

УДК 004.415:004.89

П.І. Сагайда (д-р. техн. наук, доц.), А.А. Зорі (д-р техн. наук, проф.)Державний вищий навчальний заклад
«Донецький національний технічний університет», м. Покровськ,
кафедра електронної техніки

E-mail: PaulSagayda@ukr.net; Anatolii.Zori@donntu.edu.ua

**МОДУЛЬНА СТРУКТУРНО-АЛГОРИТМІЧНА ОРГАНІЗАЦІЯ
КОМП'ЮТЕРНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОЇ ОБРОБКИ ДАНИХ
З ЕЛЕМЕНТАМИ ВБУДОВАНИХ СИСТЕМ**

Розроблено модульну структурно-алгоритмічну організацію комп'ютерних систем інтелектуальної обробки даних, що заснована на базах знань про предметну область і про характеристики алгоритмів обробки даних. Застосування запропонованої модульної організації дозволяє гнучко комплектувати системи для інтелектуальної обробки даних залежно від організаційно-технічних особливостей підприємств, у тому числі на основі вбудованих систем. Завдяки використанню розроблених структурно-алгоритмічних рішень швидкість виконання інтерпретованих запитів до онтологічних моделей зросла на 35-40%. Розроблено варіанти організації алгоритмічно-програмних засобів комп'ютерних систем інтелектуальної обробки даних на платформах .NET і Java, залежно від розв'язуваних задач та обмежень обчислювальної потужності комп'ютерних систем.

Ключові слова: комп'ютерна система, інтелектуальна обробка даних, структурно-алгоритмічна організація, запит до онтологічної моделі.

DOI: 10.31474/2075-4272-2018-1-31-41-52

Загальна постановка проблеми. Відомо, що на основі інтелектуальної обробки даних (ІОД) одержують знання у вигляді моделей предметної області (ПрО) з використанням відповідного математичного апарату для їхнього представлення [1, 2]. ІОД дозволяє витягти приховані правила й закономірності в наборах даних, які є нетривіальними й можуть бути корисні для підтримки прийняття рішень та вироблення керувальних впливів на елементи організаційно-технічних систем (ОТС). Для побудови структурних моделей і моделей функціонування ПрО використовуються різні методи й алгоритми ІОД. ІОД розвивалася й розвивається на стику таких дисциплін, як статистика, теорія інформації, машинне навчання, теорія баз даних (БД), тому більшість алгоритмів і методів ІОД було розроблено на основі різних технологій обробки даних. Потрібне узагальнення таких методів і алгоритмів та виділення типових випадків їх використання, результати якого можна формалізувати у вигляді онтологій [3].

Низкою теоретичних і практичних досліджень доведено [4, 5], що в області комп'ютерної й програмної інженерії використання онтологічного підходу дозволяє обґрунтовано обирати сукупність понять і їхніх зв'язків при проектуванні об'єктно-орієнтованих програмних компонентів комп'ютерних систем (КС), при використанні різних діаграмних методик концептуального й структурно-функціонального моделювання КС, а також забезпечити формування й використання баз і сховищ знань про роботу ПрО в процесі обробки даних та отримання прихованих у даних залежностей і моделей функціонування ПрО. Проведені дослідження також показали, що великий сегмент програмного забезпечення, розробленого дотепер для забезпечення інженерії знань, організації сховищ знань (СЗ) і їх обробки на основі онтологічного підходу, засновано на платформі віртуальної

машини Java [6]. Тим часом, модулі корпоративних КС для ІОД часто функціонують на платформі .NET [7]. Під час проектування й реалізації КС для ІОД на вбудованих системах (Embedded Systems – ES), які накладають обмеження на продуктивність мікропроцесорів і обсяг використовуваної пам'яті, виникають бар'єри між розвиненим алгоритмічним і програмним забезпеченням для розгортання й використання сховищ даних і знань (СДІЗ) у КС і апаратними й програмними можливостями з опрацювання формалізованих знань. Традиційними платформами для реалізації застосувань для промислових ES, з погляду використовуваних операційних систем, є Linux- і Unix-подібні системи. Апаратно-програмні бар'єри, що виникають при цьому, проілюстровано на рисунку 1. Такі бар'єри містять у собі, крім апаратних обмежень ES, також обмеження обміну сервісами й динамічного обміну даними між платформами.

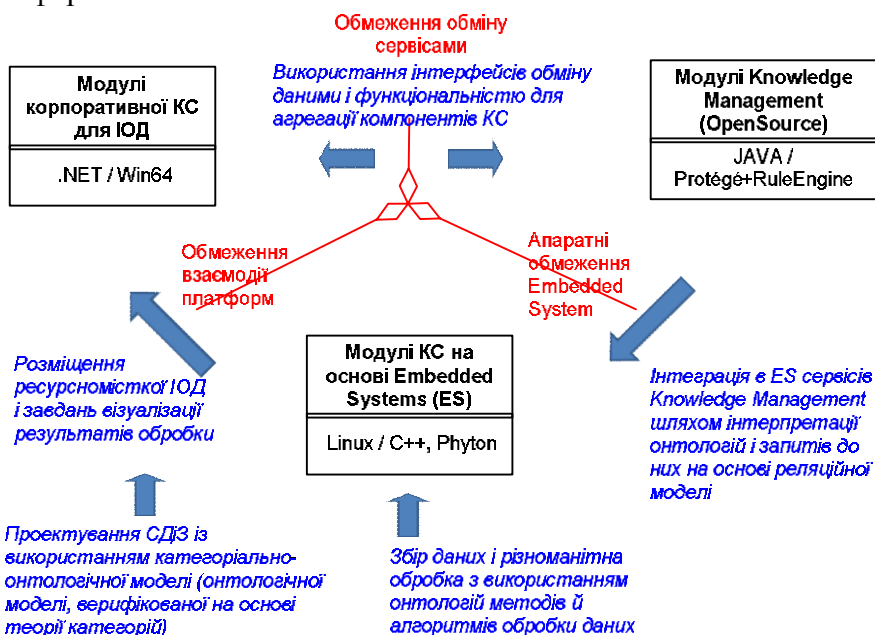


Рисунок 1 – Апаратно-програмні бар'єри та методи подолання обмежень функціональності комп'ютерних систем для інтелектуальної обробки даних, реалізованих із застосуванням Embedded Systems

Самі по собі ці бар'єри долають за допомогою ряду технологій і методик динамічного обміну даними, але ситуація ускладнюється у випадку, якщо в КС для ІОД з використанням ES планується розгортання середніх і великих СДІЗ із інтенсивним використанням їхнього вмісту в реальному масштабі часу.

