦ базується на стандартних процедурах СУ. Завершальний документ під назвою «Протокол наради» формується наприкінці наради. Цей протокол є джерелом управлінських рішень, які приймаються в СЦ, і відображає не тільки прийняті рішення, але також висвітлює хід наради. Організаційно-інформаційна технологія IC «Meeting» забезпечує прийняття рішень щодо СУ шляхом проведення нарад і підтримує автоматичне формування порядку денного наради, його правила, проєкт

рішення наради. Основні характеристики наради та протоколів створюються на робочій станції адміністратора СЦ в автоматичному режимі.

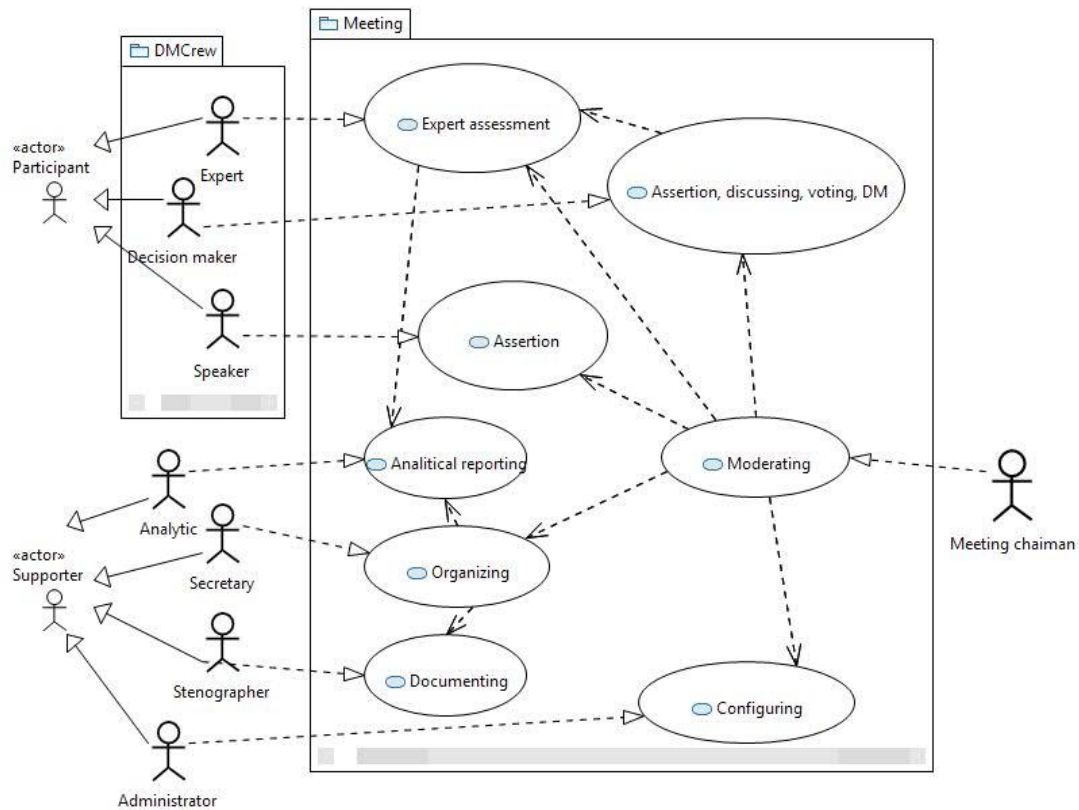


Рисунок 6.6 – Модель варіантів використання сервісу підтримки проведення нарад

Протокол, сформований у результаті наради, має стандартний формат у відповідності з затвердженими правилами та регламентами. Оскільки наради можуть бути дуже різноманітними з точки зору кількості та важливості обговорюваних питань, а також з урахуванням різноманітних регуляторних процедур, основою остаточного документа «Протокол наради» є його загальна метамодель, яка враховує основні нормативні вимоги щодо оформлення таких документів. У метамоделі враховується вирішення таких основних завдань щодо проведення наради [44]:

- 1) формулювання мотивації на основі фактичних ситуацій та тем для наради;
- 2) вибір конфігурації технологічних засобів для проведення наради у СЦ;
- 3) забезпечення підготовки наради:
 - а) формування порядку денного наради;

б) складання списку учасників із визначенням їх ролей;

в) отримання та підготовка матеріалів для звітів, експертних висновків, окремих думок та рекомендацій щодо пунктів порядку денного наради;

4) реєстрація учасників, які прибули на нараду.

Приймаючи до уваги такі складові технології СУ, як метод планування, місце і форму проведення, рівень забезпечення технологічним обладнанням, регламент підведення підсумків, візуалізацію результатів, локалізацію та масштаб нарад, проведено класифікацію нарад. У сервісі ІС «Meeting» прийнята така класифікація нарад за різними класифікаційними ознаками.

За способом планування:

- 1) оперативний (брифінг, спеціальний);
- 2) з попереднім плануванням порядку денного та правил;
- 3) змішаний.

За рівнем вибору обладнання та приміщень для проведення нарад:

1) пристосовані (адаптовані), наприклад, СЦ, парламентські зали, конференц-зали тощо;

2) непристосовані.

За локалізацією проведення:

- 1) локалізовані (сконцентровані);
- 2) розподілені (віддалені).

За підбором учасників наради:

- 1) громадські;
- 2) корпоративні;
- 3) тематичні (актуальні теми) та ін.

Життєвий цикл наради включає такі етапи (процеси) діяльності:

- 1) підготовка;
- 2) проведення;
- 3) формування протоколів;
- 4) збереження/архівування матеріалів наради;
- 5) обробка матеріалів наради.

Етап підготовки до наради включає такі операції (дії):

- 1) автоматична конфігурація технічних засобів для наради;
- 2) автоматична підготовка баз даних;
- 3) автоматизована генерація/включення ролей учасників;
- 4) підготовка матеріалів.

Запропонований сервіс підтримки проведення нарад у процесі ситуаційного управління передбачає автоматичну підготовку і формування іманентних баз даних. Автоматична підготовка баз даних здійснюється відповідно до структурної моделі наради. Структурна модель наради може бути представлена в різних нотаціях, щоб задовольнити вимоги платформ для реалізації сервісу ІТ «Meeting» та легке розуміння розробниками.

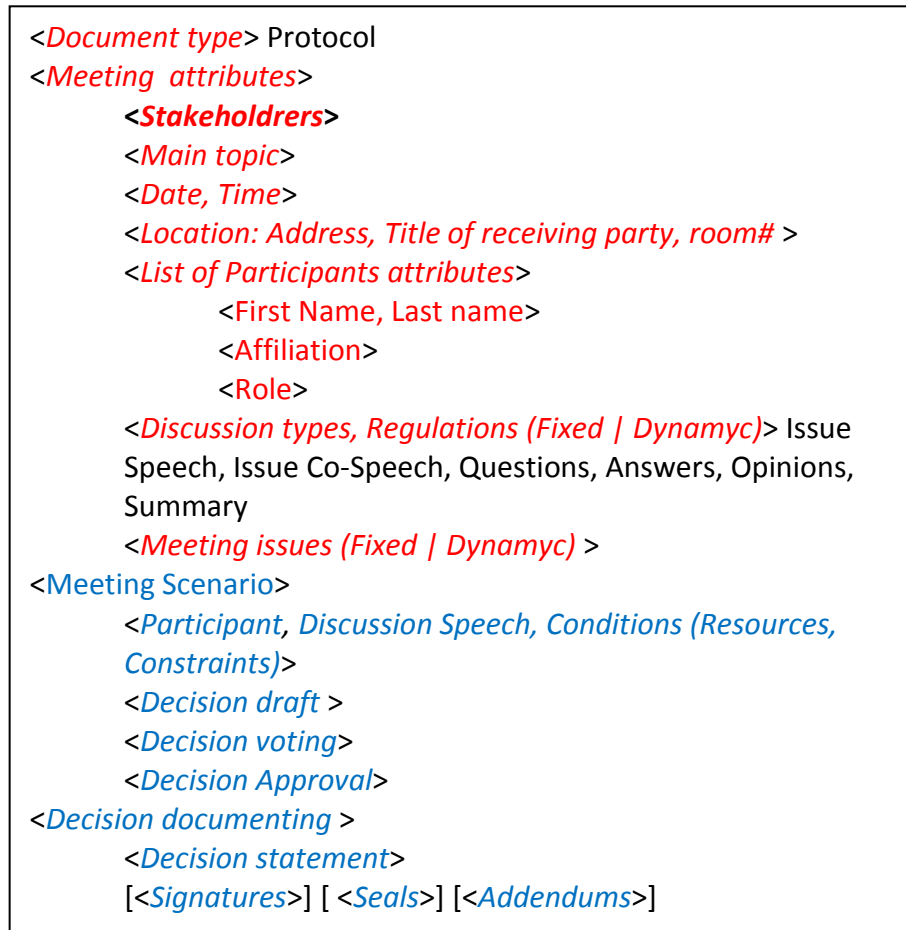


Рисунок 6.7 – Квaziформальна структурна метамодель наради

Запропонована квазіформальна структурна метамодель наради в XML-подібному форматі представлена на рис. 6.7. Цей формат використовується для більш компактного представлення семантичного ядра структурної моделі наради порівняно з використанням строгого XML-синтаксису.

На основі запропонованої квазіформальної структурної метамоделі можна просто створити строгі мовні конструкції для реалізації в сервісі проведення нарад об'єктів нарад відповідно до вимог конкретної платформи реалізації. Таким чином, запропонована модель може бути використана як шаблон для проєктування конкретних програмних об'єктів у бажаних нотаціях, зокрема, BPEL, BPEL4WS, JSON тощо.

Описані вище моделі підтримки роботи груп експертів реалізовані у програмно-технічному моделюючому комплексі «Ситуаційний центр» Інституту проблем математичних машин і систем НАН України.

6.2.2 Методи та моделі колективного прийняття рішень в агентних ССУ

Розробка методів підтримки прийняття рішень експертним шляхом є одним із важливих напрямів досліджень у сфері інформаційних технологій. Експертні оцінки можна умовно поділити на ординальні і кардинальні (рис. 6.8). Особливе місце в експертному оцінюванні та підтримці прийняття рішень займають ординальні оцінки або ранжування альтернатив. На відміну від кардинальних (чисельних) оцінок альтернатив, ординальні (рангові або порядкові) оцінки не несуть інформації про кількісне співвідношення альтернатив, а дозволяють лише впорядкувати альтернативи за заданим критерієм. Потреба в ординальному оцінюванні виникає у випадках, коли експертам важко або неможливо визначити точні абсолютні або відносні чисельні значення характеристик об'єктів (альтернатив). У таких ситуаціях експертам пропонується побудувати ранжування альтернатив, тобто розташувати їх у порядку зростання або спадання міри виразності заданої характеристики

(параметра, критерію, ознаки). Тобто, якщо експерту не вдається кількісно оцінити альтернативи, він може спробувати їх проранжувати. На практиці ординальне оцінювання застосовується при відборі кандидатів на вакантні посади, складанні виборчих списків, оцінці результатів тестів, спортивних змагань, конкурсів, у порівнянні проєктів, що конкурують між собою, тощо.



Рисунок 6.8 – Умовна класифікація експертних оцінок об'єктів/альтернатив

Умовно задачі, що виникають в ході підтримки прийняття рішень із використанням експертних даних, можна розділити на два класи (рис. 6.9) [293]. 1-й клас: задачі агрегації індивідуальних і однокритерійних експертних оцінок альтернатив із метою побудови їх рейтингу або ранжування за спільним критерієм ефективності. 2-й клас (обернена задача): визначення невідомих коефіцієнтів відносної вагомості критеріїв оцінки альтернатив на основі попереднього досвіду їх багатокритеріальної експертної оцінки.



Рисунок 6.9 – Умовна класифікація задач та методів підтримки прийняття рішень

Означення 1.1. Під ординальною оцінкою (рангом) конкретної альтернативи (об'єкта) за певною ознакою (критерієм) ми розумітимемо її порядковий номер у заданій множині альтернатив, розташованих у порядку зростання або спадання міри виразності даної ознаки.

Ординальні оцінки альтернатив можуть бути отримані експертним шляхом або на основі попередньо отриманих кардинальних оцінок. Якщо кардинальну оцінку отримати важко або неможливо, але альтернативи все-таки треба оцінити, то можна використовувати методи парних порівнянь об'єктів [63], [82]. Подібна ситуація часто виникає, коли ознаки, за якими оцінюються альтернативи, важко піддаються кількісному представленню. Саме у таких випадках, здебільшого, і застосовуються експертні методи підтримки прийняття рішень.

Слід підкреслити, що, на відміну від кардинальної оцінки об'єкта, його ординальна оцінка не несе інформації про чисельне значення міри виразності ознаки, за якою йде ранжування об'єктів у множині, а також про те, наскільки один об'єкт відрізняється від іншого за даною ознакою.

Під час парних порівнянь об'єктів, які використовуються у багатьох методах прийняття рішень, для отримання значень переваг експертам ставляться два запитання. Перше, «ординальне», дозволяє з'ясувати, яка із двох альтернатив, на думку експерта, є кращою за даним критерієм. Друге, «кардинальне», дозволяє дізнатися, наскільки саме одна альтернатива переважає іншу за мірою виразності заданого критерію. Для отримання ординальних експертних оцінок достатньо поставити експерту лише перше із двох вказаних запитань. Це означає, що, з одного боку, ранжування альтернатив несе меншу інформацію, ніж їхні кардинальні оцінки, а з іншого, на проведення експертизи, де використовуються ординальні оцінки, витрачається значно менше часу, ніж на експертизу, яка вимагає побудови матриці парних порівнянь в інших шкалах [293]. До того ж, часто експерту важко оцінити міру переваги одної альтернативи із множини над іншою. (Наприклад, для порівняння виступів спортсменів чи музикантів, що беруть участь у конкурсі, членам журі найдоцільніше використовувати ординальну шкалу).

Через це використання ординальних (рангових) оцінок є досить актуальним аспектом, якому належить особливе місце у сфері підтримки прийняття рішень.

Найчастіше для отримання експертних оцінок альтернатив, що належать до певної множини, використовуються два очевидні методи — виключення та розширення. Наведемо стислий опис кожного із вказаних методів.

При отриманні ранжування множини з m альтернатив $\{a_i, i=1..m\}$ методом виключення експерту спочатку пропонується обрати найкращу, на його думку, альтернативу із множини. Після цього дана альтернатива виключається із множини і експерту пропонується визначити найкращу альтернативу з тих, що залишилися. На $(i+1)$ -му кроці експерт має обирати найкращу з $(m-1)$ альтернатив. Таким чином, за $(m-1)$ кроків буде отримане ранжування усіх альтернатив.

Метод розширення зводиться до такої процедури. На першому кроці обирається довільна пара альтернатив (a_h, a_k) із множини і експерту пропонується визначити, яка з альтернатив у даній парі домінує над іншою. Припустимо, $a_h \succ a_k$. На наступному кроці з множини обирається довільний об'єкт a_s . Експерту пропонується визначити, яке із співвідношень $a_s \succ a_h \succ a_k$, $a_h \succ a_s \succ a_k$, $a_h \succ a_k \succ a_s$ справедливе. Для цього йому слід по черзі попарно порівняти три альтернативи. На i -му кроці алгоритму експерту буде запропонована множина з $(i+1)$ альтернативи і він муситиме здійснити i порівнянь. Загальна кількість порівнянь, які доводиться робити експерту для ранжування усієї множини альтернатив, є сумою арифметичної прогресії з $(m-1)$ члена. Отже, загалом експерту належить виконати $\frac{m(m-1)}{2}$ парних порівнянь.

Проблема отримання результуючого ранжування множини об'єктів (альтернатив) на основі їхніх ранжувань за кількома критеріями, які побудовані одним або кількома експертами, залишається актуальною на протязі кількох століть.

Першими методами побудови групового ранжування множини альтернатив були методи Борда та Кондорсе. Методи призначалися для побудови ранжування кандидатів кількома виборцями.

Визначальними факторами для результативного голосування є:

- виборці – вибір множини виборців;
- процедура – організація процедури голосування;
- обробка результатів (процесінг) – система підрахунку голосів;
- інтерпретація – спосіб вибору переможця або переможців.

У залежності від цілей і умов вибору варіантів застосовуються різні системи виборів:

1. Методи голосування з одним переможцем:

- 1.1. Проста більшість.
- 1.2. Двотурове голосування.
- 1.3. Виключне голосування.
- 1.4. Преференційні системи.

1.4.1. За критерієм Кондорсе:

- метод Кондорсе;
- метод Копленда;
- метод Кемені-Янга;
- метод мінімакса;
- метод Нансона;
- метод ранжованих пар;
- метод Шульце.

1.4.2. Голосування Букліна.

1.4.3. Метод Кумбса.

1.4.4. Пряме багатотурове голосування.

1.4.5. Підрахунок Борда.

1.5. Неранжовані методи.

1.5.1. Схвальне голосування.

1.5.2. Розподільне голосування.

2. Методи голосування з багатьма переможцями.

2.1. Пропорційна система.

2.1.1. Сукупне (накопичувальне, кумулятивне) голосування.

2.1.2. Змішаних членів.

2.1.3. Партийні списки (відкриті-закриті):

- метод Д'Хондта;
- найвищих переваг;
- найбільшого залишку;
- метод Сент-Лагу.

2.1.4. З одним переміщуванням голосом.

2.1.5. Співставлення підсумкових пар з одним переміщуванням голосом.

2.1.6. Метод Шульце з одним переміщуванням голосом.

2.1.7. Квотована система Борда.

2.1.8. Матричний вибір.

2.2. Напівпропорційна система.

2.2.1. Система з додатковим членом.

2.2.2. Сукупне (накопичувальне, кумулятивне) голосування.

2.2.3. З одним непереміщуваним голосом.

2.3. Непропорційна система.

2.3.1. Переважна більшість.

2.3.2. Преференційне блокове голосування.

2.3.3. Обмежене голосування.

3. Випадковий вибір.

3.1. Жеребкування (лотерея).

3.2. Випадкове голосування.

Даний перелік не є повним, але дає можливість оцінити різноманітність способів здійснення колективного вибору в різних умовах.

Результати порівняння наведені у табл. 6.1. [63]

Таким чином, система організації голосування СЦ повинна містити такі елементи:

- множину моделей голосуючих (виборців);
- множину моделей для різних методів голосування;
- підсистему попередньої оцінки варіантів рішень;
- підсистему управління процесом голосування у відповідності з обраною системою;
- підсистему управління профілями голосування;
- підсистему протоколювання процедур голосування.

Таблиця 6.1 – Відповідність критеріїв голосування деяким виборчим системам з одним переможцем

Системи голосування	Критерії							
	Більшість	Монотонність	Узгодженість і партнерство	Переважа за Кондорсе	Програш за Кондорсе	Незалежність від несуттєвих альтернатив	Автономність груп	Зворотна симетрія
Схвальне голосування	н/в	Так	Так	Ні	Ні	Так	н/в	Так
Підрахунок Борда	Ні	Так	Так	Ні	Так	Ні	Ні	Так
Прямі багатотурові вибори	Так	Ні	Ні	Ні	Так	Ні	Так	Ні
Кемені-Янг	Так	Так	Ні	Так	Так	Ні	Ні	н/в
Мінімаксний	Так	Так	Ні	Так	Ні	Ні	Ні	Ні
Проста більшість	Так	Так	Так	Ні	Ні	Ні	Ні	н/в
Розподілена	Ні	Так	Так	Ні	Ні	Так	Так	Так
Ранжовані пари	Так	Так	Ні	Так	Так	Ні	Так	н/в
Багатотурове голосування	Так	Ні	Ні	Ні	Так	Ні	Ні	н/в
Шульце	Так	Так	Ні	Так	Так	Ні	Так	Так

н/в – не визначено.

Обмеженість застосування різних виборчих систем визначається їх відповідністю тим чи іншим критеріям голосування, зокрема:

- критерію більшості – якщо має місце більшість, яка ставить одного кандидата вище інших кандидатів, то цей кандидат завжди виграє;
- критерію монотонності – кандидат-переможець не може програти, якщо його рейтинг вищий, і кандидат-переможений не може виграти, якщо його рейтинг є нижчим;

- критерію узгодженості (поповнення) – якщо виборці поділяються на дві групи і обидві групи віддають перевагу одному кандидату, то він є переможцем у загальних виборах;
- критерію участі – вважається, що всі виборці приймають участь у виборах;
- критерію Кондорсе – кандидат, який перемагає всіх інших кандидатів у попарних порівняннях, є переможцем;
- критерію переможеного за Кондорсе – кандидат, який програє всім іншим кандидатам у попарних порівняннях програє у виборах;
- незалежністю від несуттєвих альтернатив – результат виборів не залежить від додавання або видалення кандидатів, що програють у виборах;
- незалежністю від кандидатів клонів – результат виборів є незмінним при додаванні кандидатів, ідентичних до існуючих;
- зворотною симетрією (нейтральністю) – симетрична зміна переваг виборців призводить до відповідної зміни результатів виборів.

Під час побудови групового ранжування за методом (правилом) Борда кожен виборець будував власне ранжування кандидатів. Якщо загальна кількість кандидатів дорівнювала m , то кандидату, який отримував найвищий ранг, присвоювалося число $-(m-1)$, наступному за ним $-(m-2)$ і т.д. Найгірший, на думку виборця, кандидат, отримував оцінку 0. Після цього дані заносилися до таблиці, і оцінки кожного кандидата, надані всіма виборцями, складалися разом. Кандидат, сума оцінок якого буде найменшою, отримував перше місце у груповому ранжуванні, а кандидат, сума оцінок якого була найбільшою, – останнє.

Під час побудови групових ранжувань методом Кондорсе кожен виборець також мусить побудувати індивідуальне ранжування кандидатів. Якщо кількість індивідуальних ранжувань, у яких кандидат C_1 випереджає кандидата C_2 , перевищує кількість індивідуальних ранжувань, де перший кандидат поступається другому, то у груповому ранжуванні кандидат C_1 отримує вищий ранг.

Дані методи можна використовувати при побудові групових ранжувань альтернатив за кількома критеріями. У ролі “виборців” можуть виступати експерти (у загальному випадку з різною компетентністю) та критерії оцінки альтернатив.

Пізніше були розроблені численні модифікації вказаних методів, а також інші підходи до побудови групових ранжувань, такі як визначення медіани Кемені, правило Сімпсона та ін.

Чотирьом із п'яти вимог до процедур агрегації індивідуальних переваг, сформульованих Ерроу, відповідає процедура пошуку медіани Кемені множини індивідуальних ранжувань, і в цьому сенсі, вона є найбільш “справедливою” з усіх методів агрегації ординальних оцінок [293].

Для спрощення процедури замість медіани Кемені шукають так зване компромісне ранжування. У той час, як медіана Кемені є ранжуванням, рівновіддаленим від ранжувань, побудованих різними експертами, компромісне ранжування співпадає з ранжуванням, побудованим одним із членів експертної групи, і, як наслідок, не задовольняє вимозі відсутності диктатури, яку висуває Ерроу.

При побудові медіани Кемені враховуються ранжування, задані однаково компетентними експертами. Узагальнення даного методу на випадок різної компетентності експертів та різної вагомості критеріїв (якщо йдеться про багатокритеріальне ранжування) становить окрему актуальну задачу, у той час як методи Кондорсе та Борда вже узагальнені на вказаний випадок [293].

Слід підкреслити, що у загальному випадку результати побудови групових та багатокритеріальних ранжувань, отриманих різними методами, не співпадають.

Показники узгодженості ординальних оцінок. При розробці методів, які ґрунтуватимуться на досвіді багатокритеріального ординального оцінювання альтернатив, доцільно визначити потенційні критерії збіжності цих методів. Оскільки ординальна шкала є цілочисельною та дискретною, а ранги альтернатив не несуть інформації про точні кількісні співвідношення між альтернативами, використовувати для оцінки точності методів показники, що містять кардинальні значення (середню квадратичну похибку, норму різниці між заданим і отриманим вихідними векторами, статистичні показники, тощо) некоректно. У даному пункті пропонується стислий опис кількох показників узгодженості групових, і, відповідно, багатокритеріальних ранжувань.

Коефіцієнт рангової кореляції Кендела може використовуватися як міра узгодженості двох ранжувань.

Нехай задано матриці ординальних парних порівнянь для двох ранжувань множини з m альтернатив.

Елемент матриці ординальних порівнянь d_{ij} визначається таким чином:

$d_{ij} = 1$, якщо $a_i \approx a_j$;

$d_{ij} = 0$, якщо (альтернативи еквівалентні, також можна записувати в матрицю 0, якщо альтернативи неможливо порівняти);

$d_{ij} = -1$, якщо $a_i \prec a_j$.

Коефіцієнт рангової кореляції двох ранжувань R_1 та R_2 обчислюється за формулою

$$K = \frac{2 \sum_{i=1}^m \sum_{j=1}^m d_{ij}^{(1)} d_{ij}^{(2)}}{m(m-1)} . \quad (6.1)$$

Ідеально узгодженим ранжуванням відповідає $K=1$, а максимально неузгодженим – $K=-1$. Коефіцієнт може використовуватися лише у випадку строгих ранжувань. Для визначення ступеня узгодженості ранжувань, кількість яких може бути більшою за 2, використовується коефіцієнт конкордації [293].

Коефіцієнт конкордації множини ранжувань m альтернатив, побудованих n експертами, обчислюється за формулою

$$W = \frac{12S}{n^2(m^3 - m)} . \quad (6.2)$$

де S – сума квадратів відхилень сум рангів альтернатив від середнього значення цих сум. Доведено, що максимальне значення S дорівнює $\frac{n^2(m^3 - m)}{12}$,

причому це значення досягається у випадку повної узгодженості індивідуальних ранжувань. Отже, максимальне можливе значення коефіцієнта конкордації дорівнює 1.

Зазначимо, що як коефіцієнт рангової кореляції, так і коефіцієнт конкордації, передбачають, що експерти, які будують ранжування, є однаково компетентними. За таких умов неможливо визначити достатність міри узгодженості їхніх оцінок. У [246] введені узагальнення двох вищевказаних коефіцієнтів на випадок довільного співвідношення рівнів компетентності експертів, а також запропонований підхід до визначення порогів узгодженості ординальних експертних оцінок.

Ще один показник узгодженості двох ранжувань – «відстань між ранжуваннями» (відстань Кемені) [293].

Якщо є дві матриці ординальних парних порівнянь m альтернатив (див. вище), то відстань між ранжуваннями R_1 та R_2 обчислюватиметься за формулою

$$K = \sum_{i=1}^m \sum_{j=1}^m |d_{ij}^{(1)} - d_{ij}^{(2)}| \quad (6.3)$$

Максимальне значення показника дорівнює $4m(m-1)$. За цією величиною можна нормувати показник, щоб він змінювався у діапазоні від 0 до 1, якщо це необхідно для унаочнення.

Архітектурні моделі ССУ. Реалізація описаних експертних методів прийняття рішень потребує підтримки у проблемно-орієнтованих комп'ютеризованих системах, адаптованих до вирішення відповідних класів задач.

Діаграма прецедентів є композитним графом, що включає множину акторів і прецедентів (варіантів використання), обмежених контурами системи, асоціації між акторами (агентами) та прецедентами, відношення між прецедентами та відношення узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в тому, що проєктована система представляється у вигляді множини сутностей-акторів, які взаємодіють із системою за сценаріями

визначеними варіантами використання. Модель прецедентів використовують для опису сервісів, які надаються система надає агентам (акторам). Іншими словами, кожен варіант використання визначає деякий набір операцій, який виконується системою за запитами агента [307] [308]. При цьому нічого не деталізується те, яким чином буде реалізована взаємодія агентів з системою.

