

Гороховський С. С., Ямковий О. В., Кирієнко О. В., Печкурова О. М.

АГЕНТНО-БАЗОВАНА ПЛАТФОРМА ІОТ ДЛЯ КОНТРОЛЮ НАВЧАЛЬНОГО ПРОЦЕСУ

У теоретичній частині роботи розглянуто базові положення концепції Інтернету речей та описано вимоги, які висувають до систем IoT. Також описано основні можливості та переваги використання програмних агентів та мультиагентних систем у реалізації різних IoT.

У практичній частині запропоновано архітектурне рішення на основі мультиагентної системи для підтримки та управління комунікацією у мережі IoT обладнання. Наведене рішення здатне розв'язати проблеми неоднорідності пристроїв та комунікації, побудови гнучкої інфраструктури комунікації та інфраструктури групової комунікації в агентному середовищі, а також на рівні агента. Також було запропоновано агентну платформу, що полегшує комунікацію агентів, які працюють на різних пристроях (датчики SunSpot, смартфони) і використовують різні комунікаційні протоколи, надає можливість формувати агентів у групи для ефективного доставлення групових повідомлень.

Продемонстровано переваги такого рішення на прикладі реалізації «Розумного Університету» (системи підтримки та управління навчальним процесом, що складається з різних типів датчиків та персональних смартфонів учасників навчального процесу) за використання розробленої мультиагентної платформи.

Ключові слова: Інтернет речей (Internet of Things, IoT), моделі комунікації, мультиагентна система, програмний агент.

Вступ

Останнім часом дедалі популярнішою та актуальнішою стає концепція Інтернету речей (IoT) – мережі датчиків та різноманітних пристроїв, що програмно взаємодіють між собою через мережу Інтернет.

Термін «Інтернет речей» (IoT) вперше використав у 1999 р. британський піонер технологій Кевін Ештон для опису системи, в якій об'єкти фізичного світу можуть бути під'єднані до Інтернету за допомогою датчиків. Ештон вигадав цей термін, щоб проілюструвати здатність підключати мітки радіочастотної ідентифікації (Radio-Frequency Identification, RFID), що використовуються в корпоративних мережах, для того, щоб розраховувати та відстежувати товари без необхідності втручання людини. Сьогодні Інтернет речей став популярним терміном для опису сценаріїв, в яких можливості з'єднання та обчислень в Інтернеті поширюються на різні об'єкти, пристрої, датчики та повсякденні елементи.

«Речами» в Інтернеті речей зазвичай називають глибоко вбудовані пристрої з такими відмінними особливостями, як низька пропускання здатність, збір даних із низкою повторюваністю і малий обсяг використовуваних даних. Ці при-

строї обмінюються даними один з одним і передають дані через інтерфейси. Деякі вбудовані пристрої IoT, наприклад, охоронні відеокамери високої роздільної здатності, відеотелефони VoIP та інші, потребують для роботи широкосмислового підключення. Однак для безлічі інших продуктів передання пакетів даних потрібне лише час від часу [6].

Широкомасштабне впровадження IoT допоможе змінити на краще багато аспектів нашого життя. Технологія IoT надає можливість удосконалення сільського господарства, промисловості та виробництва та розподілу енергії шляхом збільшення доступності інформації по ланцюжку виробництва за допомогою мережевих датчиків. Однак IoT порушує багато питань та проблем, які необхідно розглянути та вирішити, щоб зрештою реалізувати потенційні вигоди.

Зрозуміло, що ця взаємодія має здійснюватися за допомогою спеціального програмного забезпечення. У літературі запропоновано багато підходів до розробки такого програмного забезпечення. Одним із ключових підходів є використання інтелектуальних програмних агентів та мультиагентних систем.

У цій роботі описано запропоноване рішення побудови системи підтримки та управління навчальним процесом «Розумний Університет» на

основі мультиагентної системи для підтримки та управління комунікацією у мережі IoT обладнання.

1. Інтернет речей

Відділ стандартів зв'язку МСЕ (Міжнародний союз електрозв'язку, International Telecommunication Union) опублікував Рекомендацію Y.2060 за назвою «Огляд Інтернету речей» (Overview of the Internet of Things) [7]. У цьому документі поняття Інтернет речей (Internet of things (IoT)) визначається як глобальна інфраструктура для інформаційного середовища, яка забезпечує можливість надання більш складних послуг шляхом з'єднання один з одним (фізичних і віртуальних) речей на основі наявних і функціонально сумісних інформаційно-комунікаційних технологій, що розвиваються.