Метою цієї роботи є розробка модульної структурно-алгоритмічної організації програмних компонентів комп'ютерних систем з елементами вбудованих систем та методики її реалізації.

У роботі запропоновано методику організації алгоритмічно-програмних засобів КС для ІОД з елементами ES, що включає такі заходи:

– проектування СДІЗ із використанням категоріально-онтологічної моделі (онтологічної моделі, верифікованої на основі теорії категорій), із застосуванням методології, розробленої в [5], що дозволяє прискорити проектування й масштабування сховищ, реалізувати раціональні структури СДІЗ;

– розміщення ресурсномісткої частини СДІЗ, реалізації алгоритмів ІОД і завдань візуалізації результатів у модулях корпоративної КС для ІОД, які функціонують на платформі .NET і використовують продуктивні засоби обчислювальної техніки; при цьому обмін оперативними даними про результати вимірів і вироблені керувальні впливи з ES виконується за допомогою відповідних технологій обміну;

– виконання збору даних і їх різноманітної обробки з використанням онтологій методів і алгоритмів ІОД, накопичуваних у СДіЗ КС, що дозволило забезпечити необхідну гнучкість і ефективність процесу обробки даних, адаптацію до мінливих умов збору даних (виникнення завад та збурювальних впливів) і змін виробничого процесу. Такий підхід уможливив модифікацію етапів і алгоритмів ІОД за рахунок зміни аксіом і правил для їхньої онтологічної моделі в складі СДіЗ за результатами прецедентів використання при розв’язанні задач здобуття моделей із даних;

– інтеграція в ES сервісів Knowledge Management на основі моделей інтерпретації онтологій і запитів у вигляді збережених процедур і тригерів для БД під управлінням локальної СУБД. Розроблено алгоритми інтерпретації правил і обмежень ПрО як у вигляді аксіом мовою Ontology Web Language (OWL) Descriptive Logic (DL) і правил мовою Semantic Web Rule Language (SWRL) [8], так і у вигляді процедур мовою Stored Procedures and Triggers Language (SPL), а також методику інтерпретації баз знань і запитів до них на основі реляційної моделі. Така методика дозволила розгорнути частину СЗ (онтологічної моделі ПрО) у вигляді реляційної БД і відповідних тригерів і запитів мовою SQL безпосередньо на платформі ES з використанням СУБД MySQL. Етапи розробленої методики і їхню роль у подоланні наявних бар’єрів проілюстровано на рисунку 1.

Залежно від потреб ОТС зі збору й обробки даних, від вимог до моделей, які отримують із даних, від обсягу розгорнутих онтологічних моделей і кількості фактів, накопичених у реляційних БД у складі СДіЗ, розроблено такі варіанти структурно-алгоритмічної організації корпоративних КС для ІОД.

На рисунку 2 наведено діаграму розгортання КС для ІОД на основі платформи .NET і СУБД MS SQL Server. Такий варіант організації КС дозволяє розгорнути БД, побудовану відповідно до розробленої методики інтерпретації СЗ у вигляді реляційної БД і використовувати для ІОД служби Analysis Services у складі СУБД [9].

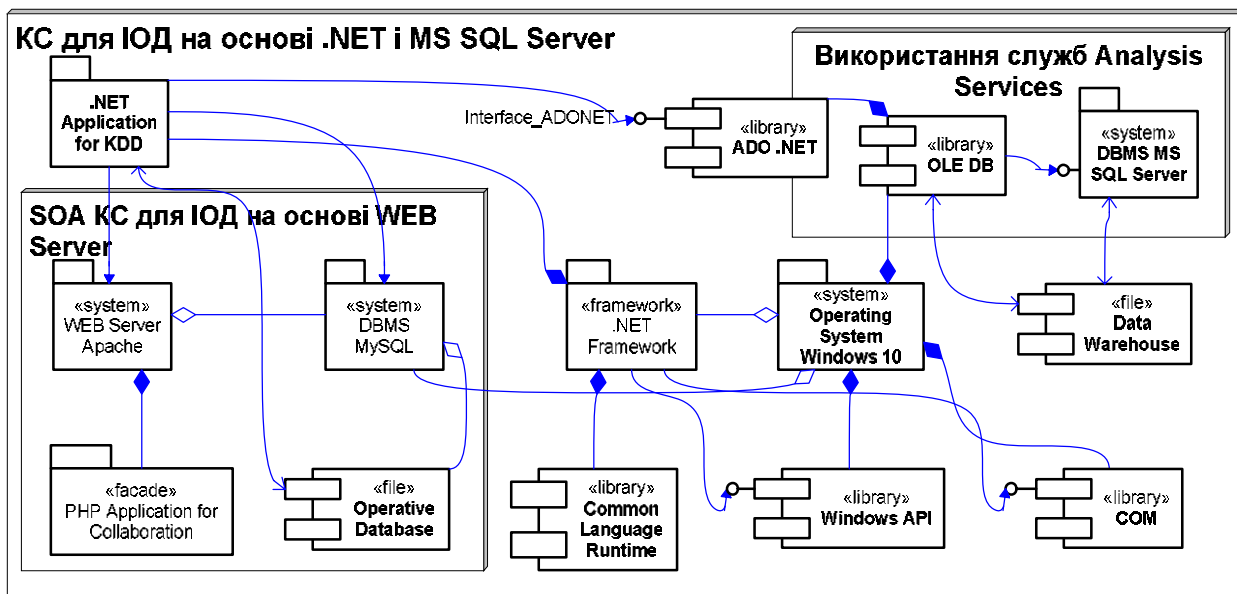


Рисунок 2 – Діаграма розгортання алгоритмічно-програмного забезпечення комп’ютерної системи інтелектуальної обробки даних на основі платформи .NET і СУБД MS SQL Server

Такий варіант організації дозволяє також забезпечити надання функціональності, реалізованої в межах цієї КС для ІОД, віддаленим застосуванням-клієнтам з використанням сервісно-орієнтованої архітектури (SOA). Одне з можливих рішень для цього – розгортання на сервері корпоративної КС веб-сервера з відповідними скриптами (наприклад, веб-сервера

Apache і скриптів мовою PHP).