Модель варіантів використання для голосування представлена на рис. 6.10. Дана діаграма показує взаємозв'язки між агентами та ССУ. Актор, тобто агент адміністратор, здійснює керування ССУ та системою голосування через інтерфейс, а актори, типу агентів користувачів, мають обмежений вибір дій, тобто тільки право на голосування та перегляд результату [42].

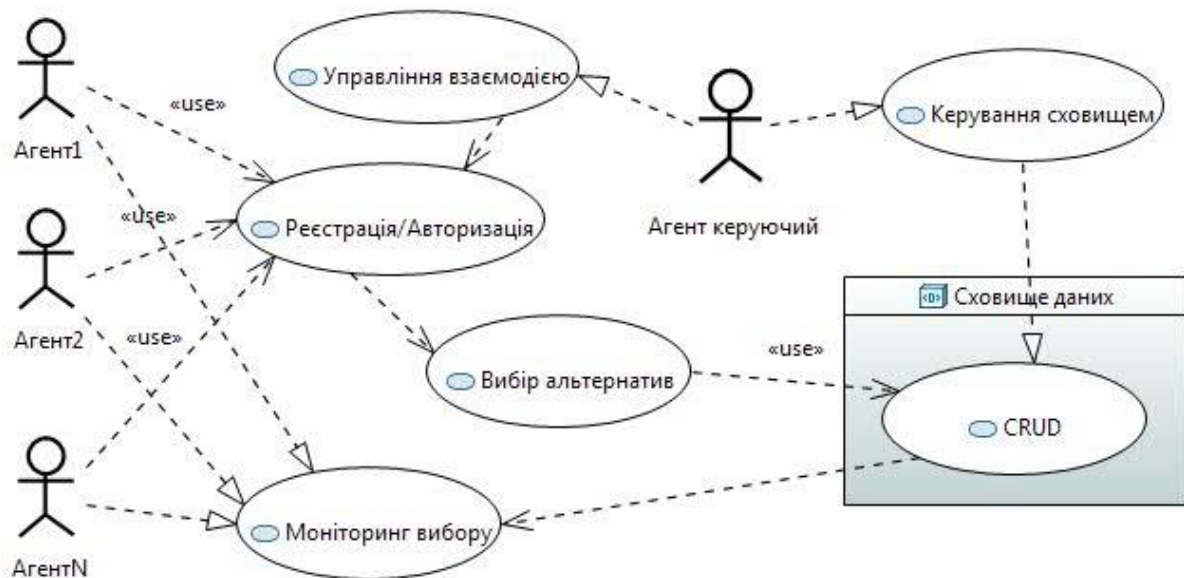


Рисунок 6.10 – Модель варіантів використання вибору з альтернатив

На діаграмі зображено відношення між агентами, прецедентами та сховищем даних. Агент може приймати участь у прийнятті рішень лише за умови попередньої реєстрації.

На основі розробленої діаграми варіантів використання створюються інші технологічні моделі програмного продукту, такі як діаграми діяльності, діаграми послідовностей, діаграми станів, діаграми компонентів і класів, діаграми розгортання.

Моделі динаміки системи голосування. Всі потоки подій одного варіанта використання, сценарії двох і більше взаємопов'язаних варіантів використання або окремий потік подій представляються моделями у вигляді діаграм діяльності.

Діаграма діяльності в UML це візуальне представлення графа діяльностей який є різновидом діаграм поведінки. Діаграма, що використовується при моделюванні діяльностей, представляє декомпозицію деякої складної діяльності, наприклад, скоординованого виконання окремих операцій і вкладених видів діяльності, які з'єднуються дугами, що позначають потоки операцій між вузлами з зазначенням їх виконавців.

Типова модель діяльності «Реєстрація/авторизація», представлена на рис. 6.11, відображає операції, які здійснюються агентами при вході у систему.

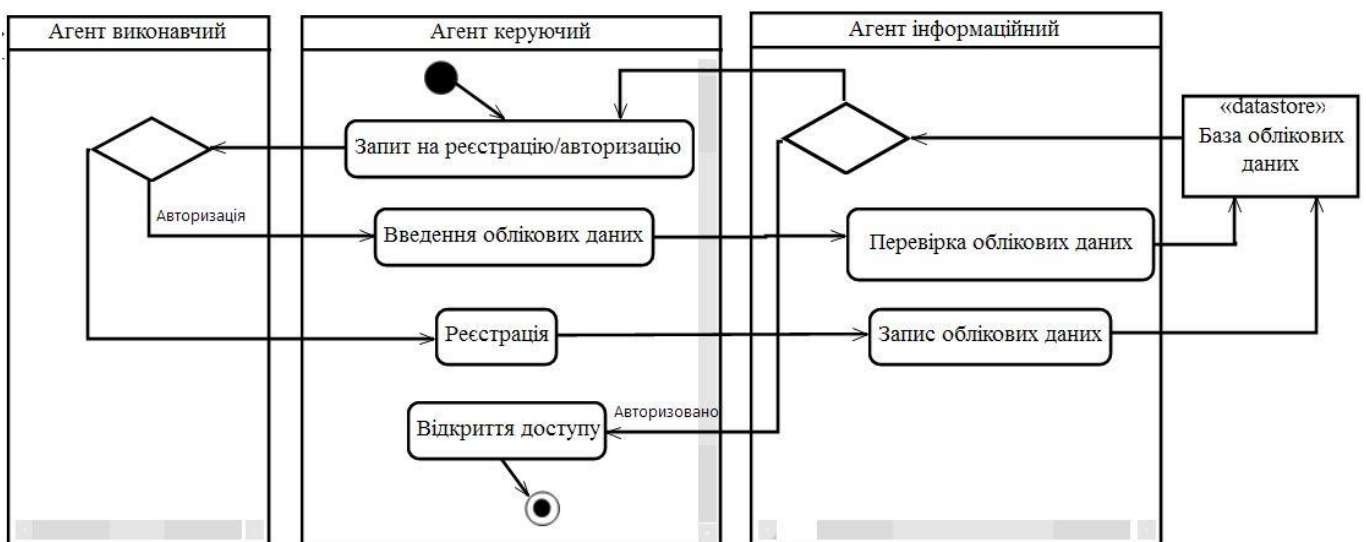


Рисунок 6.11 – Модель діяльності для процесу реєстрації/авторизації

Агенту, при реєстрації, створюється новий обліковий запис, який попередньо перевіряється сервером на можливу колізію введених облікових даних (логіна або пароля). При відсутності колізії агент отримує новий обліковий запис, а в іншому випадку, агент приймає рішення про повторну спробу створення облікового запису чи відмову. Колізія перевіряється за базою облікових даних, в якій зберігаються всі дані про раніше зареєстрованих агентів.

Після реєстрації, в наступних сеансах роботи в ССУ агент авторизується за своїми обліковими даними.

Модель діяльності при виборі альтернатив «Голосування» представлена на рис. 6.12. Ця модель визначає процедуру вибору на основі голосування, яка здійснюється у ССУ.

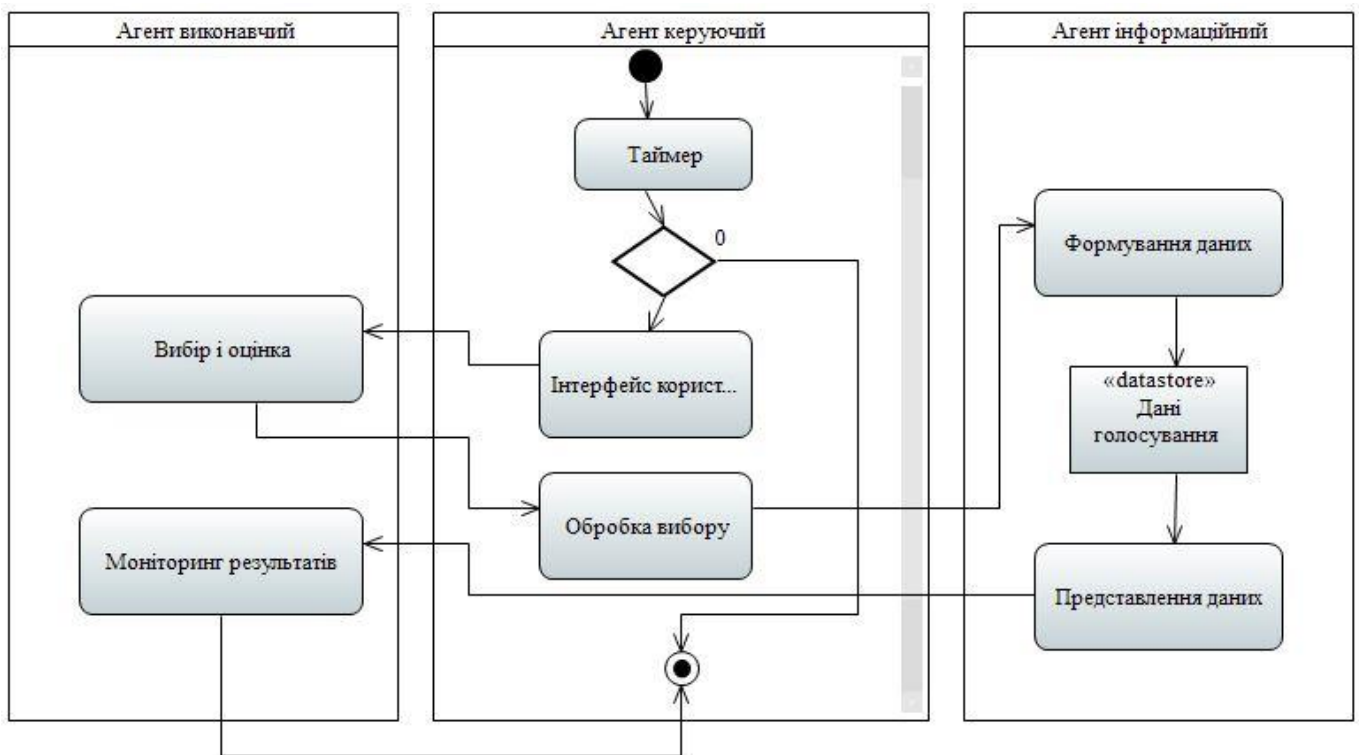


Рисунок 6.12 – Модель діяльності при виборі альтернатив

У даній моделі описується процедура проведення голосування для зареєстрованого агента. Після чого авторизації агент отримує доступ до профілю та можливість приймати участь у голосуванні як виконавчий агент. В разі невідповідності введених облікових даних або обмежень щодо доступу формується відповідне повідомлення про помилку.

Модель діяльності для здійснення вибору шляхом попарних порівнянь (модель Кондорсе, алгоритм Шульце, тощо) представлена на рис. 6.13. За цією моделлю агенти здійснюють вибір на основі ранжування можливих альтернатив, на основі якого формується вибір і оцінка виконавчого агента.

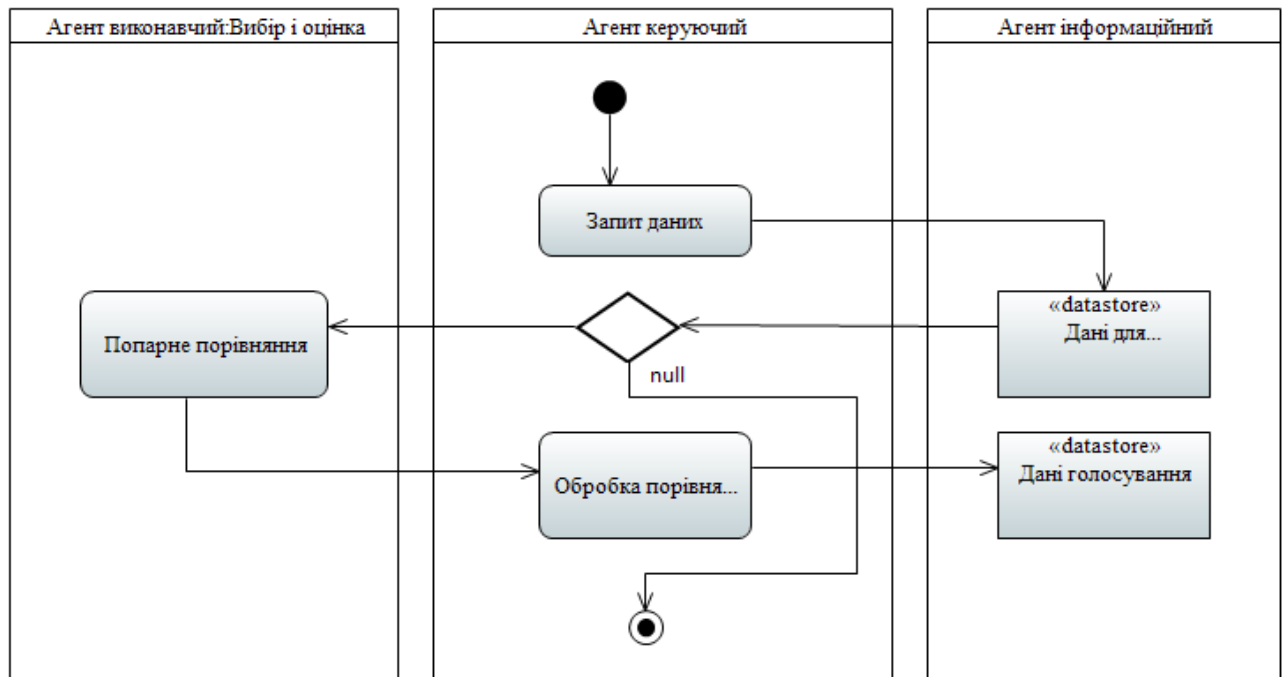


Рисунок 6.13 – Модель діяльності попарних порівнянь

На рис. 6.13 показано, що на основі цих виборок окремих агентів формується загальний результат голосування, який зберігається у сховищі даних ССУ та надається для перегляду авторизованих агентам.

Діаграма послідовностей процесів системи голосування. Для відображення поведінки об'єктів у сценаріях діяльності застосовується модель, як представляється діаграмами послідовностей. Для її побудови кожному з об'єктів сценарію ставиться у відповідність його лінія життя, яка відображає життєві цикли подій між виникненням події та завершенням її обробки. На діаграмі лінія життя позначається вертикальною пунктирною лінією, на верхівці якої у прямокутнику зображується назва об'єкта, якому вона відповідає.

Діаграма послідовності представляє всі об'єкти, які беруть участь у взаємодії. Діаграма послідовності відображає взаємодію об'єктів, впорядковану за часом, зокрема, ця діаграма відображає задіяні об'єкти та послідовність операцій, які вони виконують. На діаграмі послідовності, представлений на рис. 6.14, показано розгортання у часі процедур вибору альтернатив.

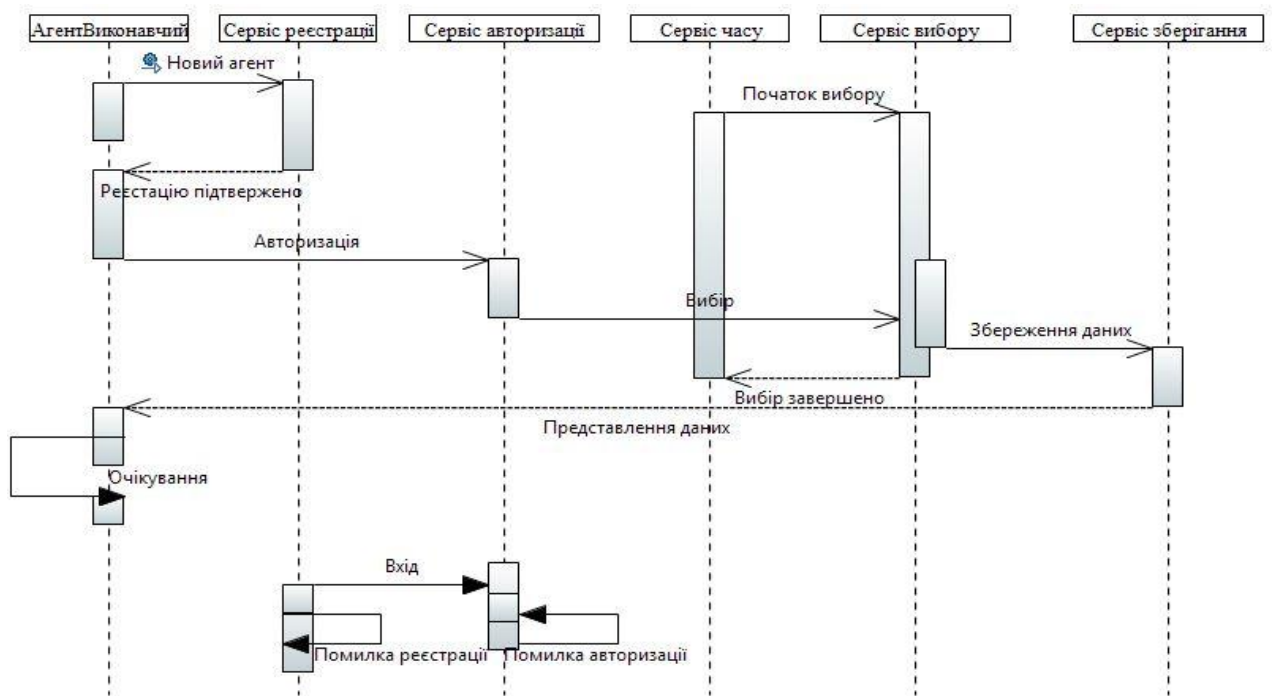


Рисунок 6.14 – Діаграма послідовності при виборі альтернатив

На діаграмі послідовностей (рис. 6.14) показано взаємодію пов'язаних об'єктів, а саме: виконавчого агента і сервісів реєстрації, авторизації, часу, вибору, збереження, представлену відповідними лініями життєвого циклу процесів. Обробка помилок показана на лінії життя сервісів реєстрації та авторизації. Інші помилки також можуть впливати на проведення процедур вибору альтернатив.

Структурні моделі системи голосування. На етапі аналізу вимог і попереднього проектування системи на основі побудованих моделей прецедентів будуються діаграми класів системи.

Діаграма класів є структурним поданням системи і представляє детальну інформацію про структуру компонентної моделі системи в термінах ієрархії класів об'єктів системи, тобто про архітектуру системи. На діаграмі класів відображено внутрішню структуру і типи відносин між окремими об'єктами і підсистемами, що приводить до розвитку концептуальної моделі системи. Діаграма класів служить для подання структурної моделі системи в термінології ієрархії класів об'єктно-орієнтованого представлення системи. Діаграма класів для реалізації процедури вибору альтернатив представлена на рис. 6.15.

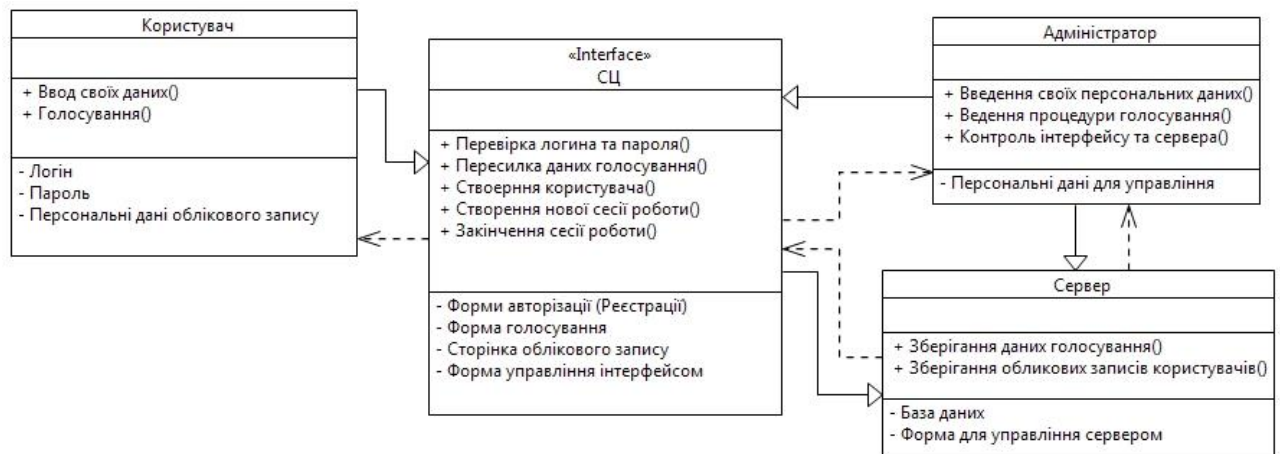


Рисунок 6.15 – Діаграма класів для реалізації процедури вибору альтернатив

Діаграма класів може відбивати, зокрема, різні взаємозв'язки між окремими сутностями предметної області, такими як об'єкти й підсистеми, а також описує їхню внутрішню структуру й типи відносин. На даній діаграмі не вказується інформація про тимчасові аспекти функціонування системи. Із цього погляду діаграма класів є подальшим розвитком концептуальної моделі проєктованої системи.

На діаграмі зображено класи такі, як Користувач, Адміністратор, Інтерфейс, Сервер. Вони мають свої атрибути та функції.

Клас Користувач має атрибути логін, пароль та персональні дані, а функціями є ввід даних у форму реєстрації та авторизації, а також можливість огляду інформації та голосування, яка надається інтерфейсом.

Клас Адміністратор має своїми атрибутами унікальний логін, який створив адміністратор, та його особистий пароль, виконує функції редагування даних на сервері та інтерфейсі, а також огляд всієї інформації, що міститься на сервері та сайті.

Клас Інтерфейс включає функції представлення головної сторінки сайту, виводу нової форми для занесення даних (логіна та пароля), різного типу повідомлення (помилки, повідомлення щодо успіху, попереджувальні та інформаційні повідомлення), а також виводу на екран таймера голосування.

Клас Сервер виконує функції зберігання даних про голосування, передачі сайту даних про результат голосування та можливі повідомлення про помилки в базі даних.

За допомогою діаграми розгортання моделюють фізичний розподіл програмних, інформаційних, апаратних компонентів системи в межах доступної інфраструктури. Особлива увага на діаграмі розгортання приділяється відображенню того, які використовуються апаратні компоненти («вузли» – сервери баз даних і додатків, web-сервер), які програмні компоненти («артефакти» – бази даних, web-додатки) працюють на кожному з них і як частини комплексу з'єднані один з одним. Таким чином, у разі проектування складної ІС її необхідно розділити на частини і досліджувати кожну частину окремо.

Діаграма розгортання, представлена на рис. 6.16, відображає розміщення та взаємодію між компонентами системи.

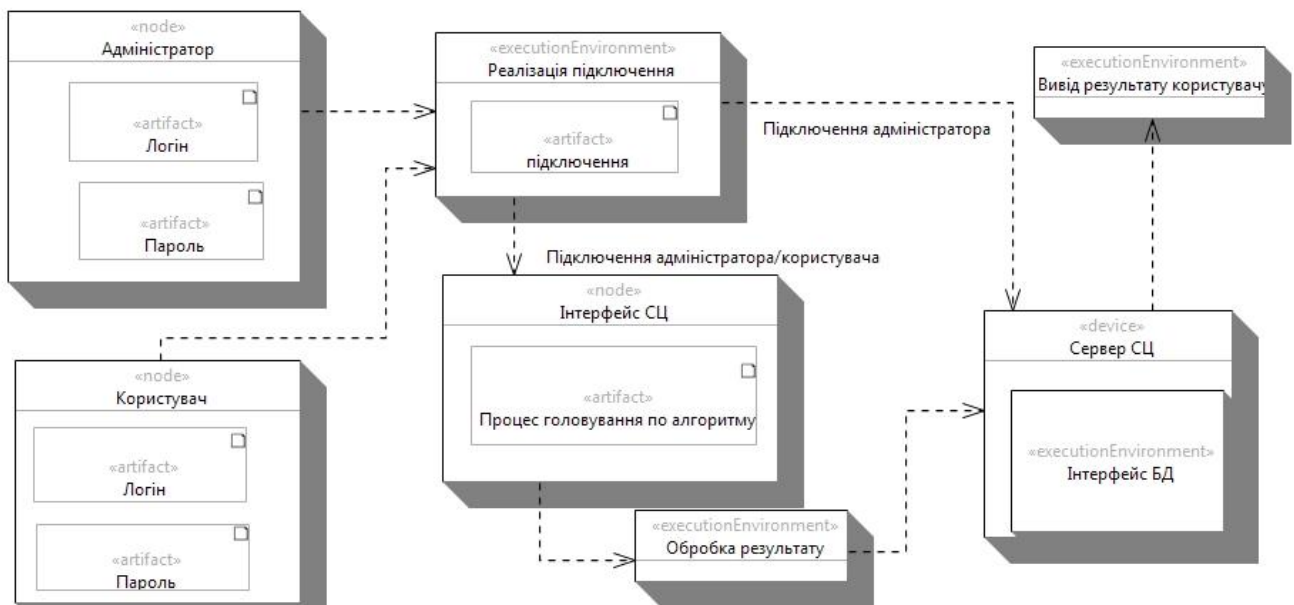


Рисунок 6.16 – Діаграма розгортання СЦ

Кожен вузол на діаграмі являє собою певний тип комп'ютерного компонента, який, в більшості випадків, є частиною інфраструктури. Ця інфраструктура може бути розподіленою, інтегрованою або конвергентною. На рисунку 6.16 показано 4

компонента інфраструктури призначені для реалізації трьох процесів процедури вибору альтернатив шляхом голосування в ССУ.

6.3 Аналіз завантаження інфраструктури ССУ

Розробка складних інформаційних систем, зокрема систем підтримки прийняття рішень, вимагає проведення попереднього дослідження таких систем, у тому числі з використанням імітаційних моделей, для перевірки ефективності закладених в основу системи концепцій, архітектурних і конструктивних рішень. Враховуючи складність організації проблемно-орієнтованих інформаційних систем, їх слід розглядати як багаторівневі архітектури, кожен рівень в яких забезпечує функціональність відповідного рівня абстракції моделі інформаційної системи. У моделі проблемно-орієнтованої інформаційної системи можна виділити такі рівні представлення (абстракції):

1. Протокольний (мережевий, інфраструктурний) рівень.
2. Операційний (командний) рівень.
3. Процедурний рівень – стандартні (типові) процедури функціонування.
4. Організаційний рівень – організовані сукупності процедур для забезпечення вирішення задач функціонування інформаційних систем.
5. Інфологічний рівень – бази даних і знань.
6. Проблемний рівень – рівень вирішення прикладних задач.

Кожен рівень характеризується множинами таких компонентів:

- алфавітом;
- словником (станами відповідного рівня);
- правилами обробки слів (функціями);
- правилами побудови ієрархій (розподілом функцій між рівнями);
- правилами трансформації функцій на кожному рівні (інтерфейси, суперпозиція функцій).

Враховуючи складність і багаторівненість абстрактного опису проблемно-орієнтованої інформаційної системи, до імітаційної моделі системи можна висунути такі вимоги:

- апаратна (конструктивна) інтерпретація;
- можливість коректного ієрархічного подання;
- можливість представлення паралельних процесів;
- можливість оперування параметрами середовища (змінні, стани, переходи, час та ін.);
- можливість розширення функцій моделі системи користувачем доступними засобами (функції користувача);
- можливість автоматичної генерації програмного коду;
- урахування історії станів компонентів системи.

Особливістю функціонування ССУ як окремого типу проблемно-орієнтованих інформаційних систем є необхідність забезпечення в них функціонування в режимі реального часу, підтримка процедур і процесів колективного (групового) обговорення й прийняття рішень. Тобто виникає необхідність моделювання розподіленого гетерогенного керованого подіями (реактивного) інформаційного середовища, що функціонує в реальному часі. Будемо вважати, що керована подіями (реактивна) система – це динамічна система, яка сприймає зовнішні дискретні впливи і відповідає своїми реакціями на ці впливи.