У роботі [4] елементи IoT зведені в просту формулу:

*Фізичні об'єкти + контролери, сенсори,
виконавчі механізми + Інтернет = IoT.*

Ця формула чітко описує суть Інтернету речей. Екземпляр IoT складається з набору фізичних об'єктів, кожен з яких містить мікроконтролер, що забезпечує інтелектуальність, і датчик, що вимірює будь-який фізичний параметр, і/або виконавчий механізм, що спрацьовує від будь-якого фізичного параметра та підтримує можливість комунікації через Інтернет або будь-яку іншу мережу.

З операційного погляду, важливо думати про те, як пристрої IoT підключаються та спілкуються між собою з точки зору технічних моделей зв'язку. У березні 2015 р. Рада Інтернет-архітектури (Internet Architecture Board, IAB) випустила архітектурний документ для мереж інтелектуальних об'єктів (RFC 7452), в якому окреслено чотири загальні моделі зв'язку, що використовуються пристроями IoT [3].

У згаданій рекомендації Y.2060 [7] описано еталонну модель IoT. Вона має чотири рівні: рівень програмного додатку, рівень підтримки послуг і підтримки програмних додатків, рівень мережі, рівень пристрою, а також можливості керування та можливості забезпечення безпеки, пов'язані з цими чотирма рівнями.

2. Програмні агенти

Агентна система (або Multi-Agent System, MAS) – програмне середовище, в якому окремі агенти взаємодіють один з одним, іноді спільно,

іноді конкурентно, а іноді й автономно реалізують свої індивідуальні цілі, отримуючи доступ до ресурсів і сервісів навколишнього середовища, а також час від часу створюючи результати для суб'єктів, які ініціювали цих агентів.

Агентні системи складаються з агентів та їх оточення. Як правило, мультиагентні системи пов'язані з програмними агентами. Проте агенти в мультиагентній системі також можуть бути роботами, людьми або групами людей.

Агенти можна поділити на види, які варіюються від простого до складного:

- пасивні агенти, або «агенти без цілей» (наприклад перешкода, яблуко або ключ у будь-якому простому моделюванні);
- активні агенти з простими цілями (наприклад, птахи у зграї, або вовк – овечка в моделі здобич – жертва);
- когнітивні агенти (складні розрахунки).

Агентні середовища можна поділити на віртуальні, дискретні, безперервні. Також вони можуть бути організовані відповідно до таких властивостей, як доступність (якщо можна зібрати повну інформацію про навколишнє середовище), детермінізм (якщо дія дає певний ефект), динаміка (скільки суб'єктів впливає на навколишнє середовище в певний момент), дискреція (якщо кількість можливих дій у середовищі є кінцевою), епізодичність (якщо дії агента в певні періоди часу впливають на інші періоди) і розмірність (якщо просторові характеристики є важливими факторами середовища та агент розглядає простір у прийнятті рішень). Дії агентів, як правило, опосередковуються через відповідне проміжне програмне забезпечення, яке пропонує абстракцію дизайну для мультиагентних систем, що надає засоби контролю доступу до ресурсів та координації агентів [2].

Агенти в мультиагентній системі мають кілька важливих характеристик:

- автономність – агенти принаймні частково незалежні, самосвідомі, автономні;
- обмеженість знань – жоден агент не володіє інформацією про всю систему, або система є надто складною для використання агентом таких знань;
- децентралізація – в системі не має існувати жодного агента, який керував би всією системою.

Основною причиною використання агентних технологій в Інтернеті речей є те, що агенти можуть працювати без втручання людини. Мобільні агенти за потреби можуть мігрувати між різними пристроями. Агенти можуть спілкуватися і як

наслідок – обмінюватися інформацією. Тому в певних ситуаціях щойно створений агент може взяти цінну інформацію з іншого агента.

Одна з ключових ідей IoT полягає в тому, щоб забезпечити підключення об'єктів у будь-який час, у будь-якому місці, з будь-якими іншими «речами», в ідеалі, з використанням будь-якої технології зв'язку та будь-якої служби. Застосування IoT є численними та різноманітними, поширюються майже на всі сфери повсякденного життя та діяльність окремих осіб і організацій. Система самоорганізації має бути спроможною впоратися з мінливістю в системі IoT, зокрема з динамічними змінами в її середовищі, всередині пристроїв, з яких складається система (наприклад, із дефіцитом ресурсів), або в глобальній мережі, яка включає додаток IoT (наприклад, збій одного з вузлів). Тому однією з найважливіших проблем, пов'язаних із розробкою програм IoT, є ефективне управління цими змінами автономно, незважаючи на їх різноманітність та складність [5].