З урахуванням описаного вище стану з розробкою програмного забезпечення для інженерії знань, для ОТС, що не мають значних вимог до масштабування СДІЗ й обсягу обробки в процесі ІОД, розроблено варіант реалізації КС для ІОД на платформі Java. Діаграму розгортання алгоритмічно-програмного забезпечення такої КС наведено на рисунку 3. Розглянутий варіант структурно-алгоритмічної організації передбачає використання редактора онтологій (наприклад, Protégé), розробку застосування для ІОД на Java, використання резонера (Pellet, Jess або Drools) і бібліотек API для доступу до онтологічних моделей і бази знань (зокрема, OWL і SWRL API). Для реалізації алгоритмів ІОД може застосовуватися одна з чинних, досить розвинених, бібліотек функцій (наприклад, на базі програмного комплексу WEKA), а також власна реалізація алгоритмів у складі застосування для ІОД.

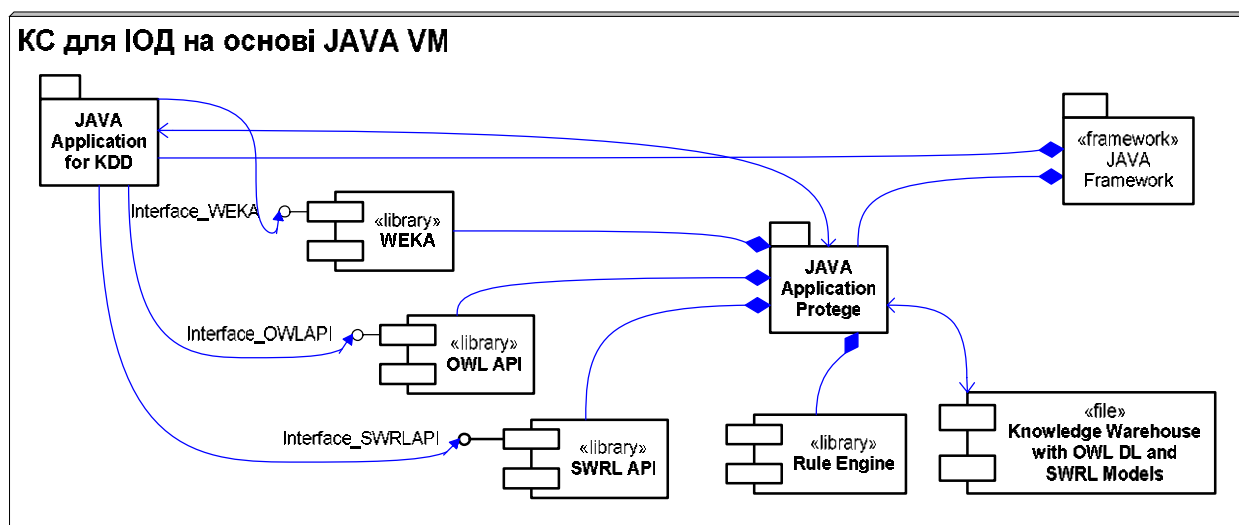


Рисунок 3 – Діаграма розгортання алгоритмічно-програмного забезпечення комп'ютерної системи на платформі Java, для організаційно-технічних комплексів, що не мають значних вимог до масштабування й обсягу обробки в процесі інтелектуальної обробки даних

Для подолання бар'єра між платформами Java і .NET, пов'язаного з використанням різних віртуальних машин для виконання застосувань, запропоновано такий варіант структурно-алгоритмічного рішення, діаграму розгортання якого наведено на рисунку 4.

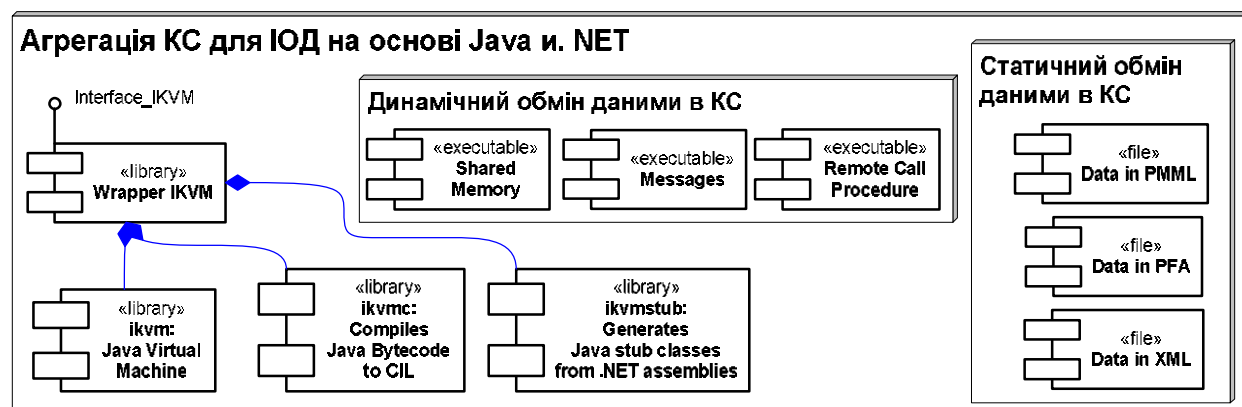


Рисунок 4 – Діаграма розгортання структурно-алгоритмічного рішення для обміну даними й функціональністю між платформами Java і .NET