Таким чином, для імітаційного моделювання ССУ необхідно визначити такі характеристики компонентів кожного із рівнів:

- стани;
- переходи (безумовні);
- події (визначаються середовищем функціонування);
- дані (визначаються середовищем функціонування);
- вузлові точки (умовні переходи, точки прийняття рішень);
- умови переходів;
- переходи за умовчанням;
- дії (підтримується мовою дій);

- паралелізм;
- ієрархія (відношення предок-нащадок) для станів і переходів;
- хронологія спрацювання.

Ефективним засобом аналітичного дослідження та імітаційного моделювання структурованих реактивних дискретних систем є програмне середовище комп'ютерної математики MATLAB-Simulink із пакетом StateFlow.

Засоби MATLAB-Simulink-StateFlow дозволяють створювати ієрархічні розподілені імітаційні моделі динамічних систем, які можуть містити різноманітні компоненти і відображати як функціональний, так і структурний аспекти побудови модельованої системи. Для опису імітаційної моделі застосовується математична модель динамічної системи, що визначається кортежем [309]:

$$MM_{DC} = \langle T, X, \Omega, H, Z, Y, \Phi_1, \Phi_2 \rangle, \quad (6.4)$$

де T – універсум відношень між елементами системи, $X = X_1 \times \dots \times X_{NX}$ – простір вхідних впливів, $\Omega = \Omega_1 \times \dots \times \Omega_{N\Omega}$ – простір впливів зовнішнього середовища, $H = H_1 \times \dots \times H_{NH}$ – простір внутрішніх параметрів системи, $Z = Z_1 \times \dots \times Z_{NZ}$ – простір внутрішніх станів системи, $Y = Y_1 \times \dots \times Y_{NY}$ – простір вихідних і залежних змінних; $\Phi_1(z_0, \mathbf{x}(t), \mathbf{h}(t), \boldsymbol{\omega}(t), t) = \mathbf{z}(t) = (z_1(t), \dots, z_{NZ}(t))^T \in Z$ – оператор зміни внутрішніх станів системи в залежності від початкового внутрішнього стану та зміни з часом вхідних впливів, внутрішніх параметрів, зовнішнього середовища, $\Phi_2(\mathbf{z}(t), t) = \mathbf{y}(t) = (y_1(t), \dots, y_{NY}(t))^T \in Y$ – оператор зміни виходів системи в залежності від зміни внутрішніх станів системи.

Така математична модель є цілком адекватною при моделюванні систем на одному рівні ієрархії (горизонтальна модель) і не відображає трансформацію функцій і цілей системи на різних рівнях ієрархії. Тому на основі моделі (6.4) може бути створена ієрархічна математична модель [9]:

$$IMM_{DC} = \langle MM_{DC}^i, \Phi_3^{i-1}(\tilde{T}_{i-1}) \rangle, \quad (6.5)$$

де $i=1, \dots, m$ – номер рівня абстракції представлення моделі, $\Phi_3^{i-1}(\tilde{T}_{i-1})$ – оператор трансформації функцій і цілей між рівнями абстракції $i-1$ та i , $\tilde{T}_{i-1} = T_i \cap T_{i-1}$ – відношення між рівнями абстракції імітаційної моделі. Оскільки характеристики модельного середовища кожного з рівнів однакові, то відмінність моделювання різних рівнів полягає у семантиці компонентів моделі кожного з рівнів, причому семантика компонентів повинна відповідати рівню абстракції представлення даного рівня.

Крім названих вище вимог до ССУ та особливостей їх використання, слід також зауважити про необхідність забезпечення багатofункціональності, адаптивності та ефективності застосування таких систем на основі типових архітектурних рішень. Тому задача створення ССУ певного призначення формулюється як задача конфігурування заданої інформаційної інфраструктури для вирішення конкретних прикладних задач з урахуванням обмежень цієї інфраструктури. Цільова функція побудови проблемно-орієнтованої інформаційної системи може бути сформульована в вигляді набору вимог до ресурсів – обчислювальних, інформаційних, комунікаційних, організаційних, часових та інших, а обмеження інфраструктури – у вигляді наявної можливості надання ресурсів, у яких є потреба. Деякі з підходів до оптимізації архітектур інформаційних систем з урахуванням наявних ресурсів та ефективності їх застосування описані в роботах.

Таким чином, запропонована ієрархічна математична модель забезпечує адекватне представлення семантики різних рівнів представлення системи без втрати цілісності подання системи у цілому і може бути використана для імітаційного моделювання проблемно-орієнтованих інформаційних систем із застосуванням пакета MATLAB-Simulink-StateFlow на основі теорії масового обслуговування.

Системи масового обслуговування (СМО), до яких відносяться ССУ, є класом математичних схем, розроблених у теорії масового обслуговування і різних застосуваннях для формалізації процесів функціонування систем, які за своєю суттю є процесами обслуговування.

Як процес обслуговування можуть бути представлені різні за своєю фізичною природою процеси функціонування економічних, виробничих, технічних та інших систем, наприклад, потоки постачань продукції деякому підприємству, потоки деталей і комплектуючих виробів на складальному конвеєрі цеху та ін. При цьому характерною для таких об'єктів є випадкова поява заявок (вимог) на обслуговування і завершення обслуговування у випадкові моменти часу, тобто стохастичний характер процесу їх функціонування.

Таким чином, можуть бути представлені різні за своєю фізичною природою процеси: економічні, технічні, виробничі та ін.

У СМО можна виділити два стохастичні процеси:

- надходження заявок на обслуговування;
- обслуговування заявок.

Потік подій – послідовність подій, що відбуваються одна за одною в деякі моменти часу. У СМО виділятимемо два потоки:

- вхідний потік – множина моментів часу надходження в систему заявок;
- потік обслуговування – множина моментів закінчення обробки системою заявок.

У загальному випадку СМО елементарного виду може бути представлена так, як це показано на рис. 6.17.

Потік подій однорідний, якщо він характеризується тільки моментами надходження цих подій і задається послідовністю $\{t_n\}$, де $\{t_n\} = \{0 \leq t_1 \leq t_2 \leq \dots \leq t_n \leq \dots\}$.

Потік подій неоднорідний, якщо він задається послідовністю $\{t_n, f_n\}$, де t_n – моменти надходження подій, а f_n – набір ознак події (пріоритети, приналежність тому або іншому джерелу, тип каналу для його обслуговування та ін.).

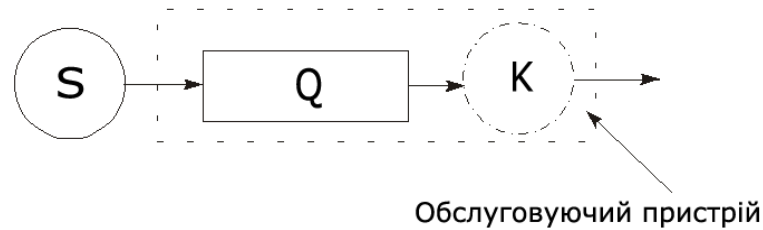


Рисунок 6.17 – Система масового обслуговування
елементарного виду: S – джерело, Q – накопичувач (черга), K – канал
обслуговування

Потік регулярний, якщо події поступають через рівні проміжки часу. Відповідно, потік нерегулярний, якщо інтервали між подіями є випадкові величини.

Режим обробки заявок на обслуговування, який використовується при надходженні в систему обслуговування, коли зайняті усі обслуговуючі прилади, визначає завдання типу СМО:

- системи із втратами (відмовами);
- системи з чергами.

У системах із чергами заявки, які не можуть бути обслужені відразу, складають чергу і за допомогою деякої дисципліни обслуговування вибираються з неї. Найбільш вживані дисципліни обслуговування:

- 1) FIFO (first in - first out) – в порядку надходження;
- 2) LIFO (last in – first out) – першою обслуговується заявка, яка поступила останньою;
- 3) SIRO (service in random order) – обслуговування у випадковому порядку;
- 4) Pr – пріоритетні системи.

Нотація Кендалла

Для короткого опису СМО Д. Кендалл увів символіку (нотацію) такого виду [310]:

$$A/B/S/m,$$

де S – число обслуговуючих приладів, m – кількість місць очікування (місткість накопичувача). A і B характеризують, відповідно, потік заявок і потік обслуговування, задаючи функцію розподілу інтервалів між заявками у вхідному потоці і функцію розподілу часів обслуговування. A і B можуть набувати таких значень:

- D – детермінований розподіл;
- M – показове (марківський процес);
- Er – розподіл Ерланга порядку r ;
- Hr – гіперпоказове;
- G – розподіл загального вигляду.

При цьому мається на увазі, що потоки є рекурентними, тобто інтервали між подіями незалежні і мають однаковий розподіл.

Іноді в нотацію Д. Кендалла додають ще один символ $/k$ – кількість джерел заявок [310]:

$$A/B/S/m[/k].$$

Базовою моделлю для аналізу систем масового обслуговування є простий (елементарний) потік [261]. Для простого потоку число подій, які потрапляють на будь-який фіксований інтервал часу, підкоряється закону Пуассона, тому його також називають стаціонарним пуассонівським потоком. Для пуассонівського потоку ймовірність того, що за інтервал часу τ станеться рівно m подій, визначається як [311]:

$$P_m(\tau) = \frac{(\lambda \tau)^m}{m!} e^{-\lambda \tau}, \quad (6.6)$$

де λ – інтенсивність простого потоку заявок.

Умова відсутності залежності між надходженням заявок на обслуговування (заявки поступають незалежно одна від одної) є найсуттєвішою умовою для

простого потоку. Важлива роль простого потоку при моделюванні визначається такими обставинами:

- 1) прості або близькі до них потоки часто зустрічаються на практиці;
- 2) при аналізі СМО можна отримати цілком задовільні результати, замінюючи вхідний потік будь-якої структури простим із тією ж інтенсивністю;
- 3) при підсумуванні великого числа ординарних стаціонарних потоків із практично будь-яким наслідком виходить потік як завгодно близький до простого (умова – потоки повинні мати рівномірно малий вплив на суму потоків).

Простий потік є стохастичним потоком, який задає найбільш напружений режим роботи для моделі, і тому реальні характеристики модельованої системи будуть не гіршими, ніж отримані в результаті експерименту з заміною реальних потоків на прості оцінки характеристик потоків із пуассонівським розподілом надходження заявок на обслуговування.

Діяльність пов'язана з ситуаційним управлінням у ССУ проводиться на базі організаційно-технічних комплексів ситуаційних залів (ситуаційних кімнат, ситуаційних центрів), обладнаних мережевими комп'ютерними засобами та автоматизованими робочими місцями (АРМ) учасників процесів ситуаційного управління. Для створення і дослідження властивостей мережі Ситуаційного залу (СЗ) використовуються засоби MATLAB-Simulink. Метою дослідження є визначення параметрів завантаження мережі в залежності від інтенсивності надходження заявок (або опитування) АРМів СЗ.

Однією з найпростіших моделей СМО є модель із детермінованим розподілом та обслуговуванням заявок на одному обслуговуючому пристрої, тобто, за нотацією Кендалла, D/D/1. Реалізація моделі СМО типу D/D/1 у середовищі MATLAB-Simulink з детермінованим потоком заявок і детермінованим часом обслуговування має вигляд, наведений на рис. 6.18 [312].

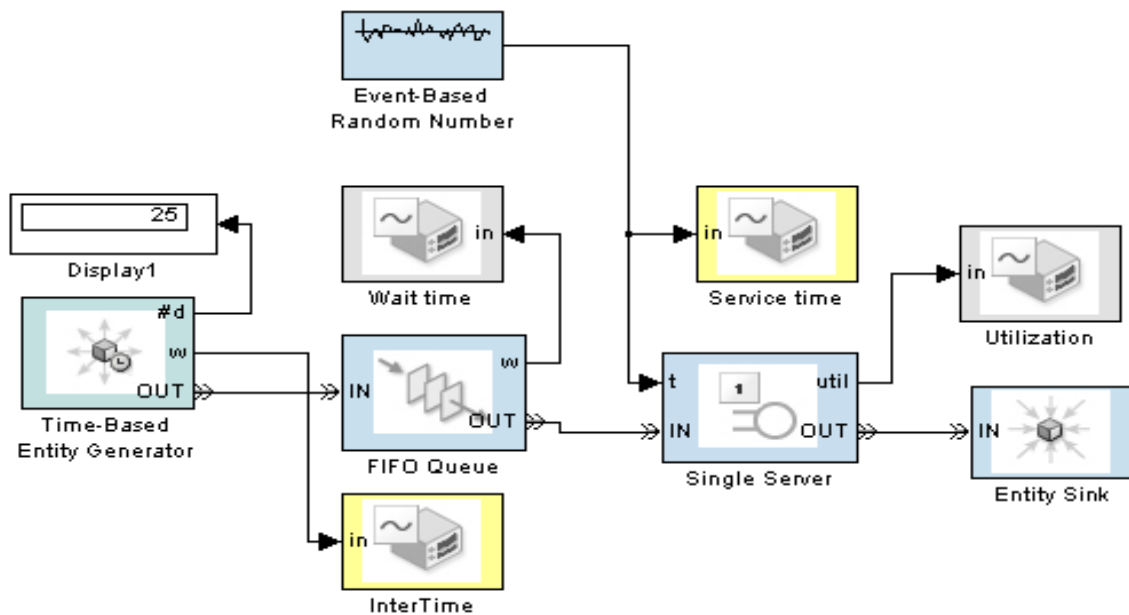


Рисунок 6.18 – Модель СМО типу D/D/1

Залежність між інтенсивністю надходження заявок і завантаженням сервера наведено на рис. 6.19, а залежність між інтенсивністю надходження заявок і часом очікування в черзі наведено на рис. 6.20.

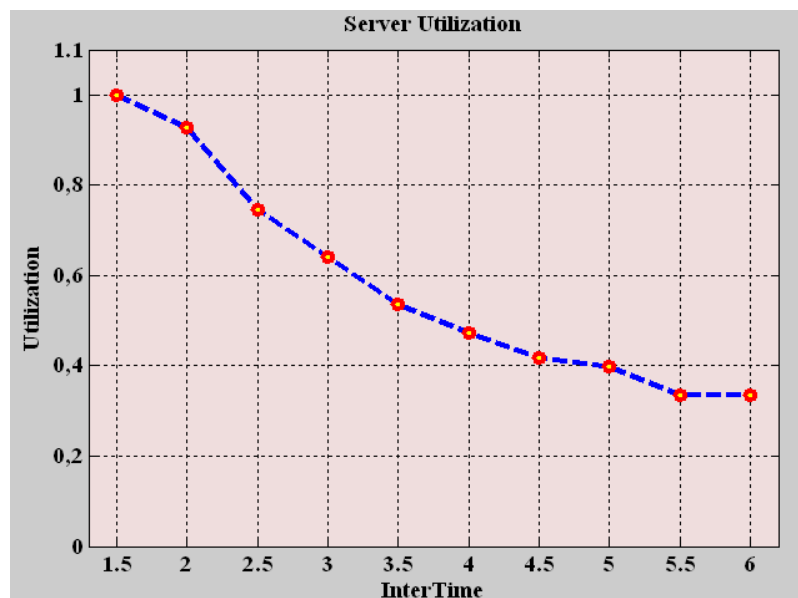


Рисунок 6.19 – Залежність між інтенсивністю надходження заявок і завантаженням сервера

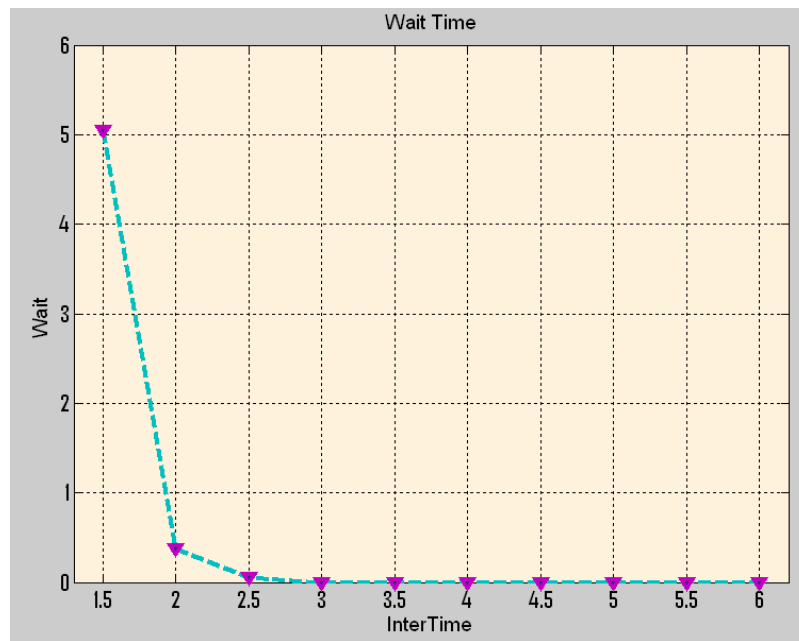


Рисунок 6.20 – Залежність між інтенсивністю надходження заявок і часом очікування в черзі

Більш наближеною до реальних умов є модель із показовим розподілом надходження і обробки заявок на одному обслуговуючому пристрої (марківський процес). Модель СМО типу М/М/1 має вид, наведений на рис. 6.21.

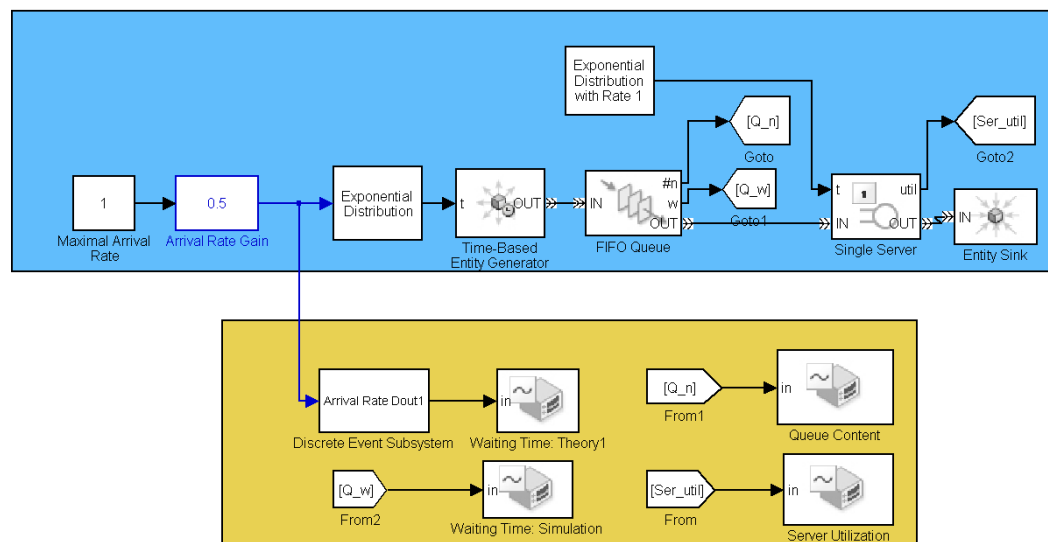
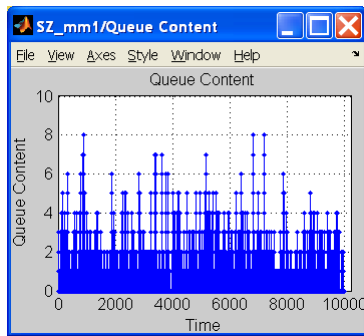
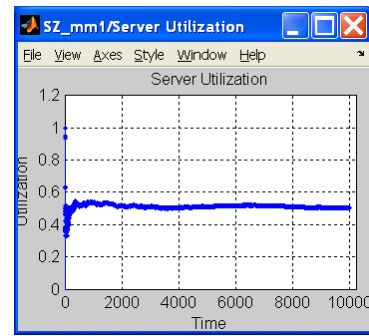


Рисунок 6.21 – Модель СМО типу М/М/1

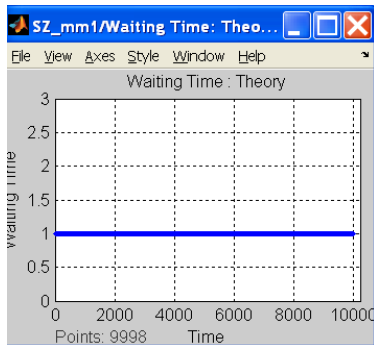
Співвідношення між теоретичними параметрами та моделлю при різних інтенсивностях потоку заявок показано на рис. 6.22, 6.23.



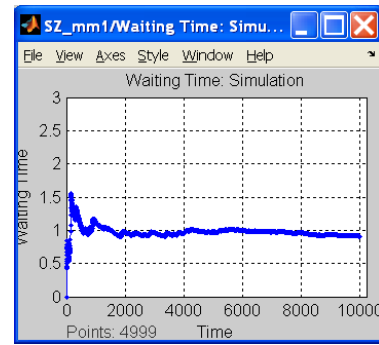
а) надходження заявок



б) використання сервера

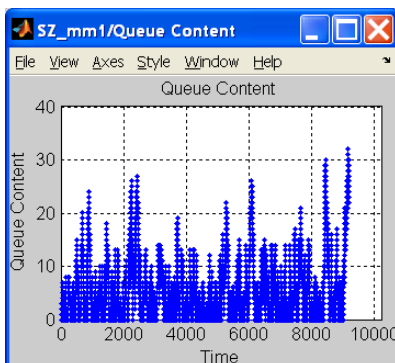


в) теоретичне очікування

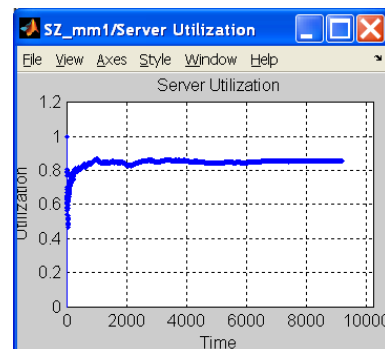


г) модельне очікування

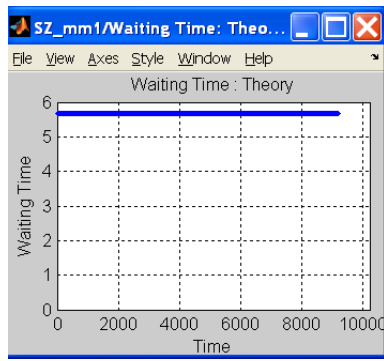
Рисунок 6.22 – Інтенсивність надходження заявок $\lambda_p=0,5$



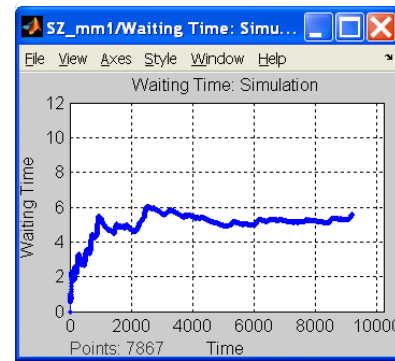
а) надходження заявок



б) використання сервера



в) теоретичне очікування



г) модельне очікування

Рисунок 6.23 – Інтенсивність надходження заявок $\lambda_p=0,85$

Як видно з результатів моделювання в системах із марківськими процесами, надходження і обробка заявок при інтенсивностях надходження заявок вище середнього рівня значно збільшується навантаження на сервер і чергу, що при обмеженості черги може призвести до втрат пакетів.

Модель СМО типу M/D/1 із марківським потоком заявок і детермінованим часом обслуговування має вигляд, наведений на рис. 6.24.

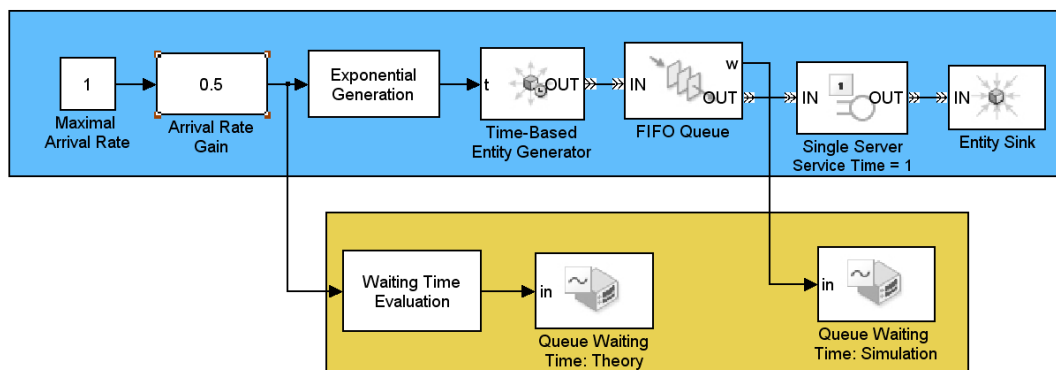
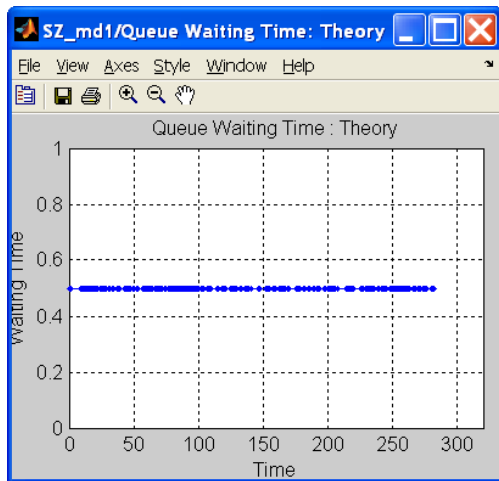
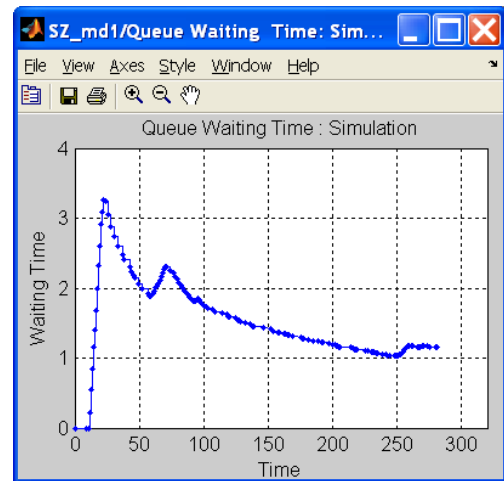


Рисунок 6.24 – Модель СМО типу M/D/1

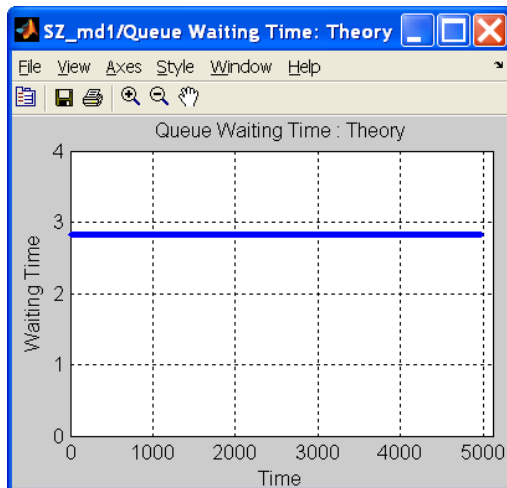
Співвідношення між теоретичними параметрами та моделлю при різних інтенсивностях потоку заявок показано на рис. 6.25, 6.26.