Розподіленим та мінливим характером систем IoT можна керувати за допомогою автономності, контекстно-осмисленості та самоадаптаційних властивостей програмних агентів, що підтримують розробку автономних застосунків IoT. Найбільш відповідним рішенням є використання агентів як будівельних блоків програми IoT, що забезпечить функціональність розподіленої програми та виконання завдань самоконтролю. Автономність, властива програмним агентам, робить їх слухним вибором для розробки самокерованих систем IoT [1].

3. Проектування платформи «Розумного Університету»

Агентні технології мають необхідні властивості для управління розподіленістю та багатьма іншими вимогами IoT, однак наявні агентні платформи не зовсім вирішують неоднорідність цих середовищ, а також не надають явної підтримки для поширення даних у групі пов'язаних IoT вузлів. Наявні агентні платформи не підтримують гнучкі інфраструктури, де була б можливість модифікувати та розширити інфраструктуру як на етапі розгортання, так і під час роботи застосунку. Далі в роботі буде описано агентну платформу, що частково вирішує наведені вище обмеження та недоліки наявних агентних платформ. Платформа також дає змогу агентам самостійно розв'язувати проблеми зв'язку, використовуючи відповідний плагін. Для наочності буде продемонстровано роботу та переваги платформи

на декількох сценаріях, що стосуються контролю навчального процесу факультету, та показано її користь з погляду реконфігурації та обміну даними через неоднорідні бездротові технології, що дуже важливо для IoT.

Розроблювана платформа є розширенням, конкретною реалізацією абстрактної архітектури FIPA (The Foundation for Intelligent Physical Agents). Крім того, згідно зі специфікаціями FIPA, розробник архітектури сам обирає, які абстрактні елементи реалізовувати і як це робити. І навпаки, реалізація може містити додаткові опції або функції, які вимагають від розробника обробки як абстрактних, так і специфічних для платформи елементів. Тобто реалізація абстрактної архітектури FIPA не забороняє введення корисних елементів для створення хорошої агентної системи, вона просто встановлює мінімально необхідні елементи.

Спочатку уточнимо, що ми розумітимемо під терміном «Розумний Університет». Нам потрібно ввести до розгляду значну кількість датчиків, які можуть забезпечити корисну інформацію про середовище та стан. Крім цього, працівники університету (викладачі, співробітники служби безпеки та спеціалісти деканатів та кафедр) носять особисті кишенькові пристрої (смартфони), що є важливим джерелом даних і можуть використовуватися у наданні конкретних контекстно-залежних послуг. Датчики надають дані, які можуть використовуватися допоміжними сервісами, які допомагають адміністрації і співробітникам служби безпеки в їхній роботі, а також для надання послуг, відповідно до місця розташування студентів.

Система має надавати різні послуги для цільових користувачів:

- для адміністрації система надає інформацію про присутність або відсутність студентів та викладачів в аудиторіях, доступність аудиторій, а також допомагає ділитися і поширювати інформацію між викладачами та студентами;
- для співробітників служби безпеки, ноди IoT (датчики і смартфони) надають інформацію про присутність людей і фізичні показники у приміщенні (температура, світло, вологість тощо) і відправляють глобальні повідомлення різним групам людей, залученим у навчальному процесі;
- у випадку зі студентами ми спираємося на той факт, що абсолютна більшість людей використовує смартфони, тому наша система миттєвих повідомлень може надавати різні види інформації, основаної на місцезнаходженні.

Платформу розроблено як мультиагентну систему (MAS), агенти якої вбудовані в пристрої (наприклад, планшети, смартфони) учасників навчального процесу та датчики, розташовані всередині будівлі. Мультиагентна система має чотири типи агентів:

- агент TeacherAgent для викладачів;
- агент SecurityAgent, який допомагає співробітникам служби безпеки;
- агент StudentAgent для студентів;
- SensorAgent для датчиків.

Кожен із них працює в певному пристрої. Ці агенти розгортаються в агентному середовищі, яке працює на комп'ютері. Такий набір дає нам різні сценарії, які розглянемо нижче.

Опишемо основні сценарії комунікації, які допоможуть проілюструвати проблеми, пов'язані з використанням агентних технологій в IoT:

1. Неоднорідність пристроїв. Нам потрібні інтелектуальні агенти, здатні працювати на будь-якому пристрої IoT, – від простих датчиків до смартфонів. Крім того, необхідно мати агентів, що легко налаштовуються і здатні адаптуватися до апаратних та програмних ресурсів, доступних у кожному пристрої.

2. Гетерогенність спілкування. У неоднорідному середовищі IoT необхідно забезпечити технологічні засоби, що дають змогу спілкуватися агентам, які не підтримують однакові транспортні протоколи.