Запропоноване й апробоване рішення в процесі розробки впроваджених КС складається із трьох компонентів: динамічного обміну даними між застосуваннями, що є компонентами КС і виконуються в різних процесах, з використанням методик, докладно досліджених у [10]; статичного обміну даними між двома платформами в КС із використанням відповідних форматів зберігання даних про моделі; використанням проміжного коду як обгортки (wrapper). Розглянемо третій компонент такого рішення з агрегації двох платформ у межах однієї КС докладніше, на прикладі wrapper IKVM [11]. Його використання якісно вирішує проблему розглянутого бар'єра між платформами. До складу IKVM входять такі засоби: віртуальна машина (VM), яка виконує емуляцію роботи VM Java; засіб трансформації байт-кодів Java у байт-коди CIL (.NET); засоби доступу до просторів імен. Таким чином, застосування .NET дозволяє використовувати, наприклад, OWLAPI і SWRLAPI, написані мовою Java, для ефективного виконання запитів до БЗ, її редагування й логічного виведення за допомогою резонерів (які також є застосуваннями Java). Загальну схему агрегації компонентів КС для ІОД, організованих на платформах Java і .NET з використанням технологій статичного й динамічного обміну даними, а також програмного комплексу IKVM, наведено на рисунку 5. Такий варіант структурно-алгоритмічної організації КС для ІОД дозволяє використовувати наявні розробки (у тому числі OpenSource) в області інженерії знань, запитів до СДіЗ і логічного виведення на основі онтологічних моделей, наданих платформою Java, спільно з функціональними можливостями засобів розробки й бібліотек для платформи .NET і можливостями клієнт-серверних СУБД з інтегрованими службами ІОД (наприклад, AnalysisServices у MS SQL Server).

Аналіз ситуації на ринку програмного забезпечення для ІОД, а також практичні дослідження функціональних можливостей, наданих для використання в режимі клієнт-сервер із застосуванням різних технологій динамічного обміну даними для доступу до наявних бібліотек функцій, програмних пакетів, веб-служб і хмарних сервісів, дозволили розробити варіант структурно-алгоритмічної організації використання сервісів ІОД у межах корпоративної КС. Схему такого варіанта наведено на рис. 6, варіант організації передбачає:

- розміщення в корпоративній мережі підприємства, як складної ОТС, сховища даних (СД) для оперативного обліку даних і СУБД для керування збором і обробкою даних, що дає можливість оперативно організувати СД для розв'язання конкретної задачі аналізу;

- розташування в корпоративній мережі підприємства, на базі виділеного сервера, пакетів для статичної обробки даних і Data Mining, що дозволить виконувати звертання до них із застосувань для ІОД за допомогою пропонованого API на основі технології Component Object Model;

- можливість організації хмарного сховища даних з використанням, наприклад, технології WebDAV [12], що підвищить безпеку даних і знизить потреби в накопичувачах даних; використання веб-сервісів з обробки даних з високою продуктивністю й за допомогою великого вибору алгоритмів ІОД (набір інструментів Google Cloud, TensorFlow і т.п.). Конкретні алгоритми і функціональність, яка реалізовує їх виконання, можуть обиратися в автоматизованому режимі, залежно від масштабу задач аналізу й обмежень на доступ і оплату даних сервісів. Обмін запитом на обробку й результатами ІОД між хмарним сервісом і локальним застосуванням зазвичай здійснюється за допомогою REST API [13].

Завданням, виконання якого вимагає особливих підходів до структурно-алгоритмічної організації, є розміщення компонентів КС для ІОД на основі ES. Такий компонент КС повинен мати можливість низькорівневого доступу до апаратних засобів для збору даних і формування керувальних сигналів технологічному встаткуванню. Як робоче рішення, використане при розробці КС для автоматизації логістичного обліку й ІОД, описане в [14], прийнято мікрокомп'ютер з Linux-подібною операційною системою (на прикладі Raspberri PI 2 і ОС Raspbian). Діаграму розгортання для КС на основі ES, використовуваної як компонент для ІОД, наведено на рисунку 7.