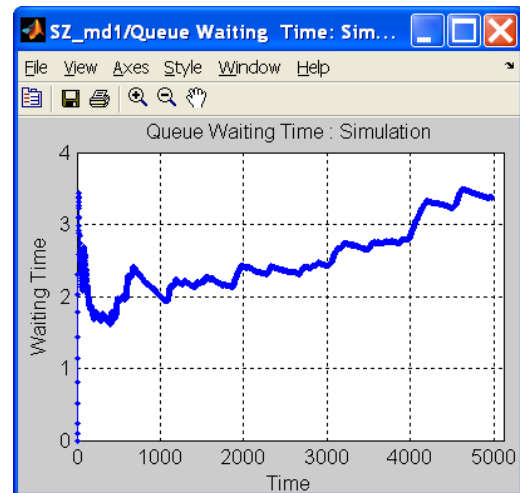
а) теоретичне очікування



б) модельне очікування

Рисунок 6.25 – Інтенсивність надходження заявок $\lambda_p=0,5$ 

а) теоретичне очікування



б) модельне очікування

Рисунок 6.26 – Інтенсивність надходження заявок $\lambda_p=0,85$

Як видно з результатів моделювання, зі збільшенням потоку заявок очікування у черзі зростає повільніше у моделі M/D/1 порівняно з моделлю M/M/1.

6.4 Висновки до розділу 6

Конвергентний підхід до побудови систем ситуаційного управління дозволяє створити відповідну задачам етапу ситуаційного управління конфігурацію компонентів системи. Важливим етапом ситуаційного управління є підготовка персоналу і користувачів. Основою технологій електронного навчання є складні програмно-технічні комплекси, які забезпечують організаційну підтримку, широкую функціональність, гнучкість і масштабованість середовищ електронного навчання. Ситуаційний характер підготовки користувачів ССУ вимагає створення засобів автоматизації адміністрування проєктів електронного навчання з можливістю оперативного визначення параметрів проєктів. На основі методів структурної оптимізації конвергентних систем розроблено систему управління проєктами конвергенції систем електронного навчання на етапах підготовки персоналу ССУ.

Однією з форм реалізації систем ситуаційного управління є ситуаційні центри. Сучасний підхід до створення ССУ базується на композиційній конвергенції відповідних сервісів в рамках ситуаційних агентів. Колективне прийняття рішень в процесі СУ здійснюється шляхом проведення нарад.

Розроблений інформаційний сервіс підтримки проведення нарад у контексті ситуаційного управління забезпечує автоматичну підготовку, підтримку ведення та формування матеріалів таких нарад із використанням як з реляційних, так і файлових баз даних із можливостями ручного введення інформації. Модель організації інформаційних об'єктів забезпечує відповідність регламенту проведення нарад і фіксацію фактичних матеріалів нарад, доступних через кореневий об'єкт. Розроблений сервіс підтримки проведення нарад у процесі ситуаційного управління передбачає автоматичну підготовку і формування імманентних баз даних. Автоматична підготовка баз даних здійснюється відповідно до структурної моделі наради. Представлений сервіс може бути розвинутий у напрямі забезпечення мобільного доступу (зі смартфонів, планшетів тощо) для оперативної участі у нарадах шляхом додавання відповідних інтерфейсів платформ без реінжинірингу

ядра розробленого сервісу обслуговування нарад. У цьому випадку користувачі також зможуть використовувати такі служби обміну повідомленнями, як чати, миттєві повідомлення, відеоконференції, дані про місцезнаходження тощо. Запропонована квазіформальна структурна метамодель наради в XML-подібному форматі дозволяє більш компактно представити семантичне ядро структурної моделі наради в порівнянні з використанням строгого XML-синтаксису. Квазіформальна структурна метамодель спрощує створення лінгвістичних конструкцій при реалізації сервісу підтримки проведення нарад відповідно до вимог конкретної платформи реалізації.

У процесі проведення нарад здійснюється, у тому числі експертне оцінювання альтернативних рішень. Одним із підходів до рейтингування та вибору альтернативних рішень є експертне оцінювання з використанням методів голосування. Реалізація експертного прийняття рішень шляхом голосування потребує комп'ютерної підтримки. Для побудови системи підтримки процедур голосування розроблено набір моделей, які можуть бути використані при створенні відповідних ситуаційних агентів.

При побудові конвергентних ССУ виникає необхідність моделювання розподіленого гетерогенного керованого подіями (реактивного) інформаційного середовища, що функціонує в реальному часі. Таке моделювання проводилось із використанням моделей теорії масового обслуговування. Результати моделювання завантаження інфраструктури показали, що інтенсивність заявок на обслуговування сервером не повинна перевищувати максимального часу обробки транзакції сервером.

Основні результати розділу опубліковані автором у працях [4, 9, 10, 13, 21, 28, 36, 37, 42, 44, 46, 58–73, 75, 77, 81–84].

ВИСНОВКИ

У дисертаційній роботі вирішена актуальна науково-технічна проблема розвитку теорії побудови конвергентних комп'ютерних систем ситуаційного управління агентного типу шляхом розробки нових онтологічних, процесних, архітектурних, композиційних моделей та розвитку методів оптимізації структури і динаміки таких систем.

Розроблені моделі та методи забезпечують структурну ефективність побудови і динамічну стійкість функціонування конвергентних систем ситуаційного управління з урахуванням особливостей сфери їх використання.

Під час дослідження були отримані такі наукові та практичні результати:

1. На основі аналізу класифікацій ситуаційних систем запропоновано узагальнену класифікацію ситуаційних систем, яка враховує особливості цільового призначення ситуаційної системи та є основою для побудови спеціалізованих моделей знань предметних областей ситуаційної діяльності.

2. На основі аналізу категорій інформації та фокусу їх практичного використання запропоновано модель трансформації інформації в циклі ситуаційного управління, виходячи зі способу представлення та контексту використання інформації, яка поєднує в єдину систему різні категорії інформації й забезпечує її адекватне використання в різних видах моделей, зокрема, у спеціалізованих моделях знань ситуаційного управління.

3. На основі аналізу моделей когнітивного циклу ситуаційного управління розроблено повну формальну модель циклу ситуаційного управління з використанням відповідних систем логічного умовиводу і прийняття рішень на кожному етапі трансформації інформації ситуаційного управління: абдукції для побудови формальних моделей гіпотез ситуацій, індукції для побудови формально обґрунтованої теоретичної моделі ситуації на основі гіпотез, прийняття рішень щодо реагування на ситуацію шляхом дедуктивного виводу з формальної моделі ситуації або шляхом застосування механізмів умовиводу на основі прецедентів, виходячи з

попереднього досвіду, формування і виконання керівних директив на основі прийнятих рішень.

4. Запропоновано агентно-орієнтовану архітектуру ситуаційних систем виходячи з особливостей когнітивної моделі перцептивного циклу та моделі трансформації інформації в ситуаційних системах. При описі поведінки «розумних» агентів враховується їх обмежена раціональність, пов'язана із проблемою вибору, обмеженнями знань, доступними ресурсами тощо. Обмежена раціональність є основою математичного моделювання при прийнятті рішень, коли прийняття рішень розглядається як раціональний процес пошуку оптимального вибору із множини альтернатив з урахуванням наявної інформації.

5. Запропоновано і розроблено метод побудови композиційної моделі конвергенції технологій ситуаційного управління з використанням моделей знань. Встановлення відповідності між формальним описом предметної області і засобами інформаційних технологій дозволило застосувати орієнтований на знання підхід до розробки систем ситуаційного управління. Запропонований підхід забезпечує формування репозиторію та конвергенцію засобів інформаційних технологій для вирішення цільових проблем ситуаційного управління шляхом композиції етапів онтологічного аналізу вимог, функціональної декомпозиції, предметної інтерпретації та фізичної реалізації. Використання запропонованої моделі процесу проектування забезпечує формалізований синтез ССУ для цільової предметної області шляхом конвергенції потрібних компонентів на основі моделей знань.

6. Розвинуто оптимізаційні методи побудови ефективної архітектури ситуаційної системи за узагальненим показником ефективності та шляхом багатопараметричної оптимізації за критерієм Парето, для якої реалізовано прототип веб-орієнтованого програмного застосунку.

7. Розроблено метод та алгоритм оцінки динамічної стійкості системи ситуаційного управління як дисипативної системи на основі аналізу зміни рівня інформаційної ентропії, яка визначає степені свободи системи при виборі шляхів у гіпотетичному динамічному ансамблі всіх можливих фіксованих тривалостей шляхів, що переходять один в інший, аналогічно переходам між конфігураційними

мікростанами ідеальної фазової траєкторії, що має відповідну ентропійну пружність, представлену знаннями та обізнаністю агентів.

8. Набули подальшого розвитку математичні методи аналізу динамічної стійкості стохастичної системи на основі принципу стійкості за Ляпуновим. Запропоновано та досліджено агентно-орієнтований підхід для включення моделі стійкості за Ляпуновим у процес ситуаційного управління динамічною стохастичною системою як цільової проєктної діяльності.

9. Удосконалено моделі аналізу подіє-орієнтованих систем на основі теорії масового обслуговування. Застосування математичного апарату систем масового обслуговування дозволяє провести таке моделювання для різних дисциплін обслуговування з різною кількістю обслуговуючих пристроїв. Для моделювання було обрано архітектуру системи масового обслуговування з одним обслуговуючим пристроєм та множиною пристроїв – генераторів заявок. Моделювання проводилось для комбінованих режимів надходження та обслуговування заявок, що відповідають детермінованому та марківському процесам. Імітаційне моделювання проводилось у порівнянні з теоретичними моделями.

10. Розроблено онтологічну модель знань для використання у системі ситуаційного реагування на інциденти безпеки, яка дозволяє забезпечити конвергентне поєднання в єдиній базі знань архітектурного, ризик-орієнтованого та адаптивних до уразливостей аспектів управління безпекою інформаційних комп'ютерних систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] А. Морозов, Л. Баран, В. Вишневський, О. Коваленко и О. Малишев, «Розділ 6. Ситуаційний центр як засіб колективної роботи законотворців,» в *Електронний парламент України: досвід створення. Наукове видання*, С. Довгий, Ред., Київ, ТОВ «Видавництво «Юстон», 2015, pp. 395-441.
- [2] О. Kovalenko и D. Velev, «Ad-Hoc Architecture of Systems for Disaster Risk Management,» в *Information Technology in Disaster Risk Reduction. IFIP Advances in Information and Communication Technology*, т. 575, Y. Murayama, D. Velev и P. Zlateva, Ред., Cham, Springer, 2020, pp. 135-145.
- [3] О. Kovalenko, V. Visnevsky и V. Kosolapov, «Models of information processing optimization for technical interoperability in a network of distributed situational centers,» *CEUR Workshop Proceedings*, т. 2608, pp. 426-435, 2020.
- [4] О. Є. Коваленко и В. Косолапов, «Моделі оптимізації завантаження мережі розподілених ситуаційних центрів,» *Математичне та комп'ютерне моделювання. Серія: Технічні науки*, № Випуск 21, pp. 102-113, 2020.
- [5] О. Є. Коваленко и В. Косолапов, «Модель стійкості агентної системи ситуаційного управління,» *Математичні машини і системи*, № 3, pp. 93-104, 2020.
- [6] О. Коваленко, «Онтологія та модель трансформації інформації в ситуаційних агентних ситсемах,» *Електронне моделювання*, т. 42, № 5, pp. 5-23, 2020.
- [7] О. Є. Коваленко, «Принципи інженерії ситуаційних систем,» *Математичні машини і системи*, № 4, pp. 65-78, 2019.
- [8] О. Kovalenko, «Knowledge Based Design of Convergent Systems of Situational Management,» *Математичні машини і системи*, № 3, pp. 67-74, 2019.
- [9] О. Є. Коваленко, «Моделювання навантаження мережної інфраструктури систем ситуаційного управління,» *Математичні машини і системи*, № 3, pp. 101-110, 2019.

[10] Т. Власова, О. Коваленко и В. Косолапов, «Організаційно-інформаційний сервіс для технології прийняття рішень у ситуаційному центрі шляхом проведення нарад,» *Математичні машини і системи*, № 1, pp. 68-79, 2019.

[11] О. Є. Коваленко, «Композиційна конвергенція інформаційних технологій у системах ситуаційного управління на основі моделі знань предметної області,» *Математичне моделювання в економіці*, т. 16, № 2, pp. 39-44, 2019.

[12] О. Коваленко, «Модель оцінки ефективності архітектури конвергентної системи ситуаційного управління,» *Моделювання та інформаційні технології: Збірник наукових праць ІПМЕ ім. Г.Є.Пухова НАН України*, № 87, p. 11–14, 2019.

[13] О. Коваленко, «Системна інженерія та життєвий цикл систем,» *Електронне моделювання*, т. 40, № 6, p. 61–82, 2018.

[14] О. Коваленко, «Моделі агентно-орієнтованих систем ситуаційного управління,» *Математичні машини і системи*, № 2, pp. 96-102, 2018.

[15] О. Е. Kovalenko и Т. Kovalenko, «Knowledge Model for Decision Support System Security Management,» *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2016), Proceedings*, pp. 165-170, 19 February 2019.

[16] О. Е. Kovalenko, «The Formalization of Organizational Support Creation for Systems of Situational Management,» *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education Proceedings*, pp. 292-301, 15 August 2016.

[17] О. Коваленко, «Побудова багаторівневих мультиагентних інформаційних систем заснованих на знаннях,» *Information Technology and Security*, т. 4, № 2(7), pp. 146-154, 2016.

[18] О. Коваленко, «Застосування хмарних технологій при побудові систем ситуаційного управління,» *Information Technology and Security*, т. 3, № 1, pp. 11-18, 2015.

[19] О. Е. Kovalenko, «Knowledge Models for Organizational Maintenance of Situation Centers,» *Proceedings of International Conference on Application of*

Information and Communication Technology and Statistics in Economy and Education, pp. 241-248, 15 September 2015.

[20] О. Є. Коваленко, «Застосування модальної логіки при прийнятті рішень на моделях знань,» *Математичне та комп'ютерне моделювання. Серія: Технічні науки: зб. наук. праць*, № 6, pp. 106-112, 2012.

[21] О. Коваленко, «Оптимізація архітектур модульних систем електронного навчання,» *Математичне та комп'ютерне моделювання. Серія: Технічні науки : зб. наук. праць*, № 1, pp. 95-99, 2008.

[22] A. Kovalenko и D. Velev, «Compositional Model of Design Process in the Problem-Oriented CAD of Applied Software,» *Engineering Simulation*, т. 16, p. 129 – 133, 1998.

[23] A. Kovalenko и D. Velev, «Computer-aided programming system for intelligent measurement systems in electronics,» *Engineering Simulation*, т. 12, № 4, pp. 642-649, 1995.

[24] О. Kovalenko, V. Vishnevsky и V. Kosolapov, «Towards Creating the Network of Situational Governance Centers and Decision Making Technologies in Distributed Environments,» *15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering – Proceedings (TCSET-2020)*, Т. %1 из %22020-May, pp. 540-545, 07 May 2020.

[25] О. Kovalenko, «Information Taxonomy and Ontology for Situational Management,» *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 – Proceedings*, т. 2, pp. 94-97, 7 November 2018.

[26] О. Kovalenko и Т. Kovalenko, «Knowledge Model and Ontology for Security Services,» *2018 IEEE 1st International Conference on System Analysis and Intelligent Computing, SAIC 2018 – Proceedings*, p. Article number 8516875, 31 October 2018.

[27] V. Kosolapov и О. Kovalenko, «Agent Based Modelling Support for Systems of Situational Management,» *2018 IEEE 1st International Conference on System Analysis and Intelligent Computing, SAIC 2018 – Proceedings*, pp. 244-248, Article number 8516818, 31 October 2018.

[28] T. Vlasova, O. Kovalenko и V. Kosolapov, «Organizational-Information Technology for Providing and Decisions Making in Situational Management,» *14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2018 – Proceedings Volume 2018-April*, pp. 152-157, 10 April 2018.

[29] О. Коваленко и V. Kosolapov, «Optimization of Loading of a Network of Distributed Situational Centers,» *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : тези доповідей 9-ї Міжнародної наукової конференції*, pp. 89-91, 14-15 Травня 2020.

[30] О. Є. Коваленко, «Архітектурний підхід до управління життєвим циклом систем управління інформаційною безпекою на основі моделей знань,» в *Кібербезпека енергетики : збірка праць конференції*, Київ , Інститут проблем моделювання в енергетиці ім.Г.Є. Пухова НАН України, 2019, pp. 5-7.

[31] О. Коваленко, «Елементи архітектури систем кібер-енергетики,» в *Безпека енергетики в епоху цифрової трансформації : матеріали науково-практичної конференції*, Київ, Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, 2019, pp. 69-71.

[32] О. Коваленко, «Онтологія та трансформація інформації в системах ситуаційного управління,» в *Сучасні проблеми інформатики в економіці, управлінні, освіті та подоланні наслідків Чорнобильської катастрофи : матеріали XVII Міжнародного наукового семінару*, Київ , Національна академія управління, 2018, pp. 12-16.

[33] О. Е. Kovalenko, «On Consolidation of Information in Systems of Situational Management,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : тези доповідей 8-ї Міжнародної наукової конференції*, Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2018, pp. 118-119.

[34] О. Є. Коваленко и Т. О. Коваленко, «Конвергенція та конфігурування інформаційних технологій в системах ситуаційного управління,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : тези доповідей 8-ї*

Міжнародної наукової конференції, Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2018, pp. 122-123.

[35] О. Коваленко, «Інформаційна технологія консолідації даних в системах ситуаційного управління,» в *Сучасні проблеми інформатики в економіці, управлінні, освіті та подоланні наслідків Чорнобильської катастрофи : матеріали XVI Міжнародного наукового семінару*, Київ, Національна академія управління, 2017, pp. 13-17.

[36] О. Коваленко, «Ситуаційне управління як проектна діяльність,» *Системи підтримки прийняття рішень. Теорія і практика*, pp. 28-31, 5 червня 2017.

[37] D. Veleв и О. Kovalenko, «Research on the Applicability of Virtual Reality in Disaster Risk Reduction,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей одинадцятої дистанційної науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2017, pp. 32-35.

[38] Т. Коваленко и О. Коваленко, «Модель знань для управління безпекою інформаційної системи,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей одинадцятої дистанційної науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2017, pp. 113-116.

[39] О. Коваленко, «Обработка модальностей при принятии решений у системах ситуаційного управління,» в *Теорія прийняття рішень : праці VIII міжнародної школи-семінару*, Ужгород, Ужгородський національний університет, 2016, pp. 139-140.

[40] О. Коваленко, «Застосування онтологій в системах ситуаційного управління,» в *Сучасні проблеми інформатики в управлінні, економіці, освіті та подоланні наслідків Чорнобильської катастрофи : матеріали XV Міжнародного наукового семінару*, Київ, Національна академія управління, 2016, pp. 84-89.

[41] О. Коваленко, «Технологічні моделі ситуаційного управління,» в *Проблеми виведення з експлуатації об'єктів ядерної енергетики та відновлення*

оточуючого середовища INUDECO'16 : збірка матеріалів першої міжнародної конференції, Славутич , Славутицька філія НТУУ «КПІ», 2016, pp. 166-168.

[42] О. Є. Коваленко и Т. О. Коваленко, «Елементи модельно-орієнтованого підходу технології розробки систем ситуаційного управління,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : тези доповідей VII міжнародної наукової конференції*, Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2016, pp. 97-98.

[43] О. Коваленко, «Формалізація діяльності систем ситуаційного управління,» в *Сучасні проблеми інформатики в управлінні, економіці та освіті : матеріали XIV-го міжнародного наукового семінару*, Київ, СІК ГРУП Україна, 2015, pp. 119-123.

[44] О. Коваленко, «Стандартизація формального опису системної архітектури ситуаційних центрів,» *Системи підтримки прийняття рішень. Теорія і практика: Збірник доповідей 10-ої дистанційної науково-практичної конференції з міжнародною участю*, pp. 111-114, 8 червня 2015.

[45] О. Коваленко, «SWOT-аналіз технологій ситуаційного управління,» в *Обчислювальний інтелект (результати, проблеми, перспективи) : праці Міжнародної науково-практичної конференції*, Черкаси , Видавець Чебаненко Ю., 2015, p. 354.

[46] О. Коваленко, «Моделі організаційного забезпечення інформаційних технологій ситуаційних центрів,» в *Сучасні проблеми інформатики в управлінні, економіці та освіті : матеріали XIII-го міжнародного наукового семінару*, Київ, Національна академія управління, 2014, pp. 105-110.

[47] О. Е. Kovalenko, «The Architecture of Agent-Oriented Situation Management System,» *Proceedings of 3rd International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE-2013)*, pp. 621-627, 24 July 2014.

[48] О. Коваленко, «Інтеграція знань в агентно-орієнтовані системи підтримки прийняття рішень,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : тези доповідей VI Міжнародної наукової*

конференції, Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2014, р. 77.

[49] О. Коваленко, «Модель організації віртуальних інформаційних систем для розподіленої колективної обробки інформації», в *Обчислювальний інтелект (результати, проблеми, перспективи) : матеріали 2-ї Міжнародної науково-технічної конференції*, Черкаси, Маклаут, 2013, р. 367.

[50] О. Коваленко, «Архітектура сервісів організаційного забезпечення ситуаційного центру», *Системи підтримки прийняття рішень. Теорія і практика: Збірник доповідей IX-ої дистанційної науково-практичної конференції з міжнародною участю*, pp. 60-63, 3 Червня 2013.

[51] O. Kovalenko, «Agents Knowledge Models for Situation Management Systems», в *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2012)*, Sofia, Bulgaria, University of National and World Economy (UNWE), 11 January 2013, pp. 125-131.

[52] O. E. Kovalenko и D. G. Veleev, «Integration of Information Services in Agent-Oriented Situation Management System», в *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2012)*, Sofia, Bulgaria, University of National and World Economy (UNWE), 11 January 2013, pp. 132-135.

[53] О. Коваленко, «Організація агентно-орієнтованих систем ситуаційного управління», в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2012, pp. 63-65.

[54] O. E. Kovalenko, «Functional Approach in the Simulation Modeling of the Problem-Oriented Information Systems», в *Коммуникации, информационни технологии и статистика. Актуални проблеми на теорията и практиката*, София, Университетско издателство «Стопанство», 2011, pp. 278-281.

[55] А. Коваленко, «Построение систем поддержки принятия решений на принципах облачных вычислений», в *Proceedings of the International Conference on*

Application of Information and Communication Technology in Economy and Education (ICAICTEE-2011), Sofia, Bulgaria , University of National and World Economy, 2011, pp. 475-482.

[56] О. Є. Коваленко, «Елементи моделі цілеспрямованих систем орієнтованих на знання,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2011, pp. 172-174.

[57] О. Коваленко и А. Коваленко, «Аспекти функціонування інтелектуальних систем заснованих на знаннях,» *Обчислювальний інтелект (результати, проблеми, перспективи): Матеріали 1-ї Міжнародної науково-технічної конференції*, pp. 95-96, 10-13 травень 2011.

[58] О. Коваленко, «Організація комп'ютерного навчання з використанням моделей середовища навчання,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2010, pp. 160-163.

[59] О. Коваленко, «Оптимізація процесів електронного навчання на основі імітаційних моделей,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : збірник наук. праць за матеріалами четвертої міжнародної наукової конференції*, Кам'янець-Подільський, Кам'янець-Подільський національний університет імені Івана Огієнка, 2010, pp. 88-91.

[60] О. Коваленко, «Моделювання процедур експертного оцінювання у навчальних системах,» в *Комп'ютерна математика в науці, інженерії та освіті (CMSEE-2010) : матеріали Четвертої міжнародної науково-технічної конференції*, Полтава, Полтавський НТУ, 2010, p. 19.

[61] О. Коваленко, «Інтегральна модель засвоєння знань у навчальних системах,» в *Интегральные уравнения – 2009 = Integral equations – 2009 : сб. тезисов конф.*, Киев , ИПМЭ им. Г.Е.Пухова НАН Украины, 2009, pp. 91-92.

[62] А. Коваленко и Д. Велев, «Интеграция приложений электронного обучения в среде RIA/SOA,» в *Приложна информатика и статистика – съвременни подходи и методи : доклади от Международна научна конференция*, София, България, Университетско издателство «Стопанство», 2009, pp. 44-45.

[63] О. Коваленко, «Онтологічна модель системи прийняття рішень методом голосування,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2009, pp. 51-54.

[64] О. Коваленко, «Семантично-функціональний підхід до побудови багаторівневих систем електронного навчання,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2008, pp. 166-169.

[65] О. Коваленко, «Стандартизація навчальних курсів для глобального освітнянського простору,» в *Проблеми безперервного професійного розвитку лікарів і провізорів : збірник праць науково-методичної конференції з міжнародною участю*, Київ, Національна медична академія післядипломної освіти імені П.Л.Шупика, 2007, pp. 512-514.

[66] О. Коваленко и П. Вознюк, «Керування конфігурацією модульних систем електронного навчання,» в *Системи підтримки прийняття рішень. Теорія і практика*, Київ, Інститут проблем математичних машин і систем НАН України, 2007, pp. 137-140.

[67] О. Kovalenko, «Multilayer model of e-learning systems,» в *Бизнес Информатика : сборник доклади от международна научна конференция по повод 40-та годишнина на специалност "Информатика"*, София, Университет за национално и световно стопанство, 2007, p. 116 – 120.

[68] О. Коваленко, «Логічна модель стандартизованого навчального курсу для модельно-керованого процесу розробки,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : збірник наукових праць (за*

матеріалами міжнародної науково-методичної конференції), Кам'янець-Подільський, Кам'янець-Подільський державний університет, 2006, pp. 161-169.

[69] О. Коваленко, «Особливості розробки стандартизованих навчальних курсів для багаторазового використання,» в *Системи підтримки прийняття рішень. Теорія і практика : збірник доповідей науково-практичної конференції з міжнародною участю*, Київ, Інститут проблем математичних машин і систем НАН України, 2006, pp. 182-185.

[70] О. Kovalenko, «Evaluation of e-learning deployment scale,» в *Управленски, информационни и маркетингови аспекти на икономическото развитие на балканските страни : сборник доклади от международна научна конференция*, София, Университет за национално и световно стопанство, 2006, pp. 427-429.

[71] О. Kovalenko, «Efficiency of Corporate Electronic Learning,» в *Управленски, информационни и маркетингови аспекти на икономическото развитие на балканските страни : сборник доклади от международна научна конференция*, София, Университет за национално и световно стопанство, 2005, pp. 197-208.

[72] О. Коваленко, «UML-подання моделі екстремального програмування для проектування систем безпеки,» в *Проблеми впровадження інформаційних технологій в економіці : матеріали V міжнародної наук.-практ. конф.*, Ірпінь , Академія ДПС України, 2004, pp. 499-502.

[73] О. Є. Коваленко, «Моделювання технологій адаптивного проектування інформаційних систем,» в *Сучасні проблеми математичного моделювання, прогнозування та оптимізації : збірник наукових праць (за матеріалами Всеукраїнської науково-методичної конференції)*, Кам'янець-Подільський, Кам'янець-Подільський державний університет, 2004, pp. 111-115.

[74] О. Коваленко и В. Антоненко, «Технологія програмування та захист інформаційних систем,» в *Проблеми впровадження інформаційних технологій в економіці : матеріали IV міжнародної наук.-практ. конф.*, Ірпінь , Академія ДПС України, 2003, pp. 625-629.

[75] О. Коваленко, Комп'ютерний практикум з навчальної дисципліни «Системне інтегрування та адміністрування комп'ютерних мереж»: навчально-методичний посібник, Київ : Вид-во Інституту спеціального зв'язку та захисту інформації НТУУ «КПІ», 2012, р. 188.

[76] О. Коваленко, Захист інформації та програмних продуктів. Лабораторний практикум: Навч. посібник, Київ: Київський славістичний університет, 2007, р. 252.