3. Гнучка комунікаційна інфраструктура. Нині повсякчас з'являються та зникають нові системи й технології бездротового зв'язку. В останні роки ми бачили, як деякі з них вже застаріли (як от IrDa), натомість інші з'явилися (наприклад, NFC і Wi-Fi Direct). Тим не менш, досить складно розширювати наявні агентні платформи для підтримки нових можливостей. Більш гнучкі архітектури та інфраструктури мають допомагати агентним платформам легше впоратися з еволюцією і появою нових технологій, зокрема з новими транспортними сервісами, кодуванням повідомлень і т.д., і забезпечувати сумісність і інтеграцію нового та наявного агентного програмного забезпечення, а також нових комунікаційних технологій. Тому бажано мати агентну архітектуру та інфраструктуру з ефективними механізмами інтеграції. Вона повинна мати можливість легко розширюватися новими засобами комунікації, зокрема вмикати або вимикати ці механізми навіть під час виконання і використовувати їх одночасно, коли це необхідно.

4. Ефективне розповсюдження групових повідомлень. Багато програм і служб, що реалізуються в IoT, вимагають поширення даних між згрупованими вузлами. На практиці реалізацію та управління цим типом комунікації «один до одно-

го» мають здійснювати самі агенти. Кожен агент має керувати та вести список вузлів IoT, зацікавлених у певних даних, і поширювати через окремі повідомлення нові дані, коли вони доступні. Це рішення може бути дуже ресурсоємним, якщо кількість зацікавлених цільових агентів є високою, і також ускладнює реалізацію агента.

4. Сервіси платформи «Розумного Університету»

Як уже було сказано вище, розроблювана платформа розширює абстрактну архітектуру FIPA. Зокрема, платформа підтримує службу управління агентами (AMS), службу реєстрації та виявлення послуг (DF), новий сервіс для реєстрації учасників у групі (Group Management Service, GMS) і сервіс передання повідомлень (Message Transport Service, MTS), який надає можливість комунікації агентам, що зареєстровані в агентній платформі, розширеній для можливості групової розсилки повідомлень. Розповсюдження повідомлень всередині підтримується внутрішньою платформою передання повідомлень (Internal Platform Message Transport, IPMT), що реалізує MTS і вирішує проблеми взаємодії та підтримує кілька протоколів та технологій доступу до мережі для передання повідомлень.

Варто зазначити, що AMS, DF та MTS є класичними й обов'язковими сервісами, що надаються будь-якими агентними платформами (ці послуги визначаються специфікацією абстрактної архітектури FIPA). До того ж до цих послуг ми розширили MTS, щоб підтримувати ефективно групове спілкування, яке доповнюється новим сервісом GMS.

Інформація про агентів та послуги зберігається у внутрішніх базах даних платформи згідно з рекомендаціями FIPA. Таким чином, AMS зберігає агенти у вигляді кортежів, які включають ідентифікатор агента на платформі, і інформацію для контакту з агентом. Групи зберігаються як спеціальні типи агентів.

5. Внутрішній дизайн платформи

Внутрішній дизайн платформи «Розумного Університету» зображено на рис. 1. Платформу розроблено за допомогою патерну Медіатор. Основний клас IU (Intellectual University) інкапсулює взаємодію різних елементів, з яких складається внутрішній дизайн платформи, надання послуг, перелічених вище, і зберігає інформацію про агентів і групи, що зареєстровані всередині платформи, а також про послуги, що