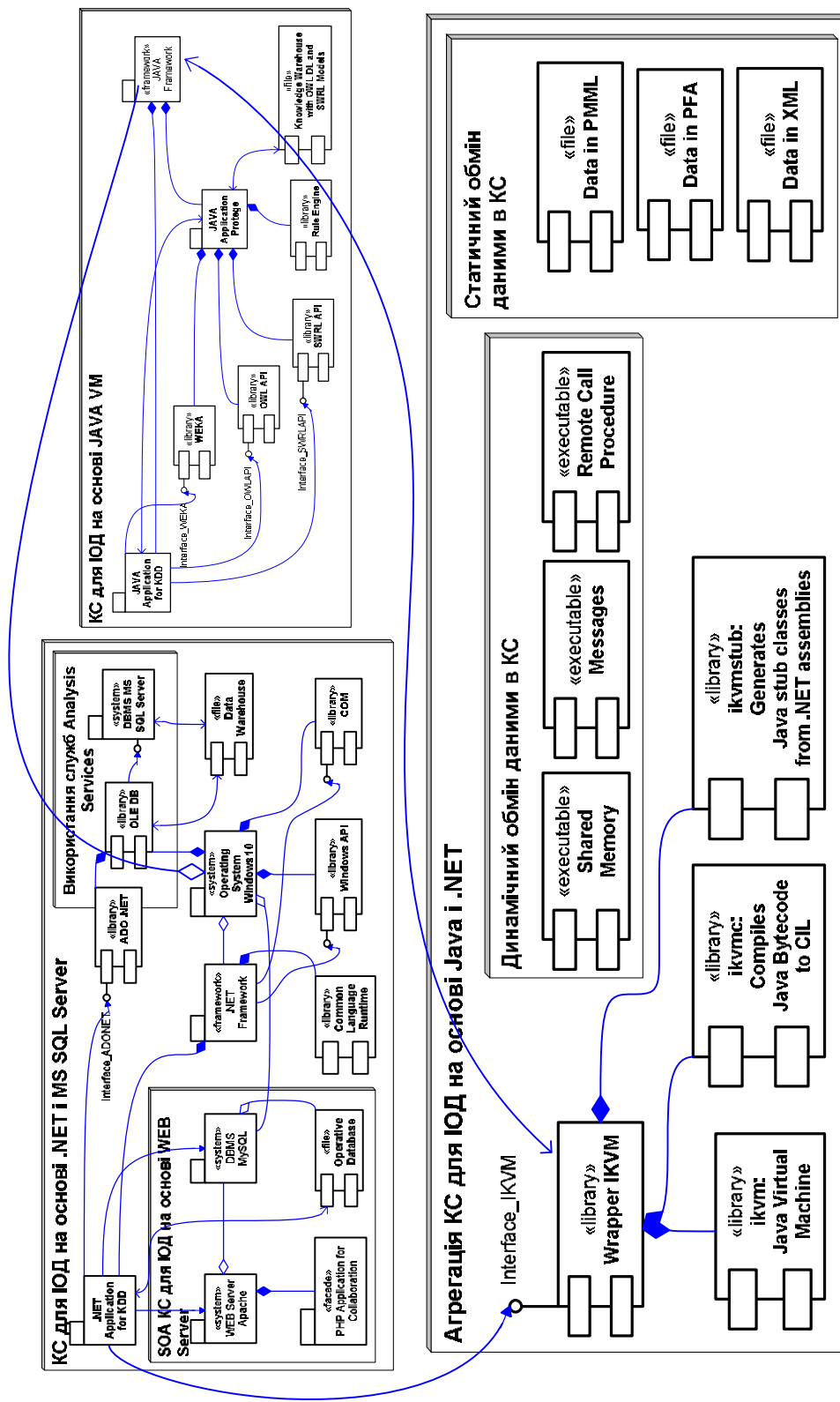


Рисунок 5 – Загальна схема агрегації компонентів комп'ютерної системи інтелектуальної обробки даних, організованих на платформах Java і .NET

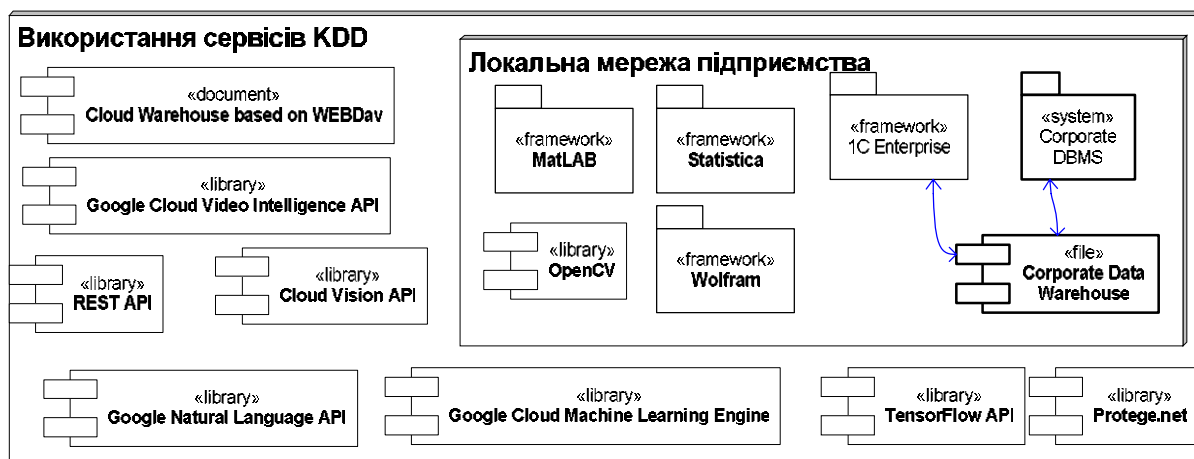


Рисунок 6 – Варіант структурно-алгоритмічної організації використання сервісів інтелектуальної обробки даних у межах корпоративної комп'ютерної системи

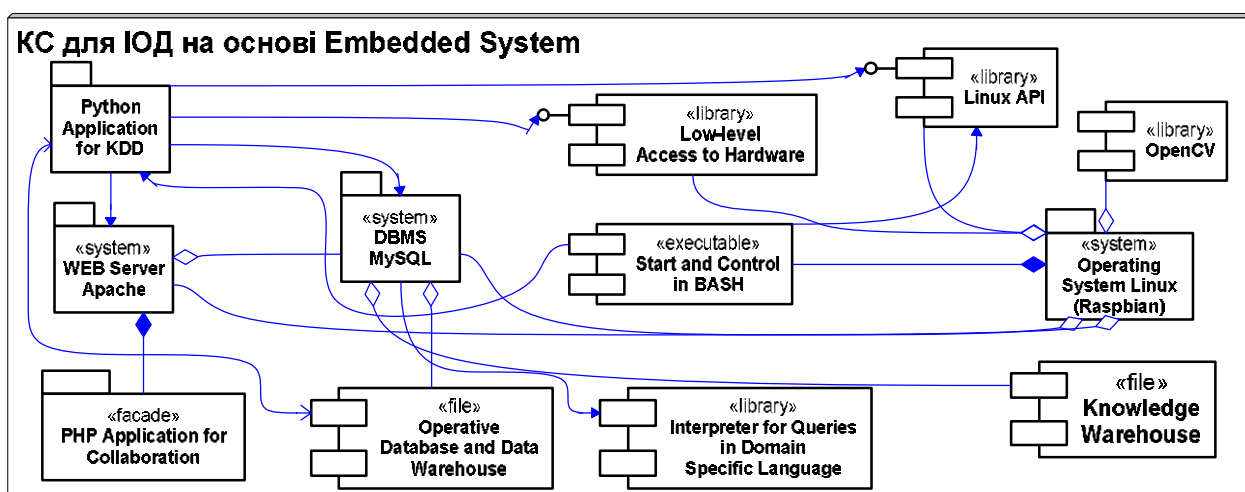


Рисунок 7 – Діаграма розгортання для алгоритмічно-програмного забезпечення комп'ютерної системи на основі Embedded Systems, використовуваної як компонент для інтелектуальної обробки даних

Для подолання апаратно-програмних обмежень, що накладаються на засоби розробки, платформи виконання об'єктного коду й масштаби обробки даних у процесі їх накопичення та ІОД, у запропонованому варіанті використано: застосування для збору й обробки даних мовою програмування Python; методику зберігання знань у вигляді реляційної схеми БД і інтерпретацію правил і запитів до СЗ у вигляді збережених процедур і тригерів; забезпечення обміну даними з віддаленими компонентами корпоративної КС для ІОД через протокол Ethernet з використанням веб-сервера Apache і відповідних скриптів мовою PHP.