[77] О. Коваленко, Дослідження операцій : навчальний посібник для спеціальностей «Менеджмент організацій», «Фінанси», «Облік і аудит», «Маркетинг», Київ: Київський славістичний університет – Міжрегіональний гуманітарний інститут, 2005, р. 55.

[78] О. Е. Kovalenko, «Models and means for service agents orchestration in situation management systems,» *Actual Problems of Economics*, т. 154, № 4, pp. 462-467, 2014.

[79] О. Є. Коваленко, «Координація взаємодії програмних агентів у системах ситуаційного управління,» *Актуальні проблеми економіки*, № 12, pp. 183-189, 2012.

[80] О. Є. Коваленко, «Мультиагентна модель колективного прийняття рішень,» *Актуальні проблеми економіки*, № 11, pp. 279-285, 2011.

[81] О. Коваленко, «Стійкість результатів обробки експертних оцінок у системах підтримки прийняття рішень,» *Актуальні проблеми економіки*, № 10, pp. 177-181, 2010.

[82] О. Коваленко, «Організація процедур прийняття рішень у ситуаційних центрах на основі голосування,» *Актуальні проблеми економіки*, № 6, pp. 289-293, 2009.

[83] О. Коваленко, «Архітектура модульних систем дистанційного навчання,» *Актуальні проблеми економіки*, № 12, pp. 172-176, 2007.

[84] О. Коваленко и В. Вишневський, «Організація системи дистанційного навчання на основі стандартних компонентів,» в *Актуальні питання фармацевтичної і медичної науки та практики*, Запоріжжя, Запорізький державний медичний університет, 2007, pp. 38-44.

[85] BKCASE, «Guide to the Systems Engineering Body of Knowledge (SEBoK),» 16 October 2018. [В Интернете]. Available: [https://www.sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_\(SEBoK\)](https://www.sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)). [Дата обращения: 21 March 2019].

[86] H. Goode и R. Machol, Systems Engineering: An Introduction to the Design of Large-Scale Systems, New York: McGraw-Hill, 1957, p. 551.

[87] J. N. Warfield, «Systems Engineering. , Report NO. , 1955, 31,» Pennsylvania State University. Navy Department, Ordnance Research Laboratory, 1955.

[88] L. v. Bertalanffy, General System Theory: Foundations, Development, Applications, New York, NY, USA: George Braziller, 1968, p. 289.

[89] A. W. Wymore, A Mathematical Theory of Systems Engineering: The Elements, Huntington, NY: Robert E. Krieger, 1967, p. 353.

[90] P. B. Checkland, Systems Thinking, Systems Practice, Hoboken, NJ: Wiley, 1981, p. 330.

[91] E. Yourdon, Managing the Systems Life Cycle: A Software Development Methodology Overview, New York: Yourdon Press, 1988, p. 158.

[92] E. Rechtin и M. Maier, The Art of Systems Architecting (Systems Engineering), Boca Raton, FL: CRC Press, 1997, p. 288.

[93] INCOSE, Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Fourth Edition ред., Hoboken, NJ: John Wiley & Sons Inc., 2015, p. 305.

[94] П. Анохин, Узловые вопросы теории функциональных систем, Москва: Наука, 1980, p. 203.

[95] М. Месарович и Я. Такахара, Общая теория систем: математические основы, Москва: Мир, 1978, p. 312.

[96] INCOSE, «A World in Motion: Systems Engineering Vision 2025,» International Council on Systems Engineering (INCOSE), 2014.

[97] S. G. Barton, «The Life Cycle and Buying Patterns,» в *Consumer Behavior*, т. II, L. H. Clark, Ред., New York, New York University Press, 1955, pp. 53-112.

[98] W. D. Wells и G. Gubar, «Life cycle concept in marketing research,» *Journal of Marketing Research*, т. 101, p. 355—363, 1966.

[99] G. Elliott, *Global Business Information Technology: an Integrated Systems Approach*, Boston: Addison-Wesley, 2004, p. 520.

[100] «ISO/IEC/IEEE 15288:2015 Systems and software engineering – System life cycle processes,» 2015. [В Інтернеті]. Available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=63711. [Дата обращения: 20 March 2019].

[101] V. Opanasenko и S. Kryvyi, «Synthesis of Neural-Like Networks on the Basis of Conversion of Cyclic Hamming Codes,» *Cybernetics and Systems Analysis*, т. 53, p. 627–635, 2017.

[102] A. Palagin, V. Opanasenko и S. Kryvyi, «Resource and Energy Optimization Oriented Development of FPGA-Based Adaptive Logical Networks for Classification Problem,» в *Green IT Engineering: Components, Networks and Systems Implementation*, т. 105, Berlin, Heidelberg, Springer International Publishing, 2017, pp. 195-218.

[103] M. Zgurovsky и A. Pavlov, «The four-level model of planning and decision making,» в *Combinatorial Optimization Problems in Planning and Decision Making*, Cham, Springer, 2019, pp. 347-406.

[104] О. Палагін и В. Опанасенко, «Формалізація процесу дослідного проектування,» *Проблеми інформатизації та управління*, т. 61, № 1, pp. 80-86, 2019.

[105] J. A. Estefan, «Survey of Model-Based Systems Engineering (MBSE) Methodologies,» 23 May 2008. [В Інтернеті]. Available: http://www.omgsysml.org/MBSE_Methodology_Survey_RevB.pdf. [Дата обращения: 21 March 2019].

[106] В. Самойлов, *Модельное конструирование компьютерных приложений*, Киев: Наукова думка, 2007, p. 200.

[107] Р. П. Абрамович, А. О. Бальва и В. Д. Самойлов, «Інтегрована технологія проектування комп'ютерних засобів сценарного типу підготовки фахівців для енергопідприємств,» *Електронне моделювання*, т. 40, № 2, pp. 27-42, 2018.

[108] А. А. Бальва, Е. А. Максименко и В. Д. Самойлов, «Структура и технология построения графической модели приложения сценарного типа,» *Збірник наукових праць Інституту проблем моделювання в енергетиці ім. ГЄ Пухова*, № 68, pp. 3-11, 2013.

[109] A. W. Wymore, *Model-Based Systems Engineering*, Boca Raton, FL: CRC Press, 1993, p. 710 .

[110] NIST, «Integration Definition for Function Modeling (IDEF0). Draft Federal Information Processing Standards, Publication 183,» National Institute of Standards and Technology (NIST), 21 December 1993. [В Інтернеті]. Available: <http://www.idef.com/wp-content/uploads/2016/02/idef0.pdf>. [Дата звернення: 21 March 2019].

[111] OMG, «OMG MDA. MDA - THE ARCHITECTURE OF CHOICE FOR A CHANGING WORLD,» OMG, 18 June 2014. [В Інтернеті]. Available: <https://www.omg.org/mda/>. [Дата звернення: 21 March 2019].

[112] OMG, «OMG Systems Modeling Language (OMG SysML™). Version 1.5,» Object Management Group (OMG), 1 May 2017. [В Інтернеті]. Available: <https://www.omg.org/spec/SysML/1.5/>. [Дата звернення: 21 March 2019].

[113] OMG, «Unified Profile for DoDAF and MODAF (UPDM). Version 2.1,» Object Management Group (OMG), August 2013. [В Інтернеті]. Available: <https://www.omg.org/spec/UPDM/>.

[114] О. Г. Додонов, О. В. Коваль, В. Р. Сенченко и В. В. Шпурик, «Автоматизована система формування сценарію аналітичної діяльності,» *Реєстрація, зберігання і обробка даних*, т. 21, № 1, pp. 11-22, 2019.

[115] А. Додонов, А. Никифоров и В. Путятин, «Тенденции и проблемы развития автоматизации управления военными силами,» *Математические машины и системы*, № 3, p. 3–16, 2019.

[116] А. Палагин и Ю. Яковлев, *Системная интеграция средств компьютерной техники*, Винница: УНІВЕРСУМ-Вінниця, 2005.

[117] ISO/IEC/IEEE, «ISO/IEC/IEEE 42010:2011 Systems and software engineering – Architecture description,» 2011. [В Інтернеті]. Available:

http://www.iso.org/iso/catalogue_detail.htm?csnumber=50508. [Дата обращения: 19 March 2019].

[118] S. Bernard, *An Introduction to Holistic Enterprise Architecture: Fourth Edition*, Bloomington, Indiana: AuthorHouse, 2020, p. 322.

[119] S. Kotusev, «Conceptual Model of Enterprise Architecture Management,» *International Journal of Cooperative Information Systems*, т. 26, № 3, pp. 1-36, 2017.

[120] S. Kotusev, «Enterprise Architecture and Enterprise Architecture Artifacts: Questioning the Old Concept in Light of New Findings,» *Journal of Information Technology*, т. 34, № 2, pp. 102-128, 2019.

[121] S. Kotusev, S. Kurnia и R. Dilnutt, «Roles of Different Artifacts in Enterprise Architecture Practice: An Exploratory Study,» в *International Conference on Information Systems (ICIS)*, India, Association for Information Systems (AIS), 2020, pp. 1-17.

[122] K. Sandkuhl, J. Stirna, A. Persson и M. Wißotzki, *Enterprise Modeling*, Cham: Springer, 2014.

[123] H. Koç и K. Sandkuhl, «Context modelling in capability management,» в *Capability Management in Digital Enterprises*, Cham, Springer, 2018, pp. 117-138.

[124] H. Koç, K. Sandkuhl, J. Stirna и J.-C. Kuhr, «Capability as a Service: Method and Tool Support for Context-Aware Business Services,» *International Journal of Information Systems in the Service Sector (IJISSS)*, т. 10, № 3, pp. 64-84, 2018.

[125] G. Koutsopoulos, M. Henkel и J. Stirna, «Improvements on Capability Modeling by Implementing Expert Knowledge About Organizational Change,» в *IFIP Working Conference on The Practice of Enterprise Modeling*, Cham, Springer, 2020, pp. 171-185.

[126] H. Schön, J. Zdravkovic, J. Stirna и S. Strahringer, «A role-based capability modeling approach for adaptive information systems,» в *IFIP Working Conference on The Practice of Enterprise Modeling*, Cham, Springer, 2019, pp. 68-82.

[127] D. W. Oliver, T. P. Kelliher и J. G. Keegan Jr., *Engineering Complex Systems with Models and Objects*, New York: McGraw Hill, 1997, p. 325.

[128] ISO, «ISO/PAS 19450:2015 Automation systems and integration – Object-Process Methodology,» 2015. [В Інтернеті]. Available: <https://www.iso.org/standard/62274.html>. [Дата звернення: 21 March 2019].

[129] W3C, «OWL 2. Web ontology language. Document Overview (Second Edition),» 11 December 2012. [В Інтернеті]. Available: <https://www.w3.org/TR/owl2-overview/>. [Дата звернення: 21 March 2019].

[130] J. N. Martin, «Overview of Emerging ISO Standards on Architecture,» 27 February 2018. [В Інтернеті]. Available: <http://gsaw.org/wp-content/uploads/2018/03/2018s05martin.pdf>. [Дата звернення: 21 March 2019].

[131] J.-L. Voirin, *Model-based System and Architecture Engineering with the Arcadia Method*, Elsevier, 2017, p. 388.

[132] M. Maier, «Architecting Principles for Systems-of-Systems,» *Systems Engineering*, т. 1, № 4, pp. 267-284, 1998.

[133] «ISO/IEC/IEEE 21839:2019. Systems and software engineering — System of systems (SoS) considerations in life cycle stages of a system,» 07 2019. [В Інтернеті]. Available: <https://www.iso.org/standard/71955.html>. [Дата звернення: 29 09 2019].

[134] «ISO/IEC/IEEE 21841:2019. Systems and software engineering — Taxonomy of systems of systems -,» 07 2019. [В Інтернеті]. Available: <https://www.iso.org/standard/71957.html>. [Дата звернення: 29 09 2019].

[135] S. C. a. P. J. M. Cook, «Towards designing innovative SoSE approaches for the Australian defence force,» *Proc. 9th Int. Conf. Syst. Syst. Eng. Socio-Technical Perspect*, № SoSE 2014, p. 295–300, 2014.

[136] G. Wells и A. Sage, «Engineering of a System of Systems,» в *System of Systems Engineering: Innovations for the 21st Century*, USA, Wiley, 2008, pp. 44-76.

[137] C. Azani, «A Multi-criteria Decision Model for Migrating Legacy System Architectures into Open Systems and Systems-of-Systems Architectures,» *A Publication of the Defense Acquisition University*, № October, pp. 320-341, 2009.

[138] А. Морозов, Л. Баран, В. Вишневецький, О. Коваленко и О. Малишев, «Розділ 6. Ситуаційний центр як засіб колективної роботи законотворців,» в

Електронний парламент України: досвід створення. Наукове видання, 3. з. р. С. Довгого, Ред., Київ, Логос, 2015, pp. 395-441.

[139] Д. Поспелов, Ситуационное управление: теория и практика, Москва: Наука, 1986, p. 288 .

[140] G. Jakobson, J. Buford и L. Lewis, «Situation Management: Basic Concepts and Approaches,» в *Information Fusion and Geographic Information Systems, LNG&C, vol. XIV / V. V. Popovich, M. Schrenk, and K. V. Korolenko, Eds. , Heidelberg, Springer, 2007, pp. 18-33.*

[141] M. Naderpour, J. Lu и G. Zhang, «An intelligent situation awareness support system for safety-critical environments,» *Decision Support Systems*, т. 59, pp. 325-340., 2014.

[142] Merriam-Webster Dictionary, «Situation,» 2018. [В Интернете]. Available: <http://www.merriam-webster.com/dictionary/situation>. [Дата обращения: 14 March 2019].

[143] J. Barwise и J. Perry, «The Situation Underground,» в *Stanford Working Papers in Semantics, vol. 1*, J. Barwise и I. Sag, Ред., Stanford Cognitive Science Group , Section D, 1980, p. 1–55.

[144] J. Barwise и J. Perry, *Situations and Attitudes*, Cambridge, MA: MIT Press, 1983.

[145] В. С. Юкаева, Управленческие решения: учеб. пособие, 3-е изд, 3-е ред., Москва: Изд. Торг. корпорация «Дашков и Ко», 2009, p. 324.

[146] Е. Падучева, Семантические исследования: Семантика времени и вида в русском языке; Семантика нарратива. 2-е изд., испр. и доп., М.: Языки славянской культуры, 2010. – 480 с.

[147] А. А. Морозов, «Ситуационные центры. Понятия и определения./А.А.Морозов.,» *Математичні машини і системи*, № 1, pp. 48-54, 2016.

[148] В. Глушков, Введение в АСУ - 2-е изд., испр. и доп., Киев: Техника, 1972, p. 320.

[149] R. Axelrod, *The Structure of Decision: Cognitive Maps of Political Elites*, Princeton: University Press, 1976.

[150] U. Neisser, *Cognition and Reality: Principles and implications of cognitive psychology*, San Fransisco: W H Freeman and Company, 1976, p. 230 .

[151] P. M. Salmon, N. A. Stanton, G. H. Walker, C. Baber, D. P. Jenkins, R. McMaster и M. S. Young, «What really is going on? Review of situation awareness models for individuals and teams,» *Theoretical Issues in Ergonomics Science*, т. 9, № 4, pp. 297-323, 2008.

[152] K. Smith и P. Hancock, «Situation awareness is adaptive, externally directed consciousness,» *Human Factors*, № 37, p. 137–148, 1995.

[153] M. Endsley, «Toward a Theory of Situation Awareness in Dynamic Systems,» *The Journal of the Human Factors and Ergonomics Society*, т. 37, № 1, pp. 32-64, 1995.

[154] J. Lundberg, «Situation awareness systems, states and processes: a holistic framework,» *Theoretical Issues in Ergonomics Science*, т. 16, № 5, pp. 447-473, 2015.

[155] S. Holub и S. Kunytska, «The concept of multi-agent intellectual monitoring systems,» в *Projekt interdyscyplinarny projektem XXI wieku - Tom 2. Processing, transmission and security of information*, Bielsko-Biale, Wydawnictwo naukowe Akademii Techniczno-Humanistycznej w Bielsku-Bialej, 2019, pp. 183-188.

[156] S. Kunytska и S. Holub, «Multi-agent Monitoring Information Systems,» в *Mathematical Modeling and Simulation of Systems. MODS 2019. Advances in Intelligent Systems and Computing*, т. 1019, A. Palagin, A. Anisimov, A. Morozov и S. Shkarlet, Ред., Cham, Springer, 2019, pp. 164-171.

[157] V. I. Gorodetski, «Self-Organization and Multiagent Systems: II. Applications and the Development Technology,» *Journal of Computer and Systems Sciences International*, т. 51, № 3, p. 391—409, 2012.

[158] В. И. Городецкий, О. Л. Бухвалов, П. О. Скобелев и И. Майоров, «Современное состояние и перспективы индустриальных применений многоагентных систем,» *Управление большими системами*, № 66, pp. 94-157, 2017.

[159] E. Hutchins, *Cognition in the Wild*, MIT Press, 1995, p. 381.

[160] N. A. Stanton, M. Salmon, G. H. Walker и P. A. Hancock, «State-of-Science: Situation Awareness in individuals, teams and systems,» *Ergonomics*, т. 60, № 4, pp. 247-258, 2017.

[161] C. S. Peirce, «Notes on Ampliative Reasoning,» в *The Collected Papers of Charles S. Peirce, 8 vols., ed. by Hartshorne, C, Weiss, P. and Burks, A. W., 4 ред., т. 2, C. Hartshorne, P. Weiss и A. W. Burks, Ред., Cambridge, Belknap Press of Harvard University Press, 1931–1966.*

[162] В. А. Светлов, «Методологическая концепция научного знания Чарлза Пирса: единство абдукции, дедукции и индукции,» *Логико-философские штудии*, № 5, pp. 165-188, 2008.

[163] В. Г. Шерстюк, «Абдуктивная модель вывода по прецедентам в динамической сценарно-прецедентной системе,» *Проблеми інформаційних технологій*, № 2, pp. 55-61, 2014.

[164] В. Вагин и К. Хотимчук, «Абдукция в задачах планирования работ в сложных объектах,» в *Искусственный интеллект и принятие решений*, т. 1, Москва, Ленанд, 2011, pp. 3-13.

[165] A. Kakas, A. Michael и C. Mourlas, «Acip: Abductive constraint logic programming,» *Journal of Logic Programming*, т. 44, № 1-3, pp. 129-177, 2000.

[166] D. McSherry, «Explanation in recommender systems,» *Artificial Intelligence Review*, т. 24, № 2, pp. 179-197, 2005.

[167] Г. И. Рузавин, «Абдукция как метод поиска и обоснования объяснительных гипотез,» в *Теория и практика аргументации*, Москва, Институт философии РАН, 2001, pp. 28-48.

[168] G. Shafer, *Mathematical Theory of Evidence*, Princeton University Press, 1976.

[169] J. Yang и D. Xu, «Evidential Reasoning Rule for Evidence Combination,» *Artificial Intelligence*, т. 205, p. 1–29, 2013.

[170] C. Siow, J. Yang и B. Dale, «A new modelling framework for organisational self-assessment: development and application,» *Quality Management Journal*, т. 8, № 4, p. 34–47, 2001.

[171] J. Ma, H. Ning, R. Huang, H. Liu, L. T. Yang, J. Chen и G. Min, «Cybermatics: A holistic field for systematic study of cyber-enabled new worlds,» *IEEE Access*, т. 3, pp. 2270-2280, 5 November 2015.

[172] National Science Foundation [US], «Cyber-Human Systems (CHS),» 9 July 2018. [В Интернете]. Available: https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=504958. [Дата обращения: 20 March 2019].

[173] А. Ф. Верлань, М. Ф. Ус, А. В. Пискун и В. А. Федорчук, Когнитивное управление в интеллектуальных обучающих системах, Черкассы: РИО Черкасского Института управления, 2002, р. 104.

[174] А. Верлань, И. Чмырь и Ю. Фуртат, «Построение формальной модели вопросно-ответного взаимодействия пользователя с автоматизированной системой,» *Электронное моделирование*, т. 36, № 1, pp. 41-48, 2014 .

[175] Ю. Яковлев, «О виртуализации рабочих мест и их компонентов при построении компьютерных систем и сетей,» *Управляющие системы и машины*, № 3, pp. 44-56, 2017.

[176] A. I. Petrenko, «Service-oriented computing in a cloud computing environment,» *Computer Science and Applications*, т. 1, № 6, pp. 349-358, 2014.

[177] M. Roco, «Coherence and Divergence of Megatrends in Science and Engineering,» *Journal of Nanoparticle Research*, т. 4, № 1-2, pp. 9-19, 2002.

[178] W. S. Bainbridge и M. C. Roco, *Handbook of Science and Technology Convergence*, Springer, 2016, p. 1154.

[179] M. C. Roco и W. S. Bainbridge, «The new world of discovery, invention, and innovation: convergence of knowledge, technology, and society,» *Journal of Nanoparticle Research*, т. 15, № 9, pp. 1-17, 2013.

[180] M. Roco, «Convergence-Divergence Process,» в *Handbook of Science and Technology Convergence*, W. Bainbridge и M. Roco, Ред., Cham, Springer, 2016, pp. 79-93.

[181] SolarWinds MSP, «Converged IT Infrastructure Overview and Benefits,» 7 January 2020. [В Интернете]. Available:

<https://www.solarwindsmisp.com/blog/converged-it-infrastructure>. [Дата обращения: 5 September 2020].

[182] M. Burgess, *In Search of Certainty: The science of our information infrastructure*, Sebastopol, California: O'Reilly Media, 2015, p. 472.

[183] Ю. М. Лисецкий, *Иновационные информационные технологии. Опыт интеграции и внедрения*, Киев: ЛАТ&К, 2020, p. 276.

[184] O. Rolik, S. Telenyk и E. Zharikov, «Management of services of a hyperconverged infrastructure using the coordinator,» в *International Conference on Computer Science, Engineering and Education Applications*, Cham, Springer, 2018, pp. 456-467.

[185] Ю. Лисецкий и Ю. Саблий, «Гиперконвергентная технология с открытым кодом,» *Математические машины и системы*, № 1, p. 49–55, 2019.

[186] А. Коваленко, *Специализированные устройства адаптивной цифровой обработки сигналов многоэлементных первичных преобразователей*, Киев: ФОЛ Института электродинамики АН УССР, 1990, p. 16.

[187] А. Й. Наконечний, Р. І. Стахів и Р. А. Наконечний, *Обробка сигналів : навч. посіб.*, Львів: Нац. ун-т "Львівська політехніка", 2017, p. 2017.

[188] Р. Акоф и Ф. Эмери, *О целеустремленных системах*, Москва: «Советское радио», 1974, p. 272 .

[189] В. Гладун, *Партнерство с компьютером: Человеко-машинные целеустремленные системы*, Киев : «Port-Royal», 2000, p. 128.

[190] U. Grenander, *General Pattern Theory: A Mathematical Study of Regular Structures*, New York, NY: Oxford University Press Inc., 1993.

[191] Г. Фреге, *Избранные работы*, Москва : Изд-во: «ДИК», 1997.

[192] З. Г.Н., «Метаинформатика, искусственный интеллект и основания языка науки,» в *Интеллектуальные системы управления*, а. Р. С. Васильев, Ред., Москва, Машиностроение, 2010, pp. 7-16.

[193] Н. Д. Арутюнова, *Теория метафоры: Сборник статей: Пер. с англ., фр., нем., исп., польск. яз.*, Н. Д. А. и М. А. Журинская, Ред., Москва: Прогресс, 1990, p. 512.

[194] Д. Лакофф и М. Джонсон, Метафоры, которыми мы живём, Москва: Едиториал УРС, 2004.

[195] Merriam-Webster dictionary, «Definition of information,» [В Интернете]. Available: <https://www.merriam-webster.com/dictionary/information>. [Дата обращения: 12 January 2018].

[196] L. Floridi, «Semantic Conceptions of Information,» 2015. [В Интернете]. Available: <https://plato.stanford.edu/entries/information-semantic>. [Дата обращения: 12 January 2018].

[197] «ISO/IEC 2382-2015(en), Information technology — Vocabulary,» 2015. [В Интернете]. Available: <https://www.iso.org/obp/ui/#iso:std:63598:en>. [Дата обращения: 12 January 2018].

[198] R. L. Ackoff, «From data to wisdom,» *Journal of Applied Systems Analysis*, т. 16, pp. 3-9, 1989.

[199] P. Adriaans, «Information,» 2012. [В Интернете]. Available: <https://plato.stanford.edu/archives/fall2013/entries/information>. [Дата обращения: 12 January 2018].

[200] «Ontology Lookup Service,» [В Интернете]. Available: <https://www.ebi.ac.uk/ols/search?q=information+ontology&groupField=iri&exact=on&start=0>. [Дата обращения: 12 January 2018].

[201] «Information Artifact Ontology,» 2017. [В Интернете]. Available: <https://github.com/information-artifact-ontology/IAO>. [Дата обращения: 12 January 2018].

[202] Wikipedia, «Decision cycle,» 2017. [В Интернете]. Available: https://en.wikipedia.org/wiki/Decision_cycle. [Дата обращения: 07 02 2019].

[203] О. Коваленко, «Застосування онтологій в системах ситуаційного управління,» в *Сучасні проблеми інформатики в управлінні, економіці, освіті та подоланні наслідків Чорнобильської катастрофи: матеріали XV Міжнародного наукового семінару*, Київ-Світязь, 4-8 липня 2016.

[204] R. Priemer, *Introductory Signal Processing*, World Scientific, 1990, p. 752.

[205] C. S. Peirce и V. Welby, *Semiotics and Significs*, C. H. (ed.), Ред., Bloomington IN: Indiana University Press, 1977, p. 201.

[206] ISO/IEC, «ISO/IEC 11404:2007(en) Information technology — General-Purpose Datatypes (GPD),» 2007. [В Интернете]. Available: <https://www.iso.org/standard/39479.html>. [Дата обращения: 07 02 2019].

[207] P. Panov, L. N. Soldatova и S. Džeroski, «Generic ontology of datatypes,» *Information Sciences*, т. 329, pp. 900-920, 2016.

[208] «Knowledge,» 2018. [В Интернете]. Available: <https://en.oxforddictionaries.com/definition/knowledge>. [Дата обращения: 12 January 2018].

[209] «Definition of 'knowledge',» 2018. [В Интернете]. Available: <https://www.collinsdictionary.com/dictionary/english/knowledge>. [Дата обращения: 12 January 2018].

[210] F. v. Harmelen, V. Lifschitz и B. P. (Eds.), Ред., *Handbook of Knowledge Representation*, Elsevier B.V., 2008.

[211] D. Rooney, B. McKenna и P. Liesch, *Wisdom and Management in the Knowledge Economy*, NewYork, London: Routledge, Taylor&Francis Group, 2010, p. 259 .

[212] S. Ryan, «Wisdom,» 8 January 2007. [В Интернете]. Available: <https://plato.stanford.edu/archives/win2014/entries/wisdom>. [Дата обращения: 20 March 2019].

[213] Д. Нейман и М. О., *Теория игр и экономическое поведение*, Москва: Наука, 1970, p. 707.

[214] L. Savage, *The Foundations of Statistics*, New York: Dover Publications, 1972, p. 310.

[215] D. Kahneman и A. Tversky, «Prospect Theory: An Analysis of Decision under Risk,» *Econometrica*, т. 47, № 2, p. 263–291, 1979.

[216] A. Tversky и D. Kahneman, «Advances in prospect theory: Cumulative representation of uncertainty,» *Journal of Risk and Uncertainty*, т. 5, № 4, p. 297–323, 1992.

[217] E. Blasch, É. Bossé и D. A. Lambert, High-level information fusion management and systems design, Artech House, 2012, p. 370.