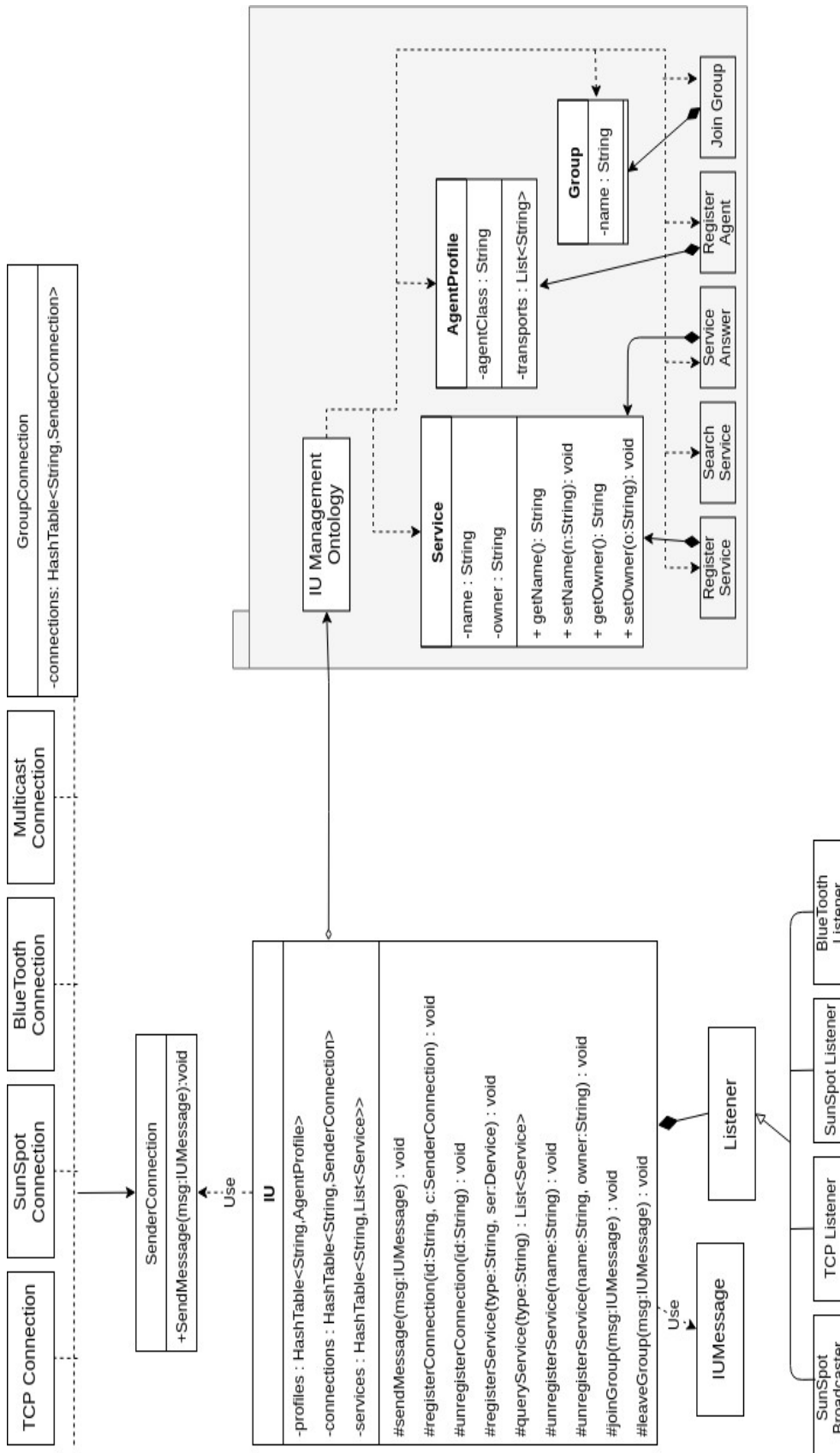


Рис. 1. UML діаграма класів платформи «Розумний Університет»

надаються агентами, які зареєстровані в мультиагентній системі. Агенти не взаємодіють безпосередньо з класом IU для доступу сервісів, що вони надають. Замість цього, всі взаємодії всередині платформи відбуваються за допомогою мови комунікації агентів (Agent Communication Language, ACL), яка є стандартним форматом повідомлень для агентних платформ FIPA. Нашою імплементацією цього стандарту є IUMessage (див. рис. 1), форматом повідомлення якого є звичайна стрічка.

Як тільки агент починає свою роботу, його перша взаємодія полягає в тому, щоб приєднатися до агентної платформи. У реєстраційних запитках присутні слухачі. Платформа надає певних слухачів для різних протоколів і технологій (TCP, SunSpot, Bluetooth Listener). Зв'язок між слухачами та класом IU моделюється за допомогою патерну Спостерігач. Предмети є слухачами, а IU – спостерігачем. Усередині ці класи створюють екземпляри потоків із сокетами, які слухають запити з портативних пристроїв (пристрої з підтримкою Android, мобільні телефони з профілем MIDP). Агенти надсилають запит про реєстрацію в агентній платформі через цей слухач. У реєстраційному повідомленні вони вказують свій тип, ідентифікатор і сукупність і тип транспортних протоколів, які він підтримує (наприклад, Bluetooth). Ця інформація зберігається в екземплярі класу AgentProfile.

Профілі агентів зберігаються в хеш-таблиці (атрибут *profiles*), індексовані за ідентифікатором агента. Ці дані будуть використані для надання агенту доступу до Системи Обміну Повідомленнями. Крім того, платформа додає кожного агента до групи згідно з типом агента. Реєстрацію повинен пройти кожен агент (TeacherAgent, SecurityAgent, SensorAgent та StudentAgent).

У системі кожний агент реєструє послуги, які він може надати в системі. Ми проілюструємо, як два невідомі агенти, зареєстровані у платформі, можуть зустрітися та співпрацювати, використовуючи сервіси платформи. Ця функція полегшує взаємодію агентів у відкритій системі. Це можна продемонструвати за допомогою такого сценарію.