Для організації масштабованої КС для ІОД на платформі .NET з компонентами на базі ES, яка має можливість використовувати функціональність бібліотек функцій, статистичних пакетів, пакетів і веб-служб Data Mining і ІОД, розроблено схему структурно-алгоритмічного забезпечення, наведену у вигляді діаграми розгортання на рисунку 8.

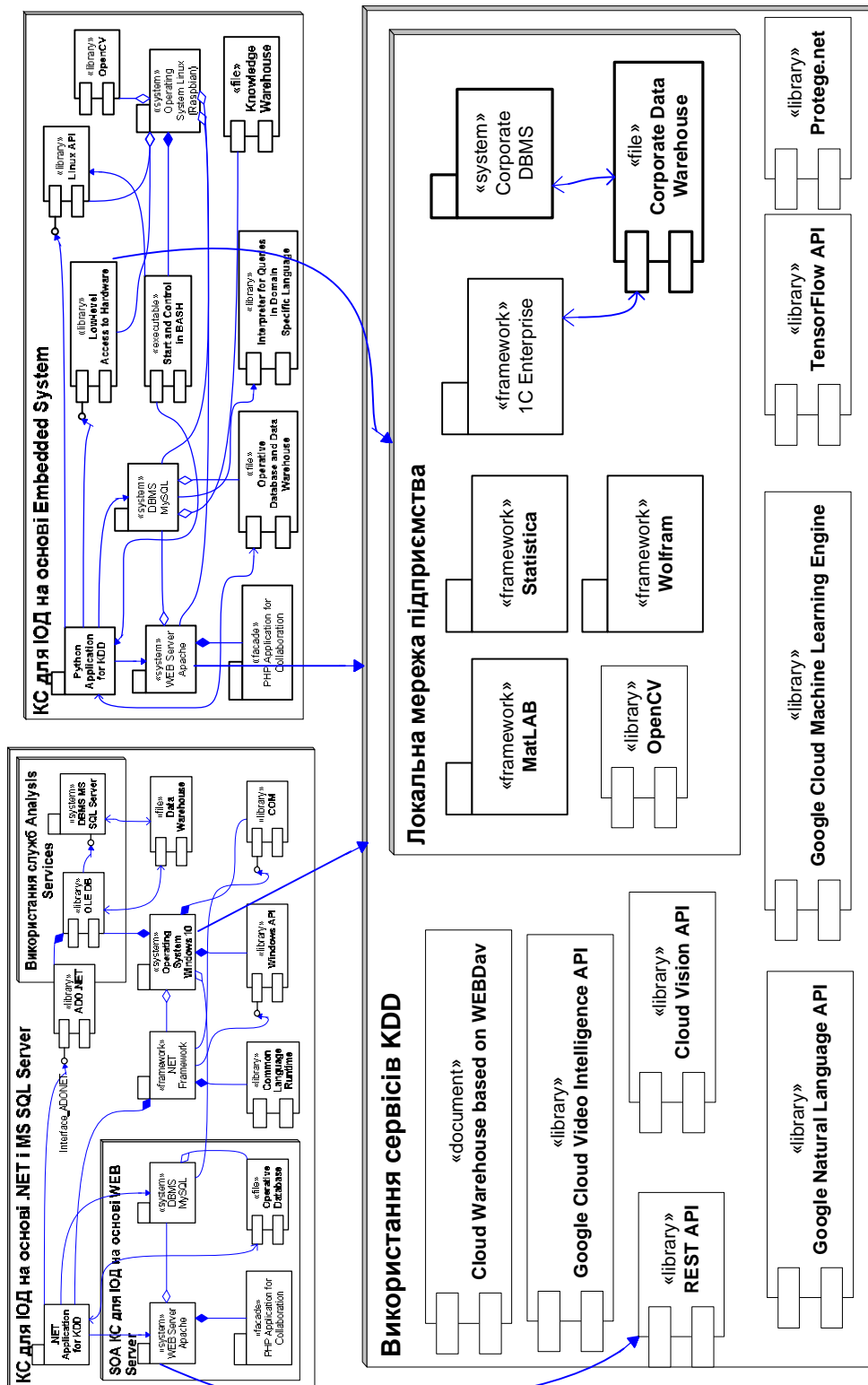


Рисунок 8 – Схема організації алгоритмічно-програмного забезпечення масштабованої комп'ютерної системи інтелектуальної обробки даних на платформі .NET з компонентами на базі Embedded Systems

Результати експериментальних досліджень і впровадження розроблених варіантів структурно-алгоритмічної організації КС, реалізованих на однаковій апаратній платформі, наведено на рис. 9.

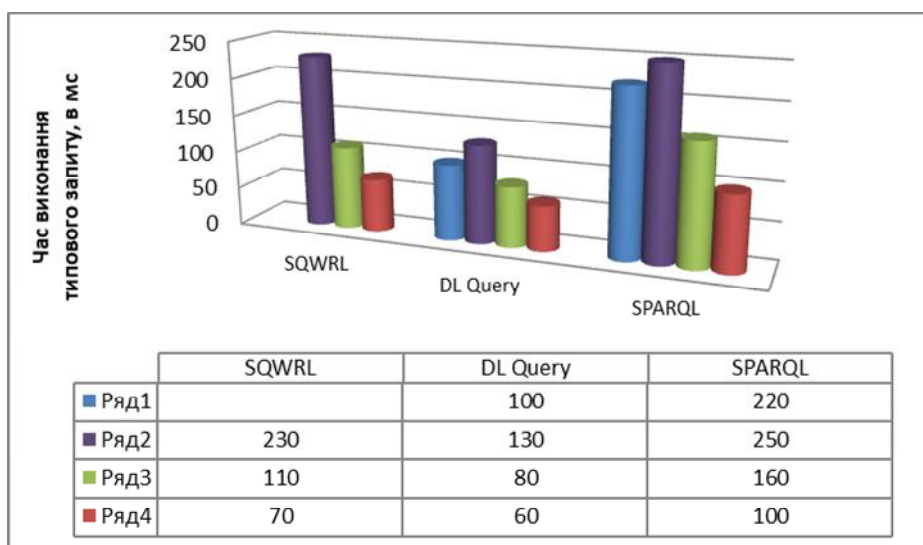


Рисунок 9 – Результати експериментальних досліджень і впровадження розроблених варіантів структурно-алгоритмічної організації комп'ютерних систем для інтелектуальної обробки даних