[218] E. Blasch, «One Decade of the Data Fusion Information Group (DFIG) Model,» в *Next-Generation Analyst III*, International Society for Optics and Photonics, 2015, p. 94990L.

[219] В. п. р. Смирнова, Ред., Семантика модальных и интенциональных логик, Москва: Прогресс, 1981, p. 424.

[220] D. Fensel и e. al., «On-to-knowledge: Ontology-based tools for knowledge management,» *Proceedings of the eBusiness and eWork*, pp. 18-20, 2000.

[221] O. Petrenko и A. Petrenko, «A Model-driven Ontology Approach for Developing Service System Applications,» *Journal of Computer Science Applications and Information Technology*, т. 4, № 2, pp. 1-7, 2017.

[222] О. Палагін и М. Петренко, «Про деякі особливості побудови онтологічних моделей предметних областей,» *Control systems & computers*, № 3, pp. 23-37, 2019.

[223] А. Палагин, С. Крытый и Н. Петренко, Онтологические методы и средства обработки предметных знаний, Луганск: Изд-во ВНУ им. В. Даля, 2012, p. 324.

[224] А. А. Петренко и А. И. Петренко, «Семантическое модельно-управляемое моделирование архитектуры системы сервисов на основе доменных онтологий,» *Електронне моделювання*, т. 38, № 5, pp. 43-60, 2016.

[225] Э. Мендельсон, Введение в математическую логику, Москва: Наука, 1971, p. 320.

[226] V. K. Singh и R. Jain, «Overall Framework for Situation Recognition: Overview,» в *Situation Recognition Using EventShop*, Cham, Springer International Publishing, 2016, pp. 39-46.

[227] K. Devlin, «Situation theory and situation semantics,» в *Handbook of the History of Logic*, т. 7, Stanford, 2006, pp. 601-664.

[228] O. Rolik, V. Kolesnik и D. Halushko, «Decomposition-Compensation Method for IT Service Management,» в *Congress on Information Technology, Computational and Experimental Physics*, Cham, Springer, 2016, pp. 89-107.

[229] J. Zachman, «A Framework for Information Systems Architecture,» *IBM Systems Journal*, т. 26, № 3, pp. 276-292, 1987.

[230] «Federal Enterprise Architecture Framework Version 2,» 29 January 2013. [В Интернете]. Available: https://obamawhitehouse.archives.gov/sites/default/files/omb/assets/egov_docs/fea_v2.pdf. [Дата обращения: 19 March 2019].

[231] G. Kreizman, C. H. Baum и E. A. Fraga, «Gartner Enterprise Architecture: A Home for E-Government,» Gartner, 29 July 2003. [В Интернете]. Available: <https://www.gartner.com/doc/405561/gartner-enterprise-architecture-home-egovernment>. [Дата обращения: 19 March 2019].

[232] US Department of Defense, «Department of Defense Architecture Framework (DoDAF), Version 2.02,» 03 March 2011. [В Интернете]. Available: <https://dodcio.defense.gov/Library/DoD-Architecture-Framework/>. [Дата обращения: 19 March 2019].

[233] OMG, «Unified Architecture Framework (UAF),» OMG, April 2020. [В Интернете]. Available: <https://www.omg.org/spec/UAF/>. [Дата обращения: 9 September 2020].

[234] «Survey of Architecture Frameworks,» 14 10 2020. [В Интернете]. Available: <http://www.iso-architecture.org/42010/afs/frameworks-table.html>. [Дата обращения: 10 11 2020].

[235] «ISO/IEC 27000:2018 Information technology - Security techniques - Information security management systems - Overview and vocabulary,» ISO/IEC, Geneva, 2018.

[236] O. Kovalenko и T. Kovalenko, «Knowledge Model and Ontology for Security Services,» *2018 IEEE First International Conference on System Analysis and Intelligent Computing (SAIC) – Proceedings*, pp. 1-4, Article number 8516875, 01 November 2018.

[237] S. Kripke, «Semantical Considerations on Modal Logic,» *Acta Philosophica Fennica*, № 16, pp. 83-94, 1963.

[238] V. Gorodetsky, V. Samoilov и D. Trotskii, «The Reference Ontology of Collective Behavior of Autonomous Agents and Its Extensions,» *Journal of Computer and Systems Sciences International*, т. 54, № 5, p. 765—782, 2015.

[239] S. Telenyk, G. Nowakowski, E. Zharikov, Y. Vovk и O. Tokmenko, «Action Planning Logic in Intelligent IT Systems Using the Example of Bots,» в *2020 21st International Conference on Research and Education in Mechatronics (REM)*, Cracow, Poland, IEEE, 2021, pp. 1-11.

[240] С. Субботін, А. Олійник и О. Олійник, Неітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей : монографія, С. Субботін, Ред., Запоріжжя: Запорізький національний технічний університет, 2009, р. 375.

[241] С. О. Субботін, Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень, Запоріжжя: Запорізький національний технічний університет, 2008, р. 341.

[242] В. Томашевский и Д. Диденко, «Агентная архитектура распределенной дискретно-событийной системы имитационного моделирования OpenGPSS,» *Системні дослідження та інформаційні технології*, № 4, pp. 123-132, 2006.

[243] H. S. Nwana, «Software Agents: An Overview,» *The Knowledge Engineering Review*, т. 11, № 3, p. 205–244, 1996.

[244] A. S. Rao и M. P. Georgeff, «BDI Agents: From Theory to Practice,» *Proceedings of the First International Conference on Multiagent Systems (ICMAS)*, т. 1995, pp. 312-319, 1995.

[245] M. Bratman, *Intention, plans, and practical reason*, т. 10, Cambridge, MA: Harvard University Press, 1987, p. 208.

[246] I. Ravenscroft, «Folk Psychology as a Theory,» 2016. [В Інтернеті]. Available: <https://plato.stanford.edu/entries/folkpsych-theory>. [Дата звернення: 30 April 2020].

[247] H. A. Simon, «A Behavioral Model of Rational Choice,» *Quarterly Journal of Economics*, т. 69, № 1, p. 99–118, 1955.

[248] H. Dixon, «Some thoughts on economic theory and artificial intelligence, chapter 7,» в *Surfing Economics. Artificial Intelligence and Economic Theory*, 2018.

[249] M. Georgeff и A. Rao, «The Semantics of Intention Maintenance for Rational Agents, IJCAI,» 1995. [В Интернете]. Available: <https://www.ijcai.org/Proceedings/95-1/Papers/092.pdf>. [Дата обращения: 30 April 2020].

[250] M. Wooldridge, *Reasoning about rational agents*, Cambridge, Mass.: MIT Press, 2000, p. 227.

[251] Ч. Моррис, «Основания теории знаков,» в *Семиотика. Сборник переводов*, Ю. С. Степанов, Ред., Москва, Радуга, 1982, pp. 37-89.

[252] A. Tarski, «The Semantic Conception of Troth and the Foundations of Semantics,» *Philosophy and Phenomenologicai Reseach*, т. 4, № 3, p. 341–375, 1944.

[253] I. Nonaka и H. Takeuchi, *The Knowledge-creating Company: How Japanese Companies Create the Dynamics of Innovation*, Oxford: Oxford University Press, 1995, p. 284.

[254] I. Nonaka, R. Toyama и P. Byosièrè, «A theory of organizational knowledge creation: Understanding the dynamic process of creating knowledge,» в *Handbook of organizational learning and knowledge*, M. Dierkes, A. B. Antal, J. Child и I. Nonaka, Ред., New York, Oxford University Press, 2001, pp. 491-516.

[255] H. Mintzberg, *Structure in fives: Designing effective organizations*, Englewood Cliffs, N.J. : Prentice-Hall, 1983, p. 312.

[256] S. A. Kripke, «Semantical analysis of modal logic i normal modal propositional calculi,» *Mathematical Logic Quarterly*, т. 9, № 5-6, pp. 67-96, 1963.

[257] E. Emerson и J. Halpern, «Decision procedures and expressiveness in the temporal logic of branching time,» *Journal of computer and system sciences*, т. 30, № 1, pp. 1-24, 1985.

[258] М. С. Нікітченко и С. С. Шкільняк, «Композиційно-номінативні модальні логіки,» *Проблеми програмування*, № 1-2, pp. 27-33, 2002.

[259] F. Baader, I. Horrocks и U. Sattler, «Chapter 3 Description Logics,» в *Handbook of Knowledge Representation*, F. v. Harmelen, V. Lifschitz и B. Porter, Ред., Elsevier, 2008, pp. 135-180.

[260] OASIS, «Web Services Business Process Execution Language Version 2.0. OASIS Standard,» 11 April 2007. [В Интернете]. Available: <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>.. [Дата обращения: 18 03 2019].

[261] YAWL Foundation, «YAWL: Yet Another Workflow Language,» [В Интернете]. Available: <http://www.yawlfoundation.org>. [Дата обращения: 18 March 2019].

[262] F. Montesi, «Chor – choreography programming language,» [В Интернете]. Available: <http://www.chor-lang.org>. [Дата обращения: 18 March 2019].

[263] G. Decker, O. Kopp, F. Leymann и M. Weske, «BPEL4Chor: Extending BPEL for Modeling Choreographies,» *IEEE International Conference on Web Services (ICWS 2007)*, pp. 296-303, 9-13 July 2007.

[264] W3C, «Web Services Choreography Description Language Version 1.0. W3C Candidate Recommendation,» 9 November 2005. [В Интернете]. Available: <https://www.w3.org/TR/2005/CR-ws-cdl-10-20051109>. [Дата обращения: 18 March 2019].

[265] Workflow Patterns Initiative, «Workflow Patterns,» 2010. [В Интернете]. Available: <http://www.workflowpatterns.com/patterns/index.php>. [Дата обращения: 18 March 2019].

[266] А. А. Морозов и В. В. Вишневский, «Регламентные процедуры ситуационного зала.–Київ (pp. 5-7).,» *Системи підтримки прийняття рішень. Теорія і практика: Зб. доп. наук.-практ. конф. з міжнар. участю*, pp. 5-7, 9 Червень 2007.

[267] В. В. Вишневский и С. М. Симонов, «Организационные процедуры ситуационного центра,» *Математические машины и системы*, № 4, pp. 62-67, 2010.

[268] F. Daniel и B. Pernici, «Web Service Orchestration and Choreography: Enabling Business Processes on the Web, Volume 2, November 2007,» в *E-Business Models, Services, and Communications - Advances in E-Business Research Series*, т. 2, I.

Lee, Ред., Hershey-New York, Information Science Reference (IGI Global), 2008, pp. 251-274.

[269] «ISO/IEC 10746-2:2009 – Information technology — Open distributed processing — Reference model: Foundations,» 2009. [В Интернете]. Available: <https://www.iso.org/ru/standard/55723.html>. [Дата обращения: 11 May 2020].

[270] «ISO/IEC 24765:2017 – Systems and Software Engineering – Vocabulary,» 2017. [В Интернете]. Available: <https://www.iso.org/standard/71952.html>. [Дата обращения: 11 May 2020].

[271] «New European Interoperability Framework (EIF). ISA² - Interoperability solutions for public administrations, businesses and citizens,» 2018. [В Интернете]. Available: https://ec.europa.eu/isa2/sites/isa/files/eif_brochure_final.pdf. [Дата обращения: 11 May 2020].

[272] A. Koulou, M. Zemzami, N. El Hani, A. Elmir и N. Hmina, «Optimization in collaborative information systems for an enhanced interoperability network,» *Int. J. Simul. Multidisci. Des. Optim.*, № 11, p. 2, 2020.

[273] G. Leal, W. Guedria и P. H., «A semi-automated system for interoperability assessment: an ontology- based approach,» *Enterprise Information Systems*, т. 14 , № 3, pp. 308-333, 2020.

[274] G. da Silva, S. Leal, W. Guédria и H. Panetto, «Interoperability Assessment: A Systematic Literature Review,» *Computers in Industry*, т. 106, № April 2019, pp. 111-132, 2019.

[275] W. Guédria, Y. Naudet и D. Chen, «Maturity Model for Enterprise Interoperability,» *Enterprise Information Systems*, т. 9, № 1, p. 1–28, 2015.

[276] «European Interoperability Reference Architecture (EIRA). ISA² - Interoperability solutions for public administrations, businesses and citizens,» 2018. [В Интернете]. Available: <https://joinup.ec.europa.eu/asset/eia/description>. [Дата обращения: 11 May 2020].

[277] Microsoft, «Chapter 1: What is Software Architecture? In Microsoft Application Architecture Guide, 2nd Edition,» Microsoft, 01 Jan 2010. [В Интернете].

Available: [https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658098\(v=pandp.10\)](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658098(v=pandp.10)). [Дата обращения: 28 May 2019].

[278] «What is Software Architecture?», 14 January 2010. [В Интернете]. Available: [https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658098\(v=pandp.10\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ee658098(v=pandp.10)?redirectedfrom=MSDN). [Дата обращения: 15 March 2019].

[279] A. R. da Silva, «Model-Driven Engineering: A Survey Supported by a Unified Conceptual Model», *Computer Languages, Systems & Structures*, № 43, p. 139–155, 2015.

[280] D. Haywood, «An Introduction to Domain Driven Design», *Methods and Tools*, т. 17, № 4, pp. 18-37, 2009.

[281] E. Evans, «Domain Driven Design, Definitions and Pattern Summaries», March 2015. [В Интернете]. Available: https://domainlanguage.com/wp-content/uploads/2016/05/DDD_Reference_2015-03.pdf. [Дата обращения: 4 April 2019].

[282] J. Castro, M. Kolp и J. Mylopoulos, «A requirements-driven development methodology», *International Conference on Advanced Information Systems Engineering*, pp. 108-123, June 2001.

[283] S. W. Ambler, «Feature Driven Development (FDD) and Agile Modeling», 2018. [В Интернете]. Available: <http://www.agilemodeling.com/essays/fdd.htm>. [Дата обращения: 4 April 2019].

[284] Е. М. Лаврищева, «Теория объектно-компонентного моделирования программных систем», 2016. [В Интернете]. Available: http://www.ispras.ru/preprints/docs/prep_29_2015.pdf. [Дата обращения: 4 April 2019].

[285] S. Fenz и A. Ekelhart, «Formalizing information security knowledge», *Proceedings of the 4th international Symposium on information, Computer, and Communications Security*, pp. 183-194, March 2009.

[286] S. Fenz, S. Plieschnegger и H. Hobel, «Mapping information security standard ISO 27002 to an ontological structure», *Information & Computer Security*, т. 24, № 5, pp. 452-473, 2016.

[287] A. Stasiuk, R. V. Hryshchuk и L. L. Goncharova, «Mathematical differential models and methods for assessing the cybersecurity of computer networks intelligent

control of technological processes of railway power supply,» *New Means Cybernetics, Informatics, Computers Engineering and Systems Analysis*, т. 54, № 4, pp. 671-682, 2018.

[288] A. I. Stasiuk и L. L. Goncharova, «Mathematical Models and Methods of Formation of Intelligent Computer Networks for Control of Power Supply and Optimization of Power Consumption of Railways,» *Journal of Automation and Information Sciences*, т. 50, № 8, pp. 50-65, 2018.

[289] A. Verlan, M. Karimov, S. Ganiev и K. Tashev, «Organization of an integrated system of information security in information and communication systems,» в *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2012)*, Sofia, Bulgaria, University of National and World Economy (UNWE), 2013, pp. 39-46.

[290] «ISO/IEC 27005:2011 Information technology - Security techniques - Information security risk management,» 2011. [В Интернете]. Available: <https://www.iso.org/standard/56742.html>. [Дата обращения: 03 March 2018].

[291] S. Taubenberger и J. Jürjens, «IT Security Risk Analysis Based on Business Process Models Enhanced with Security Requirements,» в *Modeling Security Workshop*, Toulouse, France, September, 2008.

[292] B. D. Bunday, *Basic Optimization Methods*, London: Edward Arnold Publishers, 1984.

[293] В. Тоценко, *Методы и системы поддержки принятия решений. Алгоритмический аспект*, Киев: Наукова Думка, 2002, р. 381.

[294] R. Reiter, *Knowledge in action: logical foundations for specifying and implementing dynamical systems*, т. 16, Cambridge: MIT Press, 2001.

[295] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld и D. Wilkins, «PDDL---The Planning Domain Definition Language. Technical Report CVC TR98003/DCS TR1165.,» Yale Center for Computational Vision and Control, New Haven, CT, 1998.

[296] D. L. Kovacs, «A Multi-Agent Extension of PDDL3.1,» в *Proceedings of the 3rd Workshop on the International Planning Competition (IPC). 22nd International*

Conference on Automated Planning and Scheduling (ICAPS-2012), Atibaia, San Paulo, 2012, p. 19–27.

[297] L. Martyushev и V. Seleznev, «Maximum entropy production principle in physics, chemistry and biology,» *Physics Reports*, т. 426, № 1, pp. 1-45, 2006.

[298] M. Zgurovsky и P. Kasyanov, «Indirect Lyapunov Method for Autonomous Dynamical Systems,» в *Qualitative and Quantitative Analysis of Nonlinear Systems*, Cham, Springer, 2018, pp. 211-237.

[299] P. M. Allen, «Evolution, Population Dynamics and Stability,» *Proceedings of the National Academy of Sciences of the USA*, March, т. 73, № 3, pp. 665-668, 1976.

[300] Л. Понтрягин, В. Болтянский, Р. Гамкредидзе и М. Е.Ф., Математическая теория оптимальных процессов, Москва: Наука, 1961, p. 393.

[301] А. Морозов, В. Косолапов и О. Отрішко, «Засоби інформаційного сервісу об'єктів інтелектуальної власності,» *Математичні машини і системи*, № 3, p. 136 – 147, 2002.

[302] M. Ebberts, C. Bischoff, D. Buls, D. Oliver, E. Steenvoorden и S. Thomschke, IBM Lotus Learning Management System Handbook, International Technical Support Organization, 2003, p. 466.

[303] Т. О. Kovalenko и А. V. Miroshnychenko, «Architectural solutions for situational management,» *5th International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2015)*, pp. 346-355, 2016.

[304] «A Guide to the Project Management Body of Knowledge (PMBOK Guide). Project Management Institute (sixth edition),» 2017. [В Интернете]. Available: <https://www.pmi.org/pmbok-guide-standards/foundational/pmbok/sixth-edition>. [Дата обращения: 8 January 2018].

[305] «ISO/IEC/IEEE 16326-2009 – Systems and Software Engineering-Life Cycle Processes-Project Management,» December 2009. [В Интернете]. Available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=41977. [Дата обращения: 12 May 2018].

[306] «ISO 21500:2012 Guidance on project management,» 2012. [В Интернете]. Available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=50003. [Дата обращения: 12 May 2018].

[307] А. Ф. Верлань, И. А. Чмырь, С. Д. Кузниченко и Л. Б. Коваленко, Императивное программирование и объектно-ориентированное моделирование: Java, UML, OCL [Текст] : учеб. пособие для студентов высш. учеб. заведений, Одесса : Экология, 2013, р. 430.

[308] О. К. Закусило, А. Ф. Верлань и И. А. Чмырь, Моделирование диалоговых процессов : монография, Київ : Київський університет, 2014, р. 255.

[309] Э. Алгазинов и А. Сирота, Анализ и компьютерное моделирование информационных процессов и систем, Москва: Диалог-Мифи, 2009, р. 416.

[310] D. Kendall, «Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain,» *The Annals of Mathematical Statistics*, т. 3, № 24, р. 338–254, 1953.

[311] Х. А. Таха, «Системы массового обслуживания,» в *Введение в исследование операций = Operations Research: An Introduction*, 7-е изд ред., Москва, Вильямс, 2007, р. 629–697.

[312] В. Гулиус, «Интеллектуальная модель системы массового обслуживания с очередью типа D/D/1 в среде SimEvents (MATLab/SimuLink),» [В Интернете]. Available: http://model.exponenta.ru/cl_gva_02.html. [Дата обращения: 11 Лютий 2019].

ДОДАТКИ

Додаток А

Акти впровадження



АКТ

**про використання результатів дисертаційної роботи Коваленка
Олексія Єпифановича «Моделі і методи побудови
конвергентних систем ситуаційного управління», яка подається
на здобуття наукового ступеня доктора технічних наук**

Даний акт складено про те, що результати дисертаційної роботи старшого наукового співробітника відділу №235 Теорії та практики систем обробки та відображення візуальної інформації Інституту проблем математичних машин і систем НАН України Коваленка Олексія Єпифановича на тему «Моделі і методи побудови конвергентних систем ситуаційного управління», яка подається на здобуття наукового ступеня доктора технічних наук, впроваджено у діяльність ДП «ЕС ЕНД ТИ УКРАЇНА» при виконанні проектів системної інтеграції комп'ютерних систем.

Моделі трансформації інформації, ситуаційних агентів та оптимізації інтеграційних рішень з використанням моделей знань, запропоновані Коваленком О. Є. у дисертаційному дослідженні, були використані в процесах планування та організації робіт при реалізації проектів побудови інформаційно-технологічної інфраструктури Центру управління польотом для ДП «Укркосмос» та створення Центра тактичних операцій для військової частини А2245.

Впровадження результатів наукової роботи Коваленка О. Є. у виробничу діяльність ДП «ЕС ЕНД ТИ УКРАЇНА» дозволило скоротити терміни та знизити сукупну вартість виконання проектів за рахунок використання уніфікованого підходу при створенні моделей проєктованих систем.

Генеральний директор
ДП «ЕС ЕНД ТИ УКРАЇНА»
доктор технічних наук



Ю. М. Лисецький

Банківські реквізити:

АТ «Міжнародний Інвестиційний Банк»
Рахунок: UA723805820000026008010306833
МФО: 380582

ДП «ЕС ЕНД ТИ УКРАЇНА»

Україна · м. Київ · 03142 · пр. Академіка Палладіна, 44 «А»
Тел./факс: +38 044 238 63 88; info@snt.ua; www.snt.ua
Код ЄДРПОУ: 20032321
Інд. податковий № 200323226140



**ІНСТИТУТ ПРОБЛЕМ
МАТЕМАТИЧНИХ МАШИН
І СИСТЕМ**

03187, м. Київ
проспект Академіка Глушкова, 42
Телефон: 526 24 97, факс 526 64 57

**НАЦІОНАЛЬНА
АКАДЕМІЯ НАУК УКРАЇНИ**

ІПММС

**ИНСТИТУТ ПРОБЛЕМ
МАТЕМАТИЧЕСКИХ МАШИН
И СИСТЕМ**

03187, г. Киев
проспект Академіка Глушкова, 42
Телефон: 526 24 97, факс 526 64 57

АКТ

**про використання результатів дисертаційного дослідження
Коваленка О.Є. «Моделі і методи побудови конвергентних систем
ситуаційного управління» на здобуття
наукового ступеня доктора технічних наук
за спеціальністю 05.13.05 – Комп'ютерні системи та компоненти**

Цей Акт підтверджує, що результати дисертаційного дослідження Коваленка Олексія Єспифановича були використані при проведенні науково-дослідних робіт, які проводились у Інституті проблем математичних машин і систем НАН України впродовж 2003–2019 років за тематичним напрямом фундаментальних досліджень і науково-технічних розробок «Технології та інструментальні засоби електронного урядування. Інформаційно-аналітичні системи, системи підтримки прийняття рішень. Ситуаційні центри», зокрема у НДР «Методологічні засади створення мережі ситуаційних центрів органів державної влади України та інформаційних технологій підтримки прийняття рішень в розподіленому середовищі», ("Дельта"), 2015-2018 роки, державний реєстраційний № 0115U003014 (відповідальний виконавець); НДР «Комплексні, масштабовані програмно-технічні платформи ситуаційних центрів вищого державного рівня», ("СЦ-Інтеграція"), 2014-2015 роки, державний реєстраційний № 0114U000024; НДР «Розробка теоретичних основ побудови нових моделей, алгоритмів і розподілених інформаційних технологій підтримки прийняття рішень в системах електронного урядування», («Е-Система»), 2011-2014 роки, державний реєстраційний № 0111U002509; НДР «Розробка нових архітектурних рішень для програмно-технічних комплексів систем підтримки прийняття рішень на вищому державному рівні», («СЦ-РАДА»).

При розробці концепції побудови та напрямів розвитку ситуаційних центрів і мереж ситуаційних центрів використано уніфіковану модель трансформації інформації в когнітивному циклі ситуаційної діяльності, яка поєднує різні категорії інформації та забезпечує її адекватне використання в різних видах моделей; модель ситуаційного агента для реалізації процесу трансформації інформації, що дає змогу будувати конвергентні системи ситуаційного управління як композицію ситуаційних агентів; композиційну модель конвергенції компонентів ситуаційного управління з використанням моделей знань, що дає формалізоване представлення процесу побудови цільових ситуаційних систем на основі вимог до них; оптимізаційні моделі оцінки структурної ефективності архітектури конвергентної системи ситуаційного управління, що, на відміну від евристичної моделі розробки, дає змогу формально обґрунтувати архітектурні рішення для цільових систем; розроблено оптимізаційні моделі оцінки динамічної стійкості конвергентних систем ситуаційного управління на основі узагальнених показників, що дозволяє прогнозувати їх спроможності у процесі функціонування; технологічні моделі для побудови компонентів конвергентних систем ситуаційного управління для окремих прикладних застосувань, що є основою для уніфікації таких систем.

Директор
ІПММС НАН України
акад. НАН України, д.т.н., проф.



А.О. Морозов

П.н.с. ІПММС НАН України
к.т.н., с.н.с.

В.В. Вишневський



УНИВЕРСИТЕТ ЗА НАЦИОНАЛНО И СВЕТОВНО СТОПАНСТВО
UNIVERSITY OF NATIONAL AND WORLD ECONOMY

София 1700, Студентски арай
"Христо Ботев",
телефони: 963-00-42, 81-95-211,
81-95-266, факс: 962-39-03,
www.unwe.bg

Bulgaria, 1700 Sofia,
Studentski grad "Hristo Botev"
Phones: (+ 359 2) 963-00-42,
(+ 359 2) 81-95-211, 266,
Fax (+ 359 2) 962-39-03,
www.unwe.bg

To whom it may concern

Pukhov Institute for Modeling
in Energy Engineering (IPME)
National Academy of Sciences of Ukraine
General Naumov Str., 15, Kyiv,
03164, Ukraine

This letter confirms that the results of scientific research of Dr. Oleksii Kovalenko was implemented in project "Methodology for Designing Web-Based Integrated Information System for Risk Assessment of Natural Disasters" R&D Project, № 1-8/2011 (2011- 2014), UNWE, Bulgaria for the computer estimation and decision support about alternatives of computer system architectures for Disaster Risk Management. The content of his research was presented in Doctor of Sciences Thesis "Models and Methods for Designing of Convergent Systems of Situational Control".

The new approach for designing of computer architectures for disaster risk management information processing in complex computer systems was developed by Dr. O. Kovalenko allows to substantiate alternatives of choice of engineering decisions.

Dr. O. Kovalenko also developed new information processing model, which consolidate different categories of information in unified processing cycle. This allows to implement appropriate smart agent model in process of disaster risk management.

Sincerely,

Prof. Dr. Dimitar Velez
Director of Science Research Center for Disaster Risk Reduction
Department of Information Technology and Communications
University of National and World Economy,
Secretary of IFIP WG 5.15 for Information Technology in Disaster Risk Reduction (ITDRR)

UNSS, 8 December str., Student Town,
Sofia 1700, Bulgaria

01.02.2020

ПОГОДЖЕНО

Проректор з навчальної і виховної роботи
НУБіП України,
акад. НААН, д. с. т. н., професор


С. М. Кваша

«__» _____ 2020 р.