Сценарій 1. У нашому університеті в аудиторії є різні типи датчиків з інтегрованим SensorAgent, які можуть вимірювати та надавати дані про прискорення, яскравість, присутність людей та температуру. Робочий день майже закінчився, тому агент служби безпеки повинен закрити цю аудиторію, але перш ніж це зробити, він має перевірити, чи порожня аудиторія. Ця дія вимагає моніторингу сенсора присутності людей.

В IUManagementOntology та сама структура онтологій, що використовуються у платформі Jade. Ця онтологія містить терміни і поняття (тобто структуровані об'єкти з даними), що використовуються агентами, коли їм потрібен певний сервіс, що надається платформою. Наприклад, коли агент хоче зареєструватися в системі щоб надавати певні дані про середовище в аудиторії, він посилає повідомлення із запитом на сервіс DF, яке включає RegisterService у вмісті повідомлення з описом послуги, яку він надає. З іншого боку, коли SecurityAgent звертається до DF, щоб визначити ідентифікатор агента, який надає інформацію з датчика присутності в аудиторії, йому необхідно відправити повідомлення SearchService та певну інформацію про сервіс, який йому потрібен. Потім IU відповідає на запит повідомленням, яке включає результат ServiceAnswer. Маючи інформацію з цього повідомлення SecurityAgent надсилає запит даних до одного з SensorAgents, включеного в повідомленні, який періодично відправляє інформаційні повідомлення з даними про присутність, поки він не виявить, що університет повністю порожній і може бути закритий.

6. Керування групами в системі «Розумний Університет»

Як зазначено вище, іншою перевагою системи «Розумний Університет» є підтримка різних парадигм і технологій комунікації. Зокрема, IU підтримує однорангове та багатоадресне спілкування, що полегшує розсилання інформації згрупованими компонентами системи. Щоб додати цей тип зв'язку в агентну систему, розроблювана платформа включає концепцію груп. Група складається з набору агентів, які розділяють можливість і є частиною запрограмованої функції агента. За замовчуванням існує група для кожного типу агента, з яких складається система, але також користувач може визначити свої власні групи, беручи до уваги роль, яку відіграє агент у системі, і потреби програми. Різні групи, що формуються в університеті, проілюстровані в такому сценарії.

Сценарій 2. В університеті є групи для кожної категорії персоналу: група викладачів та адміністрації, група, яка об'єднує охоронців, студенти також організовані в різні групи. Не розподілені студенти (тобто які щойно вступили) належать лише глобальній групі студентів, але студенти організованих груп (наприклад, група другого року навчання магістерської програми КНІТ) також утворюють окрему групу з відповідним

керівництвом, який складається з викладача (наприклад, керівника програми) та ряд агентів студентів.

Групи не формуються спонтанно, однак треба мати на увазі, що належність до групи може бути динамічною, вона може змінюватись залежно від часу та контексту (відрахування студента, переведення в іншу групу тощо). Бувають ситуації, що потрібно сформувати тимчасову групу для зведених лекцій або з огляду на наявність вибірко-вих курсів, на які можуть бути записані студенти з різних напрямів і навіть факультетів. Тому групи можна створювати в будь-який час і також можуть враховувати контекстну характеристику (наприклад, локалізацію). Це може бути дуже корисно, іноді система має передавати контекстно-залежну інформацію на певний набір пристроїв. Так простіше розсилати контекстно-залежну інформацію, актуальну для відвідувачів та персоналу, наприклад, «університет скоро зачиняється» або «необхідно допомогти перенести книги в бібліотеку».

Сценарій 3. Група може бути визначена для сприяння поширенню інформації між агентами та підвищення ефективності надання послуг. В університеті є група для всіх датчиків, які контролюють певну аудиторію. Хоча ці агенти контролюють і видають різні дані (наприклад, присутність, температуру або вологість), фізично вони розташовані в тому самому приміщенні, що вважається однією з контекстних ознак, використовуваних для визначення групи. Крім того, вони належать до однієї групи, оскільки всі вони є агентом сенсора. Таким чином, коли певна аудиторія порожня, агент SecurityAgent може відправити повідомлення «порожнє приміщення» на групу датчиків у цій кімнаті, і кожен агент датчика може зменшити активність для економії енергії.