Графічні залежності на цьому рисунку відображають час виконання типового запиту відповідною мовою запитів до онтологічних моделей: Ряд1 – для застосування на платформі .NET з використанням Rule Engine Drools.NET і OWLAPI; Ряд2 – для застосування на платформі .NET з використанням бібліотеки IKVM, модуля Rule Engine на платформі Java, OWLAPI і SWRLAPI; Ряд3 – для застосування і модуль Rule Engine на платформі Java з використанням OWLAPI і SWRLAPI; Ряд4 – для застосування на платформі Win32 (некерований код) з використанням DBMS MySQL (для реляційного представлення БЗ) і інтерпретацією запитів до БЗ за допомогою SQL та SPL.

Висновки.

1. Розроблена в цій роботі модульна структурно-алгоритмічна організація комп'ютерних систем дозволяє гнучко комплектувати системи для інтелектуальної обробки даних залежно від організаційно-технічних особливостей підприємств, у тому числі на основі вбудованих систем. На відміну від уже наявних, таку структурно-алгоритмічну організацію засновано на базах знань про предметну область і про характеристики алгоритмів обробки, що дозволило забезпечити адаптивність процесу обробки до особливостей масивів даних і підвищити точність розпізнавання образів, забезпечити необхідний обсяг та оперативність виявлення прихованих залежностей.

2. Запропонована методика організації програмних компонентів комп'ютерних систем, що включають до свого складу універсальні пакети й спеціалізовані програми на основі технологій динамічного обміну даними, забезпечила можливість використання реалізованої раніше функціональності для інтелектуальної обробки даних, підтримку прийняття рішень та вироблення керувальних впливів на елементи організаційно-технічних комплексів за результатами опрацювання формалізованих знань. Завдяки використанню запропонованих методик і структурно-алгоритмічних рішень при організації та функціонуванні комп'ютерних систем інтелектуальної обробки даних швидкість виконання інтерпретованих на основі розроблених моделей і запропонованим методом запитів до онтологічних моделей зросла на (35 – 40) %.

3. Розроблено, реалізовано та впроваджено технічні рішення для алгоритмічно-програмних засобів комп'ютерних систем інтелектуальної обробки даних на платформах .NET і Java, а також варіант їх інтеграції й модульної комплектації, залежно від розв'язуваних в організаційно-технічних комплексах задач з інженерії знань та організації сховищ даних і знань, а також залежно від обмежень з обчислювальної потужності й умов функціонування компонентів комп'ютерних систем.

Перелік використаної літератури

1. Барсегян, А.А. Методы и модели анализа данных: OLAP и DataMining / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко и др. – СПб. : БХВ, 2004. – 336 с.
2. Witten, I.H. Data mining: practical machine learning tools and techniques. – 3rd ed. / Ian H. Witten, Frank Eibe, Mark A. Hall. – Burlington, USA. – 2011. – 629 p.
3. Сагайда, П.И. Онтологическое моделирование проблемной области «Интеллектуальный анализ данных для поддержки принятия решений» / П.И. Сагайда // Наукові праці ДонНТУ, серія «Обчислювальна техніка та автоматизація», Донецьк: ДонНТУ, 2010. – Вип. 171. – С. 27 – 33.
4. Hepp, M. Ontology Management: Semantic Web, Semantic Web Services, and Business Applications / M. Hepp, P. De Leenheer, A. de Moor, Y. Sure (Eds.). – Springer, 2007. – 293 p.
5. Sahaida, P. Development of methodology for data and knowledge warehouse design in computer systems for intellectual data processing / P. Sahaida // Technology audit and production reserves. Information and Control Systems. – 2018. – Vol 1., No 2(39). – P. 10 – 15.
6. Nalepa, G.J. Modeling with Rules Using Semantic Knowledge Engineering / G.J. Nalepa. – Cham: Springer, 2018. – 449 p.
7. Troelsen, A. Pro C# 5.0 and the .NET 4.5 Framework / A. Troelsen. – NY: Apress, 2012. – 1560 p.
8. Allemang, D. Semantic Web for the working ontologist: effective modeling in RDFS and OWL / D. Allemang, J. Hendler. – Waltham, USA: Morgan Kaufmann, 2011. – 384 p.
9. Microsoft SQL Server 2005 Analysis Services. OLAP и многомерный анализ данных / А. Бергер, И. Горбач [и др.] – М.: BHV, 2007. – 637 с.
10. Сагайда, П.И. Выбор технологий динамического обмена данными для компонентов интегрированной САПР в многозадачных операционных системах / П.И. Сагайда // Качество образования: управление, сертификация, признание: сборник научных работ международной научно-методической конференции. – Краматорск: ДГМА, 2013. – С. 133 – 140.
11. IKVM.NET Tutorial. Режим доступа: <http://www.ikvm.net/userguide/tutorial.html>.
12. Web-based Distributed Authoring and Versioning. Режим доступа: <https://uk.wikipedia.org/wiki/WebDAV>.
13. Representational State Transfer. Режим доступа: <https://uk.wikipedia.org/wiki/REST>.
14. Sahaida, P. Information-measuring system for vehicle and transported cargo registration using Raspberry PI 2 / P. Sahaida, S. Dobriak // Proceedings of the International Symposium on Embedded Systems and Trends in Teaching Engineering. – Nitra: Constantine the Philosopher University, 2016. – P. 35 – 39.