ЗАТВЕРДЖУЮ

Перший проректор НУБіП України,
акад. НААН, д. с. т. н., професор

**АКТ**

впровадження результатів дисертаційної роботи Коваленка Олексія Єпифановича на тему «Моделі і методи побудови конвергентних систем ситуаційного управління» на здобуття наукового ступеня доктора технічних наук у навчальний процес кафедри комп'ютерних систем і мереж Національного університету біоресурсів і природокористування України

Комісія у складі: голова – декан факультету інформаційних технологій, доктор педагогічних наук, професор Глазунова О. Г. та члени комісії: завідувач кафедри комп'ютерних систем і мереж (КСіМ) доктор технічних наук, професор Лахно В.А. та доцент кафедри КСіМ, кандидат технічних наук, доцент Гусєв Б. С. склали цей акт про те, що результати дисертаційної роботи Коваленка Олексія Єпифановича «Моделі і методи побудови конвергентних систем ситуаційного управління» впроваджені у навчальний процес та використовуються на кафедрі КСіМ з 2019-2020 навчального року при викладанні навчальної дисципліни «Технології проектування комп'ютерних систем» для студентів спеціальності 123 – Комп'ютерна інженерія.

№ п/п	Результат, що впроваджено	Форма впровадження	Ефект від впровадження
1.	Алгоритми оптимізації архітектури комп'ютерних інформаційних систем.	Навчальна програма дисципліни, лекційний матеріал та методичні розробки для виконання лабораторних робіт	Ознайомлення студентів з параметризацією архітектури комп'ютерної системи, постановкою та методом розв'язання задачі оптимізації
2.	Імітаційні моделі дослідження компонентів комп'ютерних інформаційних систем	Навчальна програма дисципліни, лекційний матеріал та методичні розробки для виконання лабораторних робіт	Ознайомлення студентів з методикою створення імітаційних моделей компонентів комп'ютерних інформаційних систем та їх дослідження

Голова комісії:

декан факультету інформаційних технологій,
доктор педагогічних наук, професор

Члени комісії:

завідувач кафедри комп'ютерних систем і мереж,
доктор технічних наук, професор

доцент кафедри комп'ютерних систем і мереж,
кандидат технічних наук, доцент



О. Г. Глазунова

В. А. Лахно

Б. С. Гусєв



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КАМ'ЯНЕЦЬ-ПОДІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ОГІЄНКА
 вул. Огієнка, 61, м. Кам'янець-Подільський, 32300; тел.: (03849) 3-05-13, факс: (03849) 3-07-83, E-mail: post@kpmu.edu.ua
 Web: <http://www.kpmu.edu.ua> код ЄДРПОУ 02125616

Від 14.02.2020 № 6/20

На № _____ від _____

ДОВІДКА
про впровадження результатів дисертаційного дослідження
«Моделі і методи побудови конвергентних
систем ситуаційного управління»
на здобуття наукового ступеня доктора технічних наук
за спеціальністю 05.13.05 – комп'ютерні системи та компоненти
КОВАЛЕНКА ОЛЕКСІЯ СПИФАНОВИЧА

Впровадження результатів дисертаційного дослідження Олексія Спифановича Коваленка здійснювалося в освітній процес Кам'янець-Подільського національного університету імені Івана Огієнка упродовж 2008-2015 років.

Зважаючи на те, що при побудові та адаптації систем ситуаційного управління для вирішення задач цільової предметної сфери виникає необхідність створення та застосування моделей, методів і засобів, що забезпечують обґрунтування та ефективність технічних рішень, які приймаються в цьому процесі, а також враховуючи вимоги до підготовки фахівців у галузі інформаційних технологій за освітніми ступенями бакалавр та магістр, викладачі кафедри інформатики Кам'янець-Подільського національного університету імені Івана Огієнка використовують в освітньому процесі результати дисертаційного дослідження О.С. Коваленка на тему «Моделі і методи побудови конвергентних систем ситуаційного управління» під час викладання навчальних дисциплін «Системи та методи прийняття рішень» та «Інтегровані системи управління» та при виконанні курсових і кваліфікаційних робіт. Зокрема використовувались композиційна модель конвергенції компонентів системи ситуаційного управління з використанням моделей знань, що дає формалізоване представлення процесу побудови цільових ситуаційних систем на основі вимог до них та оптимізаційні моделі оцінки структурної ефективності архітектури конвергентної системи ситуаційного управління.

Також результати дисертаційної роботи О.С. Коваленка «Моделі і методи побудови конвергентних систем ситуаційного управління» використовуються для вдосконалення теоретико-методологічного забезпечення освітнього процесу.

Результати впровадження обговорено та схвалено на засіданні кафедри інформатики Кам'янець-Подільського національного університету імені Івана Огієнка (протокол № 1 від 10 січня 2020 року).

Проректор з наукової роботи,
доктор фізико-математичних наук, професор



I.M. Конет

Додаток Б

Список публікацій здобувача за темою дисертації

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Морозов А.О., Баран Л.Б., Вишневський В.В., Коваленко О.Є., Малишев О.В. Розділ 6. Ситуаційний центр як засіб колективної роботи законотворців. *Електронний парламент України: досвід створення. Наукове видання* / за заг. ред. С.О. Довгого. Київ : Логос, 2015. С. 395-441. ISBN: 978-617-7361-07-6.
2. Kovalenko O., Velev D. Ad-Hoc Architecture of Systems for Disaster Risk Management. *Information Technology in Disaster Risk Reduction. IFIP Advances in Information and Communication Technology* / Y. Murayama, D. Velev, P. Zlateva (eds.). 2020. Vol. 575. Springer, Cham. P. 135–145. DOI: 10.1007/978-3-030-48939-7_12. (Scopus Q3, DBLP). ISSN 1868-4238, ISSN 1868-422X (electronic), ISBN 978-3-030-48938-0, ISBN 978-3-030-48939-7 (eBook).
3. Kovalenko O., Visnevsky V., Kosolapov V. Models of information processing optimization for technical interoperability in a network of distributed situational centers. *CEUR Workshop Proc.* 2020. Vol. 2608. P. 426–435. (Scopus, DBLP). ISSN 1613-0073.
4. Коваленко О. Є., Косолапов В.Л. Моделі оптимізації завантаження мережі розподілених ситуаційних центрів. *Математичне та комп'ютерне моделювання. Технічні науки.* 2020. Вип. 21. С. 102–113. DOI: 10.32626/2308-5916.2020-21.102-113. (BASE, PKP Index, ICI, SIS, Academic Resource Index (ResearchBib), Google Scholar). ISSN 2308-5916.
5. Коваленко О. Є., Косолапов В.Л. Модель стійкості агентної системи ситуаційного управління. *Математичні машини і системи.* 2020. № 3. С. 93–104. DOI: 10.34121/1028-9763-2020-3-93-104. (Google Scholar). ISSN 1028-9763.
6. Коваленко О. Є. Онтологія та модель трансформації інформації в ситуаційних агентних системах. *Електронне моделювання.* 2020. Т. 42, № 5. С. 5–23. DOI: 10.15407/emodel.42.05.005. (Index Copernicus International, CrossRef, Ulrich's

Periodicals Directory, Google Scholar, Cambridge Scientific Abstracts (CSA), Computer and Information Systems Abstracts, INIS Collection, e-LIBRARY, BIHITI PAH). ISSN 0204-3572 (print), ISSN 2616-9525 (online).

7. Коваленко О. Є. Принципи інженерії ситуаційних систем. *Математичні машини і системи*. 2019. № 4. С. 65–78. DOI: 10.34121/1028-9763-2019-4-65-78. (*Google Scholar*). ISSN 1028-9763.

8. Kovalenko O.E. Knowledge Based Design of Convergent Systems of Situational Management. *Математичні машини і системи*. 2019. № 3. С. 67–74. DOI: 10.34121/1028-9763-2019-3-67-74. (*PIHЦ, Google Scholar*). ISSN 1028-9763.

9. Коваленко О. Є. Моделювання навантаження мережної інфраструктури систем ситуаційного управління. *Математичні машини і системи*. 2019. № 2. С. 101–110. (*PIHЦ, Google Scholar*). ISSN 1028-9763.

10. Власова Т.М., Коваленко О.Є., Косолапов В.Л. Організаційно-інформаційний сервіс для технології прийняття рішень у ситуаційному центрі шляхом проведення нарад. *Математичні машини і системи*. 2019. № 1. С. 68–79. (*Google Scholar*). ISSN 1028-9763.

11. Коваленко О. Є. Композиційна конвергенція інформаційних технологій у системах ситуаційного управління на основі моделі знань предметної області. *Математичне моделювання в економіці*. 2019. Т. 16, № 2. С. 39–44. DOI: 10.35350/2409-8876-2019-15-2-40-45. (*Index Copernicus, Academic Resource Index (ResearchBib), Google Scholar*). ISSN 2409-8876 (print), ISSN 2663-9068 (online).

12. Коваленко О.Є. Модель оцінки ефективності архітектури конвергентної системи ситуаційного управління. *Моделювання та інформаційні технології: зб. наук. праць ІПМЕ ім. Г.Є. Пухова НАН України*. 2019. Вип. 87. С. 11–14. DOI: 10.5281/zenodo.3612232. ISSN 2309-7647. (*Google Scholar*). ISSN 2309-7647.

13. Коваленко О.Є. Системна інженерія та життєвий цикл систем. *Електронне моделювання*. 2018. Т. 40, № 6. С. 61-82. DOI: 10.15407/emodel.40.06.061. (*Index Copernicus International, CrossRef, Ulrich's Periodicals Directory, Google Scholar*,

Cambridge Scientific Abstracts (CSA), Computer and Information Systems Abstracts, INIS Collection). ISSN 0204-3572 (print), ISSN 2616-9525 (online).

14. Коваленко О.Є. Моделі агентно-орієнтованих систем ситуаційного управління. *Математичні машини і системи*. 2018. № 2. С. 96–102. (*Google Scholar*). ISSN 1028-9763.

15. Kovalenko O. E., Kovalenko T. Knowledge Model for Decision Support System Security Management. *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2016)*: proc. (Sofia, 19 February 2019). Sofia, 2019. P. 165–170. (*EBSCOHost, ProQuest, Google Scholar*). ISSN 2367-7635 (print), ISSN 2367-7643 (online).

16. Kovalenko O. E. The Formalization of Organizational Support Creation for Systems of Situational Management. *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE-2015)*: proc. (Sofia, 15 August 2016). Sofia, 2016. P. 292–301. (*EBSCOHost, ProQuest, Google Scholar*). ISSN 2367-7635 (print), ISSN 2367-7643 (online).

17. Коваленко О.Є. Побудова багаторівневих мультиагентних інформаційних систем, заснованих на знаннях. *Information Technology and Security*. 2016. Т. 4, № 2 (7). С. 146–154. (*Index Copernicus, Google Scholar*). ISSN 2411-1031 (print), ISSN 2518-1033 (online).

18. Коваленко О.Є. Застосування хмарних технологій при побудові систем ситуаційного управління. *Information Technology and Security*. 2015. Т. 3, № 1. С. 11–18. (*Index Copernicus, Google Scholar*). ISSN 2411-1031 (print), ISSN 2518-1033 (online).

19. Kovalenko Oleksii E. Knowledge Models for Organizational Maintenance of Situation Centers. *International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE)*: proc. (Sofia, 15 September 2015). Sofia, 2015. P. 241–248. (*EBSCOHost, ProQuest, Google Scholar*). ISSN 2367-7635 (print), ISSN 2367-7643 (online).

20. Коваленко О. Є. Застосування модальної логіки при прийнятті рішень на моделях знань. *Математичне та комп'ютерне моделювання. Технічні науки: зб. наук. праць. Кам'янець-Подільський*, 2012. Вип. 6. С. 106–112. (BASE, PKP Index, NSD, Google Scholar). ISSN 2308-5916.

21. Коваленко О.Є. Оптимізація архітектур модульних систем електронного навчання. *Математичне та комп'ютерне моделювання. Технічні науки: зб. наук. праць. Кам'янець-Подільський*, 2008. Вип. 1. С. 95–99. (BASE, PKP Index, NSD, Google Scholar). ISSN 2308-5916.

22. Kovalenko A.E., Velez D.G. Compositional Model of Design Process in the Problem-Oriented CAD of Applied Software. *Engineering Simulation*. 1998. Vol. 16, Issue 1. P. 129–133. Gordon & Breach Science Publ Inc, Newark, NJ, United States. ISSN 10631100. (Scopus).

23. Kovalenko, A.E., Velez, D.G. Computer-aided programming system for intelligent measurement systems in electronics. *Engineering Simulation*. 1995. Vol. 12, Issue 4. P. 642–649. Gordon & Breach Science Publ. Inc., Newark, NJ, United States. ISSN 10631100. (Scopus).

Наукові праці, які засвідчують апробацію матеріалів дисертації:

24. Kovalenko O., Vishnevsky V., Kosolapov V. Towards Creating the Network of Situational Governance Centers and Decision Making Technologies in Distributed Environments. *15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET-2020)*: proc. (07 May 2020). 2020. Vol. May. P. 540–545. DOI: 10.1109/TCSET49122.2020.235491. (Scopus). ISBN 978-1-7281-5566-1 (electronic).

25. Kovalenko O. Information Taxonomy and Ontology for Situational Management. *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT-2018)*: proc. (7 November 2018). 2018. Vol. 2, Article number 8526723. P. 94–97. DOI: 10.1109/STC-CSIT.2018.8526723. (Scopus, Web of Science Core Collection). ISBN 978-1-5386-6464-3 (electronic).

26. Kovalenko O., Kovalenko T. Knowledge Model and Ontology for Security Services. *2018 IEEE 1st International Conference on System Analysis and Intelligent Computing (SAIC-2018)*: proc. (31 October 2018). 2018. Article number 8516875. DOI: 10.1109/SAIC.2018.8516875. (*Scopus, Web of Science Core Collection*). ISBN 978-1-5386-7196-2 (electronic).
27. Kosolapov, V., Kovalenko, O. Agent Based Modelling Support for Systems of Situational Management. *2018 IEEE 1st International Conference on System Analysis and Intelligent Computing (SAIC-2018)*: proc. (31 October 2018). 2018. Article number 8516818. DOI: 10.1109/SAIC.2018.8516818. (*Scopus, Web of Science Core Collection*). ISBN 978-1-5386-7196-2 (electronic).
28. Vlasova T., Kovalenko O. and Kosolapov V. Organizational-Information Technology for Providing and Decisions Making in Situational Management. *14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET-2018)*: proc. (10 April 2018). 2018. Vol. April. P. 152–157. DOI: 10.1109/TCSET.2018.8336176. (*Scopus*). ISBN 978-1-5386-2556-9 (electronic).
29. Kovalenko O., Kosolapov V. Optimization of Loading of a Network of Distributed Situational Centers. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. 9-ї Міжнар. наук. конф. (м. Кам'янець-Подільський, 14–15 травня 2020 р.). Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2020. С. 89–91.
30. Коваленко О. Є. Архітектурний підхід до управління життєвим циклом систем управління інформаційною безпекою на основі моделей знань. *Кібербезпека енергетики* : зб. пр. конф. Київ : Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, 2019. С. 5–7.
31. Коваленко О.Є. Елементи архітектури систем кібер-енергетики. *Безпека енергетики в епоху цифрової трансформації* : програма та матеріали наук.-практ. конф. Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України (м. Київ, 20 грудня 2019 р.). Київ : ІПМЕ ім. Г.Є. Пухова НАН України, 2019. С. 69–71.

32. Коваленко О.Є. Онтологія та трансформація інформації в системах ситуаційного управління. *Сучасні проблеми інформатики в економіці, управлінні, освіті та подоланні наслідків Чорнобильської катастрофи* : матеріали XVII Міжнар. наук. семінару (м. Київ – оз. Світязь, 2–6 липня 2018 р.). Київ : Національна академія управління, 2018. С. 12–16.

33. Kovalenko O. E. On Consolidation of Information in Systems of Situational Management. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. 8-ї Міжнар. наук. конф., присвяченої 100-річчю Національної академії наук України та 100-річчю Кам'янець-Подільського національного університету імені Івана Огієнка. Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2018. С. 118–119.

34. Коваленко О. Є., Коваленко Т. О. Конвергенція та конфігурування інформаційних технологій в системах ситуаційного управління. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. 8-ї Міжнар. наук. конф., присвяченої 100-річчю Національної академії наук України та 100-річчю Кам'янець-Подільського національного університету імені Івана Огієнка. Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2018. С. 122–123.

35. Коваленко О.Є. Інформаційна технологія консолідації даних в системах ситуаційного управління. *Сучасні проблеми інформатики в економіці, управлінні, освіті та подоланні наслідків Чорнобильської катастрофи* : матеріали XVI Міжнар. наук. семінару (м. Київ – оз. Світязь, 3–7 липня 2017 р.). Київ : Національна академія управління, 2017. С. 13–17.

36. Коваленко О.Є. Ситуаційне управління як проектна діяльність. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. одинадцятої дистанційної наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2017. С. 28–31.

37. Velev D.G., Kovalenko O.E. Research on the Applicability of Virtual Reality in Disaster Risk Reduction. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. одинадцятої дистанційної наук.-практ. конф. з міжнар. участю.

Київ : Інститут проблем математичних машин і систем НАН України, 2017. С. 32–35.

38. Коваленко Т.О., Коваленко О.Є. Модель знань для управління безпекою інформаційної системи. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. одинадцятої дистанційної наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2017. С. 113–116.

39. Коваленко О.Є. Обробка модальностей при прийнятті рішень у системах ситуаційного управління. *Теорія прийняття рішень* : праці VIII міжнар. школи-семінару (м. Ужгород, 26 вересня – 1 жовтня 2016 р.). Ужгород : УжНУ, 2016. С. 139–140.

40. Коваленко О.Є. Застосування онтологій в системах ситуаційного управління. *Сучасні проблеми інформатики в управлінні, економіці, освіті та подоланні наслідків Чорнобильської катастрофи* : матеріали XV Міжнародного наукового семінару, (м. Київ – оз. Світязь, 4–8 липня 2016 року). Київ : Національна академія управління, 2016. С. 84–89.

41. Коваленко О.Є. Технологічні моделі ситуаційного управління. *Проблеми виведення з експлуатації об'єктів ядерної енергетики та відновлення оточуючого середовища INUDECО'16* : зб. матеріалів першої міжнар. конф. (м. Славутич, 25–27 квітня 2016 р.). Славутич : Славутичська філія НТУУ «КПІ», 2016. С. 166–168.

42. Коваленко О. Є., Коваленко Т. О. Елементи модельно-орієнтованого підходу технології розробки систем ситуаційного управління. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. VII міжнар. наук. конф. (м. Кам'янець-Подільський, 20–21 квітня 2016 р.). Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2016. С. 97–98.

43. Коваленко О.Є. Формалізація діяльності систем ситуаційного управління. *Сучасні проблеми інформатики в управлінні, економіці та освіті* : матеріали XIV-го міжнар. наук. семінару (м. Київ – с. Світязь, 29 червня – 3 липня 2015 р.). Київ : СІК ГРУП Україна, 2015. С. 119–123.

44. Коваленко О.Є. Стандартизація формального опису системної архітектури ситуаційних центрів. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доповідей десятої дистанційної наук.-практ. конф. з міжнар. участю. Київ: Інститут проблем математичних машин і систем НАН України, 2015. С. 111–114.
45. Коваленко О.Є. SWOT-аналіз технологій ситуаційного управління. *Обчислювальний інтелект (результати, проблеми, перспективи)* : праці Міжнар. наук.-практ. конф. (м. Київ – Черкаси, 12–15 травня 2015 р.). Черкаси : видавець Чебаненко Ю., 2015. С. 354.
46. Коваленко О.Є. Моделі організаційного забезпечення інформаційних технологій ситуаційних центрів. *Сучасні проблеми інформатики в управлінні, економіці та освіті* : матеріали XIII-го міжнар. наук. семінару (м. Київ – с. Світязь, 30 червня – 4 липня 2014 р.). Київ: Національна академія управління, 2014. С. 105–110.
47. Kovalenko O. E. The Architecture of Agent-Oriented Situation Management System. *Proceedings of 3rd International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2013)* (Sofia, Bulgaria, December 6 – 7th 2013). Sofia, Bulgaria : University of National and World Economy (UNWE), 24 July 2014. P. 621–627. (EBSCOHost, ProQuest).
48. Коваленко О.Є. Інтеграція знань в агентно-орієнтовані системи підтримки прийняття рішень. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : тези доп. VI Міжнар. наук. конф. (м. Кам'янець-Подільський, 4–5 квітня 2014 р.). Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2014. С. 77.
49. Коваленко О.Є. Модель організації віртуальних інформаційних систем для розподіленої колективної обробки інформації. *Обчислювальний інтелект (результати, проблеми, перспективи)* : матеріали 2-ї Міжнар. наук.-техн. конф. (м. Черкаси, 14–18 травня 2013 р.). Черкаси: Маклаут, 2013. С. 367.
50. Коваленко О.Є. Архітектура сервісів організаційного забезпечення ситуаційного центру. *Системи підтримки прийняття рішень. Теорія і практика* :

зб. доп. дев'ятої дистанційної наук.-практ. конф. з між нар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2013. С. 60–63.

51. Kovalenko O. Agents Knowledge Models for Situation Management Systems. *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2012)* (Sofia, Bulgaria , 5–6 October 2012). Sofia, Bulgaria : University of National and World Economy (UNWE), 11 January 2013. P. 125–131. ISBN 978-954-92247-4-0.

52. Kovalenko O. E., Velev D. G. Integration of Information Services in Agent-Oriented Situation Management System. *Proceedings of International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE–2012)* (Sofia, Bulgaria, 5–6 October 2012). Sofia, Bulgaria : University of National and World Economy (UNWE), 11 January 2013. P. 132–135. ISBN 978-954-92247-4-0.

53. Коваленко О.Є. Організація агентно-орієнтованих систем ситуаційного управління. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ: Інститут проблем математичних машин і систем НАН України, 2012. С. 63–65.

54. Kovalenko O. E. Functional Approach in the Simulation Modeling of the Problem-Oriented Information Systems. *Комунікації, інформаційні технології и статистика. Актуални проблеми на теорията и практиката*: колективна монографія. София : Университетско издателство «Стопанство», 2011. С. 278–281.

55. Коваленко А.Е. Построение систем поддержки принятия решений на принципах облачных вычислений. *Proceedings of the International Conference on Application of Information and Communication Technology in Economy and Education (ICAICTEE-2011)* (Sofia, Bulgaria, 2–3 December, 2011). Sofia, Bulgaria : University of National and World Economy, 2011. P. 475–482.

56. Коваленко О. Є. Елементи моделі цілеспрямованих систем орієнтованих на знання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2011. С. 172–174.

57. Коваленко О.Є., Коваленко А.О. Аспекти функціонування інтелектуальних систем, заснованих на знаннях. *Обчислювальний інтелект (результати, проблеми, перспективи)* : матеріали 1-ї Міжнар. наук.-техн. конф. (м. Черкаси, 10–13 травня 2011 р.). Черкаси : Маклаут, 2011. С. 95–96.

58. Коваленко О.Є. Організація комп'ютерного навчання з використанням моделей середовища навчання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2010. С. 160–163.

59. Коваленко О.Є. Оптимізація процесів електронного навчання на основі імітаційних моделей. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : збірник наук. праць за матеріалами четвертої міжнар. наук. конф. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2010. С. 88–91.

60. Коваленко О.Є. Моделювання процедур експертного оцінювання у навчальних системах *Комп'ютерна математика в науці, інженерії та освіті (CMSEE-2010)* : матеріали Четвертої міжнар. наук.-техн. конф. (м. Полтава, 1–31 жовтня 2010 р.). Полтава: ПолтНТУ, 2010. С. 19.

61. Коваленко О.Є. Інтегральна модель засвоєння знань у навчальних системах. *Интегральные уравнения – 2009 = Integral equations – 2009* : сб. тезисов конф. (г. Киев, 26–29 января 2009 г.). Киев : ИПМЭ им. Г.Е. Пухова НАН Украины, 2009. С. 91–92.

62. Коваленко А., Велев Д. Интеграция приложений электронного обучения в среде RIA/SOA. *Приложна информатика и статистика – съвременни подходи и методи* : доклади от Междунар. науч. конф. (г. Равда, България, 25–26 септември 2009 г.). София : Университетско издателство «Стопанство», 2009. С. 44–45.

63. Коваленко О.Є. Онтологічна модель системи прийняття рішень методом голосування. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2009. С. 51–54.

64. Коваленко О.Є. Семантично-функціональний підхід до побудови багаторівневих систем електронного навчання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2008. С. 166–169.

65. Коваленко О.Є. Стандартизація навчальних курсів для глобального освітнянського простору. *Проблеми безперервного професійного розвитку лікарів і провізорів* : зб. пр. наук.-метод. конф. з міжнар. участю. Київ : Національна медична академія післядипломної освіти імені П.Л. Шупика, 2007. С. 512–514.

66. Коваленко О.Є., Вознюк П.Є. Керування конфігурацією модульних систем електронного навчання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2007. С. 137–140.

67. Kovalenko O. Multilayer model of e-learning systems. *Бизнес Информатика* : сб. доклади от междунар. науч. конф. по повод 40-та годишнина на специалност „Информатика“ (г. София, 11–12 Октомври 2007 г.). София : Университет за национално и световно стопанство, 2007. С. 116–120.

68. Коваленко О.Є. Логічна модель стандартизованого навчального курсу для модельно-керованого процесу розробки. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : зб. наук. пр. (за матеріалами міжнар. наук.-метод. конф.). Кам'янець-Подільський : Кам'янець-Подільський державний університет, 2006. С. 161–169.

69. Коваленко О.Є. Особливості розробки стандартизованих навчальних курсів для багаторазового використання. *Системи підтримки прийняття рішень. Теорія і практика* : зб. доп. наук.-практ. конф. з міжнар. участю. Київ : Інститут проблем математичних машин і систем НАН України, 2006. С. 182–185.

70. Kovalenko O. Evaluation of e-learning deployment scale. *Управленски, информационни и маркетингови аспекти на икономическото развитие на балканските страни* : сб. доклади от междунар. науч. конф. (г. София, 3 ноември 2006 г.). София : Университет за национално и световно стопанство, 2006. С. 427–429.

71. Kovalenko O. Efficiency of Corporate Electronic Learning. *Управленски, информационни и маркетингови аспекти на икономическото развитие на балканските страни* : сб. доклади от междунар. науч. конф. (г. София, 4 ноември 2005 г.). София : Университет за национално и световно стопанство, 2005. С. 197–208.

72. Коваленко О.Є. UML-подання моделі екстремального програмування для проектування систем безпеки. *Проблеми впровадження інформаційних технологій в економіці* : матеріали V міжнар. наук.-практ. конф. (м. Ірпінь, 15–17 травня 2004 р.). Ірпінь : Академія ДПС України, 2004. С. 499–502.

73. Коваленко О. Є. Моделювання технологій адаптивного проектування інформаційних систем. *Сучасні проблеми математичного моделювання, прогнозування та оптимізації* : зб. наук. пр. (за матеріалами Всеукр. наук.-метод. конф.) (м. Кам'янець-Подільський, 23–24 вересня 2004 р.). Кам'янець-Подільський: Кам'янець-Подільський державний університет, 2004. С. 111–115.

74. Коваленко О.Є., Антоненко В.М. Технологія програмування та захист інформаційних систем. *Проблеми впровадження інформаційних технологій в економіці* : матеріали IV міжнар. наук.-практ. конф. (м. Ірпінь, 15–17 травня 2003). Ірпінь : Академія ДПС України, 2003. С. 625–629.

