

В.Л. Міронова, Київ  
М.В. Пирог, Київ

## МІКРОСЕРВІСНА АРХІТЕКТУРА ІНТЕЛЕКТУАЛЬНИХ ТРАНСПОРТНИХ СИСТЕМ ІНФОРМАЦІЙНО-АНАЛІТИЧНОГО ЗАБЕЗПЕЧЕННЯ

**Abstract.** The use of microservice architecture for the construction of information and analytical support of intelligent transport systems is considered. The architecture of the intelligent transport system is proposed based on Web Services Architecture standard as a combination of service-, message- and policy- oriented architectures, to guarantee levels of awareness in messaging processes between intellectual agents.

### Актуальність

Інтелектуальні транспортні системи ITS (Intelligent Transportation Systems) формуються на принципах інтегрування: системи управління рухом (ІТМ – Integrated Traffic Management); мобільних транспортних засобів (МТЗ); з бортових VAC (Vehicle Automation Control) засобів автоматичного управління відповідними агрегатами, процесами, комплексами [1].

Більшість відомих джерел, де розглядаються принципи підвищення обчислювальної працездатності ПАК ITS спрямовано на: методи декомпозиції складних моделей на більш прості компоненти та елементи; паралельних обчислень у розподілених центрах дистанційної спеціалізації ІТ; хмарні технології з застосуванням відповідних мереж та глобальної мережі Internet; спеціалізації та уніфікації розподілу функцій в межах ІТ; середовища за компетенціями та досвідом розв'язування аналогічних задач практики; захист конфіденційних даних та безпеки інформаційної діяльності в умовах OSI функціонування [2 – 3].

За кожним напрямом існують відповідні досягнення. Разом з тим, загальний ефект інтелектуальної інтеграції існуючих прогресивних технологій транспортної галузі для режимів real-time гарантовано-адаптивного управління (ГАУ) МТЗ не дозволяє значно мінімізувати витрати часу на отримання техніко-технологічного рішення (ТТР).

### Постановка задачі

Необхідно *скоординувати* рівні якості та ефективності виконання завдань інтелектуальними агентами системи ITS за критеріями безпеки життя. Тоді часткові критерії вимагають: завчасної підтримки водія (Driver Support System), режимів маневрування; запобігання колізій, зіткнень, лиха; *забезпечення* завчасного, швидкого та якісного обміну повідомленнями між виконавцями цих завдань; спрямування ресурсів на досягнення синергетичного ефекту безаварійного виконання руху МТЗ.

## Вирішення задачі

Синергетичний ефект інтеграції запропонованих спеціалізованих IAS ITS формується своєчасно та достовірно для гарантування ТТР. Це можливо тільки за умов застосування інтелектуальної технології гарантування завчасного, швидкого та якісного обміну повідомленнями між різними програмними додатками та підсистемами, що розосереджені у просторі та поєднані засобами розподілених інформаційних систем (DIS).

Кожен IAS ITS розглядається як агент, що пропонує свої ресурси (апаратні, часові, інтелектуальні тощо) як окремий сервіс з чітко визначеними форматами вхідних та вихідних повідомлень та переліком завдань, які цей агент може виконувати за визначений проміжок часу [4 – 5]. Для забезпечення синергетичної роботи всіх IAS пропонується застосування веб-сервісної технології для забезпечення функціональної стійкості, швидкодії та якісної взаємодії між компонентами ITS. Використання веб-сервісів, як мікросервісів у загальній інформаційній системі, дозволяє розділити та раціоналізувати логіку такої системи на модулі. Кожен компонент ITS являє собою незалежний блок-сервіс з чітко означеними характеристиками та протоколами взаємодії з іншими блоками.

Функціональні модулі реалізують задачі прийняття рішень у складних динамічних ситуаціях:

$$S(t) = \begin{cases} > 1 - f(a) \\ 0 - f(a, b) \\ < 1 - f(b, c) \end{cases} \quad (1)$$

де  $a$ ,  $b$ ,  $c$  необхідні і достатні дії інтелектуальних агентів для продовження у означеній ситуації нормального функціонування системи.

Означені дії можуть бути виконані за допомогою одного чи декількох мікросервісів, розподілених у просторі інтелектуальної системи та розробленими за допомогою різних технологій і стандартів.