References

1. Barsegyan, A.A., Kupriyanov, M.S., Stepanenko V.V. et al. (2004), *Metodyi i modeli analiza dannyih: OLAP i DataMinig* [Methods and models of data analysis: OLAP and Data Mining], BHV-Petersburg, St.-Petersburg, Russia.
2. Witten, I.H., Eibe, F. and Hall, M.A. (2011), *Data mining: practical machine learning tools and techniques*, Elsevier, Burlington, USA.
3. Sahaida, P.I. (2010), Ontological modeling of the problem area "Intelligent data analysis for

- decision support" [Ontological modeling problem area "Knowledge data discovery for decision support"], *Naukovi praci DonNTU, ser. «Obchisljuval'na tehnika ta avtomatizacija»*, № 171, pp. 27 – 33.
4. Hepp, M., Leenheer, P.De, de Moor, A. and Sure, Y. (2007), *Ontology Management: Semantic Web, Semantic Web Services, and Business Applications*, Springer, Cham, Switzerland.
 5. Sahaida, P. (2018), "Development of methodology for data and knowledge warehouse design in computer systems for intellectual data processing", *Technology audit and production reserves. Information and Control Systems*, vol. 1, № 2(39), pp. 10 – 15.
 6. Nalepa, G.J. (2018), *Modeling with Rules Using Semantic Knowledge Engineering*, Springer, Cham, Switzerland.
 7. Troelsen, A. (2012), *Pro C# 5.0 and the .NET 4.5 Framework*, Apress, NY, USA.
 8. Allemang, D. and Hendler, J. (2011), *Semantic Web for the working ontologist: effective modeling in RDFS and OWL*, Morgan Kaufmann, Waltham, USA.
 9. Berger, A., Gorbach, I. et al. (2007), *Microsoft SQL Server 2005 Analysis Services. OLAP i mnogomernyy analiz dannykh* [Microsoft SQL Server 2005 Analysis Services. OLAP and multidimensional data analysis], BHV, Moscow, Russia.
 10. Sahaida, P.I. (2013), "Vybor tekhnologiy dinamicheskogo obmena dannyimi dlya komponentov integrirovannoy SAPR v mnogozadachnykh operatsionnykh sistemakh" [Selection of dynamic data exchange technologies for integrated CAD components in multitasking operating systems], *Proc. of the International scientific conference "Quality of education: management, certification, recognition"*, Donbass State Engineering Academy, Kramatorsk, pp. 133 – 140.
 11. IKVM.NET Tutorial. URL: <http://www.ikvm.net/userguide/tutorial.html>.
 12. Web-based Distributed Authoring and Versioning. URL: <https://uk.wikipedia.org/wiki/WebDAV>.
 13. Representational State Transfer. URL: <https://uk.wikipedia.org/wiki/REST>.
 14. Sahaida, P. and Dobriak, S. (2016), "Information-measuring system for vehicle and transported cargo registration using Raspberry PI 2", *Proceedings of the International Symposium on Embedded Systems and Trends in Teaching Engineering*, Constantine the Philosopher University, Nitra, pp. 35 – 39.

П.И. Сагайда, А.А. Зори

ГВУЗ «Донецкий национальный технический университет»

Модульная структурно-алгоритмическая организация компьютерных систем интеллектуальной обработки данных с элементами встроенных систем

Разработана модульная структурно-алгоритмическая организация компьютерных систем интеллектуальной обработки данных, которая основана на базах знаний о предметной области и о характеристиках алгоритмов обработки данных. Применение предложенной модульной организации позволяет гибко комплектовать системы для интеллектуальной обработки данных в зависимости от организационно-технических особенностей предприятий, в том числе на основе встроенных систем. За счет использования разработанных структурно-алгоритмических решений скорость выполнения интерпретированных запросов к онтологическим моделям выросла на 35-40%. Разработаны варианты организации алгоритмически-программных средств компьютерных систем интеллектуальной обработки данных на платформах .NET и Java, в зависимости от решаемых задач и ограничений вычислительной мощности компьютерных систем.

Ключевые слова: компьютерная система, интеллектуальная обработка данных, структурно-алгоритмическая организация, запрос к онтологической модели.

P.I. Sahaida, A.A. Zori

Donetsk National Technical University

Modular structurally-algorithmic organization of computer intelligent data processing systems with embedded system elements

Modular structurally-algorithmic organization of computer systems for intelligent data processing was developed, which is based on knowledge bases of the subject area and of processing algorithm characteristics. This organization allowed to ensure the adaptability of the processing process to the characteristics of data arrays and to provide the required amount and efficiency of detection hidden dependencies. The application of the proposed modular organization allows the flexible composition of systems for intelligent data processing, depending on organizational and technical features of enterprises, including on the basis of embedded systems. Due to the use of developed structurally-algorithmic solutions in the organization and operation of computer systems of intelligent data processing, the performing speed of the ontological model requests which was interpreted on the basis of developed models and the proposed method increased by 35-40%. The technical solutions for the algorithmic and software tools of the computer systems for intelligent data processing on platforms .NET and Java are developed and implemented, as well as the variant of their integration and modular equipment, depending on the problems solved in the organizational and technical complexes on knowledge engineering and organization of data and knowledge warehouses, and depending on the limitations of computing power and the conditions of the functioning of computer system components.

Keywords: computer system, intelligent data processing, structurally-algorithmic organization, request to the ontological model.



Сагайда Павло Іванович, Україна, закінчив Краматорський індустріальний інститут, доктор техн. наук, доцент. ДВНЗ «Донецький національний технічний університет» (пл. Шибанкова, 2, м. Покровськ, 85300, Україна).

Основний напрямок наукової діяльності – розробка, моделювання та дослідження комп'ютерних систем для інтелектуальної обробки даних.



Зорі Анатолій Анатолійович, Україна, закінчив Донецький політехнічний інститут, доктор техн. наук, професор, професор кафедри електронної техніки. ДВНЗ «Донецький національний технічний університет» (пл. Шибанкова, 2, м. Покровськ, 85300, Україна).

Основний напрямок наукової діяльності – розробка методів, засобів і систем вимірювання гідрофізичних та екологічних параметрів водних середовищ, підвищення точності та швидкодії інформаційно-вимірювальних систем.