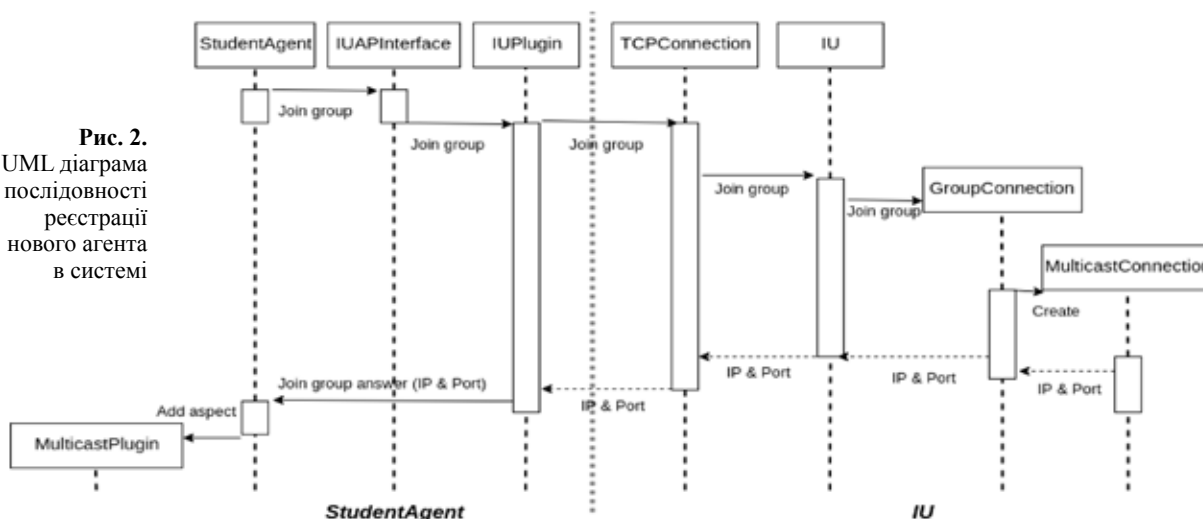
Сценарій 4. Студент зареєстрований у групах, визначених у сценарії 2, тому під час перебування в університеті він отримує кілька групо-

вих повідомлень різних типів та з різних джерел. Викладач надсилає повідомлення про скасування чи перенесення пари, або будь-який інший тип повідомлення, який може допомогти зробити навчальний процес більш організованим та зручним. Охоронець попереджає про підвищення температури в аудиторії або будь-які інші новини, пов'язані з особистою безпекою студентів.

Зі сценаріїв 3 та 4 ми бачимо велику користь від створення груп у нашій системі і як це полегшує спілкування між агентами. Створення та підтримку груп забезпечує система керування групами (GMS). Ця служба підтримує створення груп і дає змогу агентам приєднуватися та виходити з груп.

У середині керування групами відбувається таким чином: якщо агент є першим членом групи, створюється нова група, представлена екземпляром GroupConnection у GMS. Ця структура даних зберігає інформацію про активних учасників групи в атрибуті *connection*. Приєднання агента до групи та інформація, яка зберігається у відповідній GroupConnection, залежить від його профілю (клас AgentProfile на рис. 1). Якщо агент підтримує TCP/IP і мультикаст IP, створюється MulticastConnection (навіть якщо в цій групі немає MulticastConnection) і додається до GroupConnection. MulticastConnection присвоюється мультикаст IP-адреса. Після цього платформа відправляє повідомлення агентам, які підписані на мультикаст адресу і порт, прив'язаний до групи. Якщо агент не підтримує TCP/IP, то індивідуальна транспортна адреса агента (представлений реалізацією інтерфейсу SenderConnection) додається в об'єкт групового підключення. На рисунку нижче UML діаграма, яка відображає процес реєстрації першого StudentAgent, який приєднується до групи. У цьому випадку агент підтримує багатоадресне розсилання, тому процес реєстрації витягує IP та порт нового багатоадресного з'єднання всередині GroupConnection.

Рис. 2. UML діаграма послідовності реєстрації нового агента в системі



Сценарій 5. Реєстрація відвідування студентами та викладачами університету. Оскільки за допомогою Bluetooth та інших датчиків система може реєструвати розташування смартфона (а, відповідно, і його власника) в будь-який час на території університету, дані про розташування кожного з учасників навчального процесу можна додатково зберігати в базі даних. У такому випадку для викладачів та для адміністрації більше немає потреби фіксувати відвідування у традиційний спосіб (на папері), дані про присутність чи відсутність студента завжди можна перевірити в системі. Також це допоможе адміністрації університету контролювати викладачів – чи перебувають вони там, де за розкладом мають проводити пару.

Також за допомогою поєднання даних системи фіксації розташування та датчиків присутності можна фіксувати, що у коридорі біля аудиторії зібрався натовп під час пари. Це може свідчити про те, що викладач запізнюється або взагалі не прийшов на пару. Про це можна сповістити службу охорони, щоб вони відчинили аудиторію, або адміністрацію університету.