Як показано в табл. 1, кожна з задач ГАУ може бути вирішена декількома мікросервісами, вибір яких здійснюється в залежності його доступності, швидкодії, достатності вхідної інформації для запуску сервісу, доступності та відповідності стандарту протоколу обміну повідомленнями та інших протоколів ISO тощо. Як видно з таблиці для забезпечення необхідного та достатнього рівня якості та ефективності виконання завдань інтелектуальними агентами системи ITS авторами пропонується існування, як мінімум трьох альтернативних контурів виконання задачі ГАУ. Необхідне позначення резервних сервісів для забезпечення стабільності контуру та рішення задачі в цілому. Ця таблиця також має бути доповнена статистичними даними по часу  $t$  виконання кожної з задач та часу  $t'$  – обміну повідомленнями між сервісами, що входять в той чи інший контур.

Таким чином, кінцеві результати вирішення задачі ГАУ будуть отримані на двох чи більшій кількості сервісних систем. Запропонований підхід до побудови архітектури системи дозволяє мінімізувати відмови, розширити масштаби системи, розподілити навантаження між різними обчислювальними вузлами. Це, в свою чергу, дозволяє підвищити функціональну стійкість та продуктивності системи в цілому, спростити процес масштабування таких систем, додавання нових чи вилучення технологічно застарілих чи несправних ІАС, що виникають в процесі експлуатації та навчання ІТS.

Швидкодія та стабільність пропонуємих процесів прийняття рішень у ІТS залежить від архітектури взаємодії, взаємозалежності та автономності відокремлених веб-сервісів, що направлена на ГАУ МТЗ.

Таблиця 1

Приклад доступних мікро-сервісів для вирішення поточної задачі ГАУ

Доступні сервіси: технології реалізації	Задачі ГАУ у ситуації S(t)								
	f(a)	f(b)	f(c)	f(a,b)	f(a,c)	f(b,a)	f(b,c)	f(c,a)	f(cb)
S1: .Net, JavaScript, SOAP	+		+		+		+	+	
S2: Java, SOAP, XML		+		+		+	+		+
S3: NodeJS, REST API, json	+		+	+	+			+	
S4: Web API, json, xslt	+		+			+	+		+
S5: Java (Spring), JavaScript, json		+				+		+	+
S6: Python (Django), REST API, json	+			+	+	+	+		+
S7: Python, SOAP API, xml, json		+		+		+	+	+	
S8: Golang, Protobuf	+		+		+		+		
S9: C++, XML		+						+	+
S10: Ruby, REST API	+		+		+		+		+

+ – Сервіс доступний для виконання задачі ГАУ  
 + – Обраний контур сервісів для виконання задачі ГАУ  
 + – 1-й Альтернативний контур сервісів для виконання задачі ГАУ  
 + – 2-й Альтернативний контур сервісів для виконання задачі ГАУ

Згідно стандарту W3C - Web Services Architecture [6 – 8] існує 4 основні моделі архітектури систем заснованих на веб-сервісах: модель орієнтована на повідомлення, сервісно-орієнтована модель, ресурсно-орієнтована модель та модель політики доступу. Усі компоненти тим чи іншим чином пов'язані між собою (рис. 1).

Для вирішення вищезначеної задачі гарантування рівня якості та швидкості реагування ІТS аварійні ситуації та вирішення задач ГАУ авторами пропонується подальший розвиток архітектури за напрямом: Policy Model – Service Oriented Model – Message Oriented Model.

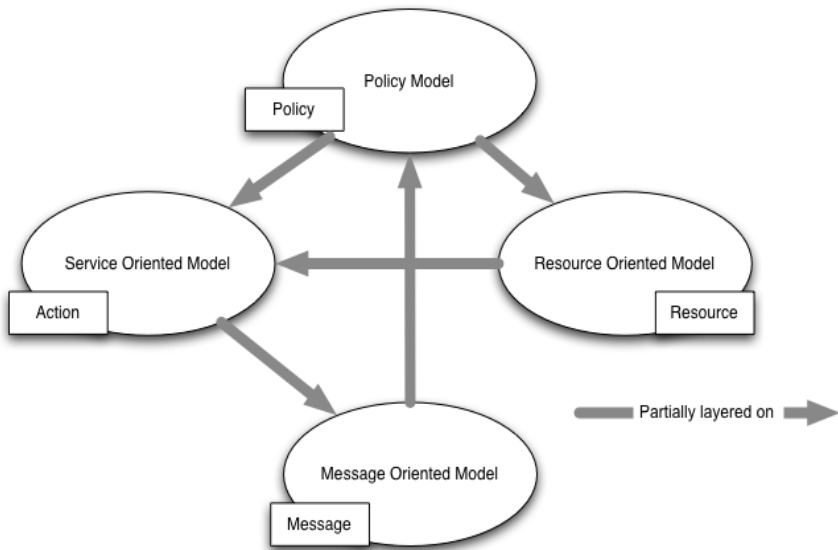


Рис. 1. Зв'язок моделей архітектури систем заснованих на веб-сервісах.