Наукові праці, які додатково відображають наукові результати дисертації:

75. Коваленко О.Є. Комп'ютерний практикум з навчальної дисципліни «Системне інтегрування та адміністрування комп'ютерних мереж»: навч.-метод. посіб. Київ: Вид-во ІСЗЗІ НТУУ «КПІ», 2012. 188 с.

76. Коваленко О.Є. Захист інформації та програмних продуктів. Лабораторний практикум: навч. посіб. Київ : КСУ, 2007. 252 с.

77. Коваленко О.Є. Дослідження операцій : навч. посіб. для спеціальностей «Менеджмент організацій», «Фінанси», «Облік і аудит», «Маркетинг». Київ : КСУ-МГІ, 2005. 55 с.

78. Kovalenko O. E. Models and Means for Service Agents Orchestration in Situation Management Systems. *Actual Problems of Economics*, 2014. N 4 (154). P. 462–467. (*Scopus, Index Copernicus, EBSCOhost, Ulrich's Periodicals Directory, EconLit, ABI/Inform (by ProQuest), Cabell's Directories*).

79. Коваленко О. Є. Координація взаємодії програмних агентів у системах ситуаційного управління. *Актуальні проблеми економіки*. 2012. № 12. С. 183–189. (*Scopus, Web of Science Core Collection, Index Copernicus, EBSCOhost, Ulrich's Periodicals Directory*).

80. Коваленко О. Є. Мультиагентна модель колективного прийняття рішень. *Актуальні проблеми економіки*. 2011. № 11. С. 279–285. (*Scopus, Web of Science Core Collection, Index Copernicus, EBSCOhost, Ulrich's Periodicals Directory*).

81. Коваленко О.Є. Стійкість результатів обробки експертних оцінок у системах підтримки прийняття рішень. *Актуальні проблеми економіки*. 2010. № 10. С. 177–181. (*Scopus, Web of Science Core Collection, Index Copernicus*).

82. Коваленко О.Є. Організація процедур прийняття рішень у ситуаційних центрах на основі голосування. *Актуальні проблеми економіки*. 2009. № 6. С. 289–293. (*Scopus, Web of Science Core Collection, Index Copernicus*).

83. Коваленко О.Є. Архітектура модульних систем дистанційного навчання. *Актуальні проблеми економіки*. 2007. № 12. С. 172–176.

84. Коваленко О.Є., Вишневецький В.В. Організація системи дистанційного навчання на основі стандартних компонентів. *Актуальні питання фармацевтичної і медичної науки та практики*. Запоріжжя: Запорізький державний медичний університет, 2007. Вип. XVIII. С. 38–44.

Додаток В

Модулі системи управління проєктами конвергенції

Таблиця В.1 — Перелік файлів програми

Назва файла	Призначення
index.php	Відповідає за вхід зареєстрованого користувача, перевіряє рівень його доступу та визначає права користувача до перегляду певних проєктів
functions.php	Набір функцій програми
install.php	Інсталяція проєкту. Являє собою скрипт, створення необхідних баз даних, таблиць і заповнення їх первинною інформацією
main.php	Головний модуль програми, в якій сконцентровано всі модулі та зв'язки між ними
config.php	Файл конфігурації системи
exit.php	Скрипт виходу з системи
ln/en.php	Англійський мовний пакет для інтерфейсу системи
ln/uk.php	Український мовний пакет для інтерфейсу системи

До переліку файлів входять файли шаблонів сторінок, які знаходяться в директорії „sh“ і мають тип „html“, файли java-скриптів для динамічної роботи з інтерфейсом програми, файли каскадних стилів „css“ і файли структури таблиць компонентів у директорії „comp“.

В.1 Фрагмент коду головного модуля системи управління проєктами конвергенції *main.php*

```

<?
    session_start();
    if(!isset($admin))
    {
        header("Location: index.php");
        exit;
    }

    if ($ch_lang)
    {
        $lng = $lang;
        session_register("lng");
    }

    require ("config.php");
    require ("functions.php");

    $conn=mysql_connect($dbserver,$dbusername,$dbpassword);
    $db=mysql_select_db($dbname,$conn);

    $sh = SFileToStr("sh/main.html");

    $admin_status = GetAdminStatus($admin);

    //-----

    if (isset($admin_status))
    {
        $sh = str_replace("{MENU}", SFileToStr("sh/menu_$admin_status.html"),
$sh);

        $f = file("ln/all.txt");
        $opt = "";
        for ($i=0; $i<count($f); $i++)
        {
            $ex = explode("\t", $f[$i]);
            $opt .= "<option value='". $ex[0]."' %s>". $ex[1]. "</option>";
            if (isset($lng) && $lng == $ex[0] ) $opt = str_replace("%s", "
selected ", $opt);
        }

        $sh = str_replace("{MENU_LANG}", $opt, $sh);
    }

    //-----

    if (isset($part))
    {
        // --- проєкти -----
        /* if ($part == "cr_proj")

```

```

{
    php_track_vars;

    if($Do_cr_proj)
    {
        if (EditProject($HTTP_POST_VARS,$admin)!=false)
        {
            header("Location: main.php?part=notify&text=Створено!");
            exit;
        }else
            $error = "Помилка при створенні нового проекту!";
    }
    if (!isset($Do_cr_proj) || isset($error) )
    {
        $sh = str_replace("{BODY}", SFileToStr("sh/form_cr_proj.html"),
$sh);
        if (!isset ($HTTP_POST_VARS["ID"])) $HTTP_POST_VARS["ID"] =
MaxIdInBase("projects") + 1;
        if (!isset ($HTTP_POST_VARS["ERROR"]) && isset($error))
$HTTP_POST_VARS["ERROR"] = $error;
        $sh = replace_values($sh, $HTTP_POST_VARS);
    }
    }else
    /*if ($part == "look_proj")
    {
        $sh = str_replace("{BODY}", SFileToStr("sh/look_proj.html"), $sh);
        $sql="SELECT * FROM projects ORDER By Id";
        $sqla=mysql_query($sql,$conn);

        $trs="";
        while ($row = mysql_fetch_array($sqla))
        {
            $tr = SFileToStr("sh/look_proj_tr.html");
            $arr = sel_values($tr);
            for ($i=0; $i<count($arr); $i++)
            {
                if (isset($row[FormatStr($arr[$i])]))
                    $tr = str_replace("{". $arr[$i]. "}",
$row[FormatStr($arr[$i])], $tr);
                else
                    $tr = str_replace("{". $arr[$i]. "}", "" , $tr);
            }
            $trs .= $tr;
        }
        $sh = str_replace("{TRS}", $trs, $sh);
    }else*/
    if ( ($part == "editproject" || $part == "edituser") && isset($id))
    {
        php_track_vars;
        $pr = str_replace("edit", "", $part);

        if($Do_cr_proj)
        {
            if (EditProject($HTTP_POST_VARS,$admin)!=false)
            {
                AddJournal("[[user]] $admin [[edit_project]] ".$_POST['NAME']
, true, 1);
                header("Location: main.php?part=notify&text=[[edited!]]");
            }
        }
    }
}

```

```

        exit;
    }else
        $error = "[[Error_when_project_edited]]";
    }
    if($Do_cr_user)
    {
        if (EditUser($HTTP_POST_VARS,$admin)!=false)
        {
            //AddJournal("[[user]] $login [[edit_project]]
".$_POST['NAME'] , true, 3);
            AddJournal("[[user]] ".$_POST["NAME"]." [[edited_by]]
".$admin."[[_om]]" , true, 1);
            header("Location: main.php?part=notify&text=[[edited!]]");
            exit;
        }else
            $error = "[[Error_with_editing_of_user]]";
    }
    if ((!isset($Do_cr_proj) && !isset($Do_cr_user)) || isset($error) )
    {
        $sh = str_replace("{BODY}", SFileToStr("sh/form_cr_$pr.html"),
$sh);
        //if (!isset ($HTTP_POST_VARS["ID"])) $HTTP_POST_VARS["ID"] =
MaxIdInBase("projects") + 1;
        //if (!isset ($HTTP_POST_VARS["ERROR"]) && isset($error))
$HTTP_POST_VARS["ERROR"] = $error;

        $conn=mysql_connect($dbserver,$dbusername,$dbpassword);
        $db=mysql_select_db($dbname,$conn);
        $sql = "SELECT * FROM ".$pr."s WHERE Id=".$id;
        $sqla=mysql_query($sql,$conn);
        $row=mysql_fetch_array($sqla);

        $arr = sel_values($sh);
        for ($i=0; $i<count($arr); $i++)
        {
            if (isset($row[FormatStr($arr[$i])]))
                $sh = str_replace("{".$arr[$i]."}",
$row[FormatStr($arr[$i])], $sh);
            else
                $sh = str_replace("{".$arr[$i]."}", "" , $sh);
        }
    }
}
}

.....

//----

if ($step == 1)
{
    if ($add_comp_group)
        $error = PQuery("INSERT INTO cu_table VALUES
('.(MaxIdInBase('cu_table')+1).", '$comp_group_name', 'COMP_GROUP', 0, 1, 1,
'')" );

```

```

        if ($add_comp && $comp_group!="")
            $error = PQuery("INSERT INTO cu_table VALUES
('.(MaxIdInBase('cu_table')+1).", '$comp_name', 'COMP', $comp_group, 1, 1,
'')" );

        if ($add_umova_group)
            $error = PQuery("INSERT INTO cu_table VALUES
('.(MaxIdInBase('cu_table')+1).", '$umova_group_name', 'UMOVA_GROUP', 0, 1,
1, '')" );

        if ($add_umova && $umova_group!="")
            $error = PQuery("INSERT INTO cu_table VALUES
('.(MaxIdInBase('cu_table')+1).", '$umova_name', 'UMOVA', $umova_group, 1, 1,
'')" );

        if ($del_umova && $del_umova_name!="")
        {
            if (RecInBase('cu_table', 'Type', 'Id', $del_umova_name ) ==
'UMOVA_GROUP' ) $add = "OR(Referral=$del_umova_name)"; else $add="";
            $error = PQuery("DELETE FROM cu_table WHERE
(Id=$del_umova_name)$add ");
        }

        if ($del_comp && $del_comp_name!="")
        {
            if (RecInBase('cu_table', 'Type', 'Id', $del_comp_name ) ==
'COMP_GROUP' ) $add = "OR(Referral=$del_comp_name)"; else $add="";
            $error = PQuery("DELETE FROM cu_table WHERE
(Id=$del_comp_name)$add ");
        }

        $sh = str_replace("{LIST_COMP_GROUP}", GenList("cu_table", "
(Type='COMP_GROUP') ORDER BY Caption"), $sh);
        $sh = str_replace("{LIST_UMOVA_GROUP}", GenList("cu_table", "
(Type='UMOVA_GROUP') ORDER BY Caption"), $sh);
        $sh = str_replace("{LIST_COMP}", GenList("cu_table", "
(Type='COMP')OR(Type='COMP_GROUP') ORDER BY Caption"), $sh);
        $sh = str_replace("{LIST_UMOVA}", GenList("cu_table", "
(Type='UMOVA')OR(Type='UMOVA_GROUP') ORDER BY Caption"), $sh);

        $sh_x = GenTD("cu_table", " (Type='UMOVA') ORDER BY Referral",
$proj_id);
        $sh = str_replace("{TDS}", $sh_x['ret'], $sh);
        $sh = str_replace("{TDS2}", $sh_x['ret2'], $sh);
        $sh = str_replace("{TRS}", GenTR($proj_id), $sh);
    }

    if (!isset($error)) $error = "";

    $sh = str_replace("{ERROR}", $error, $sh);

    //$sh = str_replace("{TRS}", GenMetaForm($base_name, $proj_id), $sh);
}
//----- журнал -----
if ($part == "journal")
{
    $sql = "SELECT * FROM journal ORDER BY Date DESC";

```

```
$sqli = mysql_query($sql,$conn);

$shx = "";

while ( $row = mysql_fetch_array($sqli) )
{
    $shx .= $row["Date"]. "-" . $row["Ip"]. "-" . $row["Msg"] . "<br>";
}

$sh = str_replace("{BODY}", $shx, $sh);
}

echo Lang($sh);

?>
```

Додаток Д

Інформаційне забезпечення системи управління проєктами конвергенції

Таблиця Д.1 – База даних системи управління проєктами конвергенції

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
USERS Таблиця користувачів	Id	INT	Унікальний ідентифікатор користувача
	Login	CHAR	Логін користувача для входу в систему
	Pass	CHAR	Пароль користувача для входу в систему
	Level	INT	Рівень доступу до системи управління проєктами (0 – супер-адміністратор із доступом до всіх даних, 1 – користувач із обмеженим доступом)
	Name	CHAR	Ім'я користувача
	Organization	CHAR	Організація, представником якої є користувач
	Autorize	INT	Авторизованість користувача (0 – неавторизовано, 1 – авторизовано)
	Date	DATETIME	Дата / час реєстрації користувача
PROJECTS Таблиця проєктів дистанційного навчання	Id	INT	Унікальний ідентифікатор проєкту
	Name	CHAR	Назва проєкту
	Autor	CHAR	Автор проєкту – користувач, який його створив
	Organization	CHAR	Організація, для якої призначений даний проєкт
	Description	CHAR	Короткий опис проєкту, додаткова інформація
	Person_name	CHAR	Ім'я контактної особи
	Person_phone	CHAR	Телефон контактної особи
	Person_mail	CHAR	Адреса електронної пошти контактної особи
LAW Таблиця прав доступу користувачів до проєктів	Date	DATETIME	Дата / час реєстрації проєкту
	Id	INT	Унікальний ідентифікатор
	User_id	INT	Ідентифікатор користувача
	Project_id	INT	Ідентифікатор проєкту
	Can_read	INT	Дозвіл переглядати дані даного проєкту для даного користувача (0 – немає дозволу, 1 – є дозвіл)

Продолження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	Can_write	INT	Дозвіл редагувати дані даного проекту для даного користувача (0 – немає дозволу, 1 – є дозвіл)
PROJECT_META Таблиця загальних даних проекту	Id	INT	Унікальний ідентифікатор
	Project_id	INT	Ідентифікатор проекту
	Type	TEXT	Назва групи, до якої належить параметр
	Caption	TEXT	Назва параметра
	Value	TEXT	Значення параметра
	Description	TEXT	Опис, коротка інформація параметра
COMPONENTS Перелік компонентів до кожного проекту	Id	INT	Унікальний ідентифікатор
	Project_id	INT	Ідентифікатор проекту
	Type	TEXT	Тип компоненти (один із декількох стандартних)
	Name	TEXT	Назва компоненти
	Enabled	INT	Увімкненість компоненти (0 – вимкнено, 1 – увімкнено)
	Certain	INT	Обов’язковість компоненти (0 – компонента додаткова, 1 – компонента обов’язкова)
	Description	TEXT	Опис компоненти, додаткова інформація
LMM_PRE Передінсталяційні дані компоненти LMM (модуль управління навчанням)	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	_responsible_contact_person_	TEXT	Відповідальна контактна особа
	_machine_assembled_	TEXT	Дані збірки апаратної частини
	__server_location_	TEXT	Розташування сервера
	__hardware_description_	TEXT	Опис апаратних засобів
	__installed_software_	TEXT	Встановлені програми
	_operating_system_installed_	TEXT	Встановлена операційна система
	__administrator_account_	TEXT	Адміністративний вхід
	___admin_user_name_	TEXT	Логін адміністратора
	___admin_password_	TEXT	Пароль адміністратора
	_network_access_configured_	TEXT	Конфігурація мережі
	__network_settings_	TEXT	Налагодження мережі

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	__ip_address__	TEXT	Ір-адреса
	__subnet_mask__	TEXT	Маска підмережі
	fully_qualified_hostname__	TEXT	Хост
	__default_gateway__	TEXT	Шлюз
	_dns_hostname_resolution_tested__	TEXT	Тестування дозволів хоста DNS
	__remote_access_available_y_n__	TEXT	Увімкнення віддаленого доступу (увімкнено / вимкнено)
	__client_software__	TEXT	Програмне забезпечення клієнта
	__user_name__	TEXT	Логін
	__password__	TEXT	Пароль
	_xfbv_on_unix_systems_installed__	TEXT	Чи встановлений XBTF на Unix-системі
	_temp__	TEXT	Резервне поле
LMM_INS Інсталяційні дані компоненти LMM	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	_temp__	TEXT	Резервне поле
Передінсталяційні дані компоненти LDAP (каталог користувача)	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	_responsible_contact_person__	TEXT	Відповідальна контактна особа
	_supported_ldap_server_identified_installed__	TEXT	Встановлений сервер для LDAP
	__server_location__	TEXT	Розташування сервера
	__operation_system__	TEXT	Операційна система
	__ldap_server_software__	TEXT	Програмне забезпечення для LDAP-сервера
	__fully_qualified_hostname__	TEXT	Хост
	__ldap_directory_port_default_is_389__	TEXT	Порт каталога LDAP (за вмовчанням 389)
	__base_distinguished_name__	TEXT	Початкова назва
	__fully_distinguished_bind_username__	TEXT	Логін відповідального користувача
	__bind_user_password__	TEXT	Пароль відповідального користувача

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	__user_identifier_attribute_e_g_uid__	TEXT	Ідентифікатор / атрибути користувача
	_ldap_bind_user_has_the_right_permission__	TEXT	Права користувача, відповідального за LDAP-сервер
	_ldap_server_accesses_from_all_lms_machines_tested__	TEXT	Доступ до LDAP-сервера з усіх LMS
	_required_users_created_identified__	TEXT	Чи вимагається створення / ідентифікація користувачів
	__was_administrator_user_name__	TEXT	Логін WAS-адміністратора
	__was_administrator_password__	TEXT	Пароль WAS-адміністратора
	__fully_distinguished_lms_administrator_group_name__	TEXT	Повна назва групи адміністраторів LMS
	__lms_administrator_user_name__	TEXT	Логін LMS-адміністратора
	__lms_administrator_password__	TEXT	Пароль LMS-адміністратора
	_temp__	TEXT	Резервне поле
LDAP_INS Інсталяційні дані компоненти LDAP	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	_provider_url__	TEXT	URL постачальника
	_base_distinguished_name__	TEXT	Основна назва
	_ldap_user_id__	TEXT	Ідентифікатор користувача LDAP
	_ldap_password__	TEXT	Пароль до LDAP
	_ldap_provider__	TEXT	Постачальник LDAP
	_temp__	TEXT	Резервне поле
DS_PRE Передінсталяційні дані компоненти DS (сервер доставки)	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	_responsible_contact_person__	TEXT	Відповідальна контактна особа
	_machine_assembled__	TEXT	Дані збірки апаратної частини

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	__server_location__	TEXT	Розташування сервера
	__hardware_description__	TEXT	Опис апаратних засобів
	__installed_software__	TEXT	Встановлені програми
	__operating_system_installed__	TEXT	Встановлена операційна система
	__administrator_account__	TEXT	Адміністративний вхід
	__user_name__	TEXT	Логін адміністратора
	__password__	TEXT	Пароль адміністратора
	__network_access_configured__	TEXT	Конфігурація мережі
	__network_settings__	TEXT	Налагодження мережі
	__ip_address__	TEXT	Ір-адреса
	__subnet_mask__	TEXT	Маска під-мережі
	__fully_qualified_hostname__	TEXT	Повна назва хоста
	__default_gateway__	TEXT	Шлюз за вмовчанням
	__dns_hostname_resolution_tested__	TEXT	Тестування дозволів хоста DNS
	__remote_access_available_y_n__	TEXT	Увімкнення віддаленого доступу (увімкнено / вимкнено)
	__client_software__	TEXT	Програмне забезпечення клієнта
	__user_name__	TEXT	Логін
	__password__	TEXT	Пароль
	temp	TEXT	Резервне поле
DS_INS Інсталяційні дані компоненти DS	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	_server_id_	TEXT	Ідентифікатор сервера
	_base_url_	TEXT	Базовий URL
	temp	TEXT	Резервне поле

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
SMTP_PRE Передінсталяційні дані компоненти SMTP-сервер	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	_responsible_contact_person_	TEXT	Відповідальна контактна особа
	_smtp_server_identified_installed_	TEXT	Встановлений сервер для SMTP
	__server_location_	TEXT	Розташування сервера
	__operating_system_	TEXT	Операційна система
	__fully_qualified_hostname_	TEXT	Повна назва хоста
	__smtp_server_software_	TEXT	ПЗ для SMTP-сервера
	__smtp_relaying_enabled_for_lmm_server_	TEXT	Можливість зміни SMTP для LMM-сервера
	temp	TEXT	Резервне поле
SMTP_INS Інсталяційні дані компоненти SMTP-сервер	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	temp	TEXT	Резервне поле
DATABASE_PRE Передінсталяційні дані компоненти «реляційна база даних»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	_responsible_contact_person_	TEXT	Відповідальна контактна особа
	_supported_database_server_identified_installed_	TEXT	Встановлений сервер для БД
	__server_location_	TEXT	Розташування сервера
	__operating_system_	TEXT	Операційна система
	__fully_qualified_hostname_	TEXT	Повна назва хоста
	__dbms_software_	TEXT	ПЗ для DBMS
	__database_instance_port_	TEXT	Порт управління БД

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	__for_db_on_unix_required_system_parameters_change_d__	TEXT	Для DB2 на Unix: вимога до зміни системних параметрів
	__for_oracle_database_instance_is_in_utf_8_format__	TEXT	Для Oracle: чи БД у форматі UTF-8
	_required_database_client_software_installed_on_all_lms_servers__	TEXT	Необхідний клієнт БД, який проінстальовано на всіх LMS-серверах
	_all_lms_servers_can_connect_to_the_database_server__	TEXT	Чи всі LMS-сервери можуть з'єднуватись із сервером бази даних
	_temp__	TEXT	Резервне поле
DATABASE_NS Інсталяційні дані компоненти «реляційна база даних»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	_temp__	TEXT	Резервне поле
HTTPS_CONTENT_PRE Передінсталяційні дані компоненти «https_content»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	_responsible_contact_person__	TEXT	Відповідальна контактна особа
	_supported_http_server_identified_installed__	TEXT	Встановлений HTTP-сервер
	__server_location__	TEXT	Розташування сервера
	__operating_system__	TEXT	Операційна система
	__fully_qualified_hostname__	TEXT	Повна назва хоста
	__http_server_software__	TEXT	ПЗ для HTTP-сервера
	_content_access_prepared__	TEXT	Підготовка доступу до контенту
	__content_directory_on_file_system__	TEXT	Каталог контенту в файловій системі
	__content_directory_access_url__	TEXT	Доступ до контенту через URL

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	__http_server_configuration_updated__	TEXT	Оновлення конфігурації HTTP-сервера
	__content_deployement_prepared__	TEXT	Підготовка розміщення контенту
	__ftp_user_name__	TEXT	Логін FTP-користувача
	__ftp_user_password__	TEXT	Пароль FTP-користувача
	__ftp_directory__	TEXT	FTP-каталог
	__ftp_directory_is_accessible_from_lmm_machine__	TEXT	Чи є доступ до FTP-каталога через LMM
	__temp__	TEXT	Резервне поле
HTTPS_CONTENT_INS Інсталяційні дані компоненти «https_content»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	__path_to_imported_course_packages__	TEXT	Шлях до імпортованих пакетів-курсів
	__ftp_user_name__	TEXT	Логін FTP-користувача
	__ftp_password__	TEXT	Пароль FTP-користувача
	__notification_email_format__	TEXT	Формат сповіщення по e-mail
	__temp__	TEXT	Резервне поле
HTTPS_LMS_DS_UI_PRE Передінсталяційні дані компоненти «https_lms_ds_ui»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проєкту
	__responsible_contact_person__	TEXT	Відповідальна контактна особа
	__supported_http_server_identified_installed__	TEXT	Встановлений HTTP-сервер
	__server_location__	TEXT	Розташування сервера
	__operating_system__	TEXT	Операційна система
	__fully_qualified_hostname__	TEXT	Повна назва хоста
	__http_server_software__	TEXT	ПЗ для HTTP-сервера

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	_websphere_plugin_installed_and_works_	TEXT	Встановлений робочий плагін для WebSphere
	_lmm_machine_can_connect_to_port_80_of_ds_s_http_server_	TEXT	Чи може LMM з'єднуватись із HTTP-сервером через 80-й порт
	temp	TEXT	Резервне поле
HTTPS_LMS_DS_UI_INS Інсталяційні дані компоненти «https_lms_ds_ui»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	temp	TEXT	Резервне поле
HTTPS_LMS_LM_M_PRE Передінсталяційні дані компоненти «https_lms_lmm»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	_responsible_contact_person_	TEXT	Відповідальна контактна особа
	_supported_http_server_identified_installed_	TEXT	Встановлений HTTP-сервер
	_server_location_	TEXT	Розташування серверу
	_operating_system_	TEXT	Операційна система
	_fully_qualified_hostname_	TEXT	Повна назва хоста
	_http_server_software_	TEXT	ПЗ для HTTP-сервера
	_websphere_plugin_installed_and_works_	TEXT	Встановлений робочий плагін для WebSphere
	_ds_machine_can_connect_to_port_80_of_lmm_s_http_server_	TEXT	Чи може LMM з'єднуватись із HTTP-сервером через 80-й порт
	temp	TEXT	Резервне поле
HTTPS_LMS_LM_M_INS Інсталяційні дані компоненти «https_lms_lmm»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	temp	TEXT	Резервне поле
WEBSPPHERE_PRE Передінсталяційні дані компоненти «WebSphere»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту

Продовження таблиці Д.1

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
	_platform_specific_tips_for_was_5_considered_	TEXT	Чи враховуються нюанси специфіки платформи для WAS 5
	_was_installed_on_all_machines_	TEXT	Встановлений WAS
	__was_admin_user_name_	TEXT	Логін адміністратора WAS
	__was_admin_password_	TEXT	Пароль адміністратора WAS
	__was_installation_path_	TEXT	Шлях для встановлення WAS
	_global_security_and_sso_enabled_	TEXT	Чи увімкнено глобальний захист і SSO
	_websphere_plugin_on_http_server_for_lms_user_interface_	TEXT	Встановлений інтерфейс користувача для плагіна WebSphere на HTTP-сервері
	_separate_application_servers_for_lms_modules_in_was_created_	TEXT	Чи створені виділені сервери додатків для LMS-модулів
	__names_of_created_application_servers_for_lms_modules_	TEXT	Перелік імен створених серверів додатків для LMS-модулів
	temp	TEXT	Резервне поле
WEBSPPHERE_IN S Інсталяційні дані компоненти «WebSphere»	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Project_id	INT	Ідентифікатор проекту
	temp	TEXT	Резервне поле
JOURNAL Журнал подій системи	Id	INT	Унікальний ідентифікатор події
	Msg	TEXT	текст повідомлення
	Vazh	INT	Важливість
	Rez	TEXT	Результат
	Ip	TEXT	Ір-адреса, з якої було вчинено подію
	Date	DATETIME	Дата / час події
	Stan		Стан
CU_TABLE Перелік і властивості компонентів і умов проектів	Id	INT	Унікальний ідентифікатор
	Caption	TEXT	Назва компоненти / умови
	Type	TEXT	Тип (компонента, група, умова)
	Referral	INT	Реферал
	Coef	FLOAT	Коефіцієнт вартості
	Enabled	INT	Увімкненість
	Temp	TEXT	Додаткове поле

Продовження таблиці Д.1

CU_CHECK Задоволеність умов компонентами	Id	INT	Унікальний ідентифікатор
	Comp_id	INT	Ідентифікатор компоненти
	Uмова_id	INT	Ідентифікатор умови
	Value	INT	Значення
CU_SEL_UMOVA Увімкненість умов відповідно до проєкту	Id	INT	Унікальний ідентифікатор
	Project_id	INT	Ідентифікатор проєкту
	Uмова_id	INT	Ідентифікатор умови
	Value	INT	Увімкненість