Модель політики доступу зосереджується на регламентації поведінки агентів та серверів. Ресурсні обмеження застосовуються до агентів, які намагатимуться отримати доступ до ресурсів ІТS та зовнішніх інформаційних систем. Завчасне представлення інтересів безпеки, якості послуг, проблем, впливає на ефективність та розв'язання проблем, пов'язаних із оперативним застосуванням. Модель, що орієнтована на повідомлення, фокусується за форматом повідомлень, з фіксуванням їх структур та передачі і т. д. – без спеціального посилання на джерело повідомлення, а також їх значення. Сутність цієї моделі визначається кількома ключовими концепціями: агентом, який надсилає та отримує повідомлення; структурою повідомлення з точки

зору заголовків та тіл повідомлень; а також механізмів, що використовуються для доставки повідомлень. Сервісно-орієнтована модель зосереджена на аспектах обслуговування, дії тощо. Очевидно, що в будь-якій розподіленій системі сервіси не можуть бути адекватно реалізовані без ключових засобів обміну повідомленнями, але повідомлення не повинні впливати на якість функціонування сервера. Сервіси опосередковуються за допомогою повідомлень між агентами, що роблять запити та агентами провайдерів, які пов'язані каналами телекомунікації Internet.

## **Висновки**

Запропонована мікросервісна архітектура інтелектуальних транспортних систем, за допомогою розвитку інформаційно-аналітичного забезпечення, гарантує підвищення продуктивності та функціональну стійкість таких систем.

Інноваційний результат забезпечує гарантований, прискорений та завчасний обмін повідомленнями з природним та штучним інтелектом між інтелектуальними агентами, що приймають участь у процесах виконання задач ГАУ у поточний період часу реалізації рейсів запланованим маршрутом.

Подальша розробка IAZ ITS спрямована на покращення взаємодії за рахунок адаптації наявної архітектури центру, сервісно-орієнтованої архітектури та архітектури заснованої на повідомленнях.

1. Баранов Г.Л. Фундаментальні властивості та відношення в сучасних системах навігації зв'язку та управління рухом/ Г.Л. Баранов, А.М. Носовський, В.І.Тарасюк// Системи управління, навігації та зв'язку . – К: ЦНДІНУ, 2011. – Вип.1 (17). – С.2-9.
2. The Intelligent Transport Systems Technology Action Plan. Електронний ресурс. Режим доступу: [<http://www.transport.govt.nz/assets/Uploads/Our-Work/Documents/Intelligent-Transport-Systems-Technology-Action-plan-June-2014.pdf>]
3. ISO 21217: "Intelligent transport systems - Communications access for land mobiles (CALM) - Architecture".
4. Баранов Г.Л., Банішевський С.А., Міронова В.Л., Пасечник Д.В. Інформаційно-аналітичне забезпечення інтелектуальних транспортних систем. Інтеграція інформаційних технологій на транспорті. Монографія. / Г.Л. Баранов, С.А. Банішевський, В.Л. Міронова, Д.В. Пасечник // Київ. НТУ, 2009 – 198с.
5. Автоматизированные системы обработки информации и управления на автомобильном транспорте: Учебник для сред. проф.образования/А.Б.Николаев, С.В.Алексахин, И.А.Кузнецов, В. Ю. Строганов; Под ред. А. Б. Николаева. – М.: Издательский центр "Академия", 2003. - 224 с., ISBN 5-7695-1184-2.
6. W3C standard. Web Services Architecture. Електронний ресурс. Режим доступу: [<https://www.w3.org/TR/ws-arch/wsa.pdf>]
7. What are Web Services? Architecture, Types, Example. Електронний ресурс. Режим доступу: [<https://www.guru99.com/web-service-architecture.html>]
8. Web Services Reliability. Електронний ресурс. Режим доступу: [<https://www.w3.org/TR/ws-arch/#reliability>]

**<http://doi.org/10.5281/zenodo.3612238>**

*Поступила 12.09.2019р.*