Сценарій 6. Оскільки студенти та викладачі поділені на групи, система може зручно та гнучко розсилати сповіщення та нагадування за розкладом як окремим групам, так і всім зареєстрованим агентам. Наприклад, нагадування про необхідність запису на вибіркові дисципліни або запрошення студентам пройти анонімне опитування про викладачів наприкінці семестру.

7. Підтримка групової розсилки повідомлень

Як і в будь-якій іншій агентній платформі, у платформі «Розумний Університет» існує сервіс доставки повідомлень між зареєстрованими в системі агентами. Для кожного з агентів система надає доступ принаймні до одного типу підключення до платформи. Після реєстрації агента через систему керування агентами встановлюється зв'язок між агентом і платформою, вико-

ристовуючи доступ до технології, яка була визначена на етапі реєстрації. З'єднання підтримуються конкретними реалізаціями інтерфейсу `SenderConnection`. Усі активні з'єднання зберігаються у хеш-таблиці. Для кожного з'єднання зберігається інформація – ідентифікатор агента або групи в системі і клас, який реалізує інтерфейс `SenderConnection`. Платформа підтримує п'ять типів з'єднань (тобто реалізацій інтерфейсу): `TCPConnection`, `BluetoothConnection`, `MulticastConnection`, `SunSpotConnection` та `GroupConnection`.

`TCPConnection` і `BluetoothConnection` використовують агенти, що працюють на портативних пристроях, для надсилання та отримання повідомлень через TCP-сокети та з'єднання Bluetooth. Ці пристрої можуть отримувати багатоадресні повідомлення за допомогою сокетів UDP через об'єкти `MulticastConnection`. `SunSpotConnection` використовують для зв'язку з нодами сенсорів `SunSPOT`. Нарешті, `GroupConnection` являє собою групу пристроїв і зберігає внутрішній список з'єднань (атрибут `connection`), який може посилатися на інші описані типи з'єднань. Завдяки такому дизайну легко додати нові пристрої та протоколи зв'язку до сервісу передання повідомлень, оскільки нам потрібно лише імплементувати одного слухача та інтерфейс `SenderConnection` для нового типу зв'язку або специфічний механізм зв'язку. Описані можливості вирішують поставлену вище проблему забезпечення гетерогенності спілкування.

Імплементацію кожного з аспектів поширення повідомлень у платформі реалізує інтерфейс `FIPAAgentPlatform`. Цей інтерфейс дає змогу здійснювати безперешкодний доступ між усіма учасниками платформи. Клас `IUAPInterface` розширює інтерфейс `FIPAAgentPlatform` сервісами, що дозволяють приєднуватися до груп і покидати їх.

Усередині реалізація комунікації розділена на дві основні частини: доступ до послуг платформи IU (служб AMS, DF та GMS), а також надсилання та отримання повідомлень через сервіс передавання повідомлень (`IUPlugin`, `MulticastPlugin` та `BluetoothPlugin`).

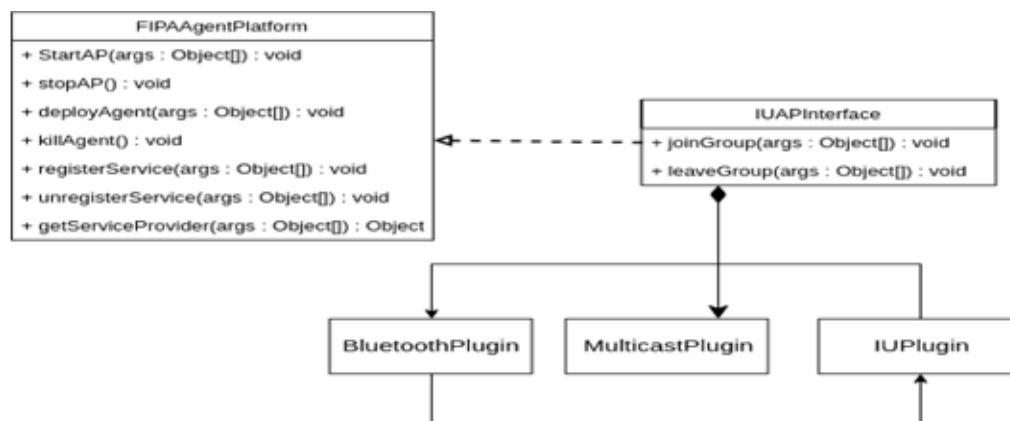


Рис. 3.
UML діаграма класів IUPlugin

Класи IUPlugin і BluetoothPlugin дають змогу надсилати повідомлення всередині платформи «Розумний Університет», використовуючи певну технологію транспортування або доступу. Вони реалізуються як потоки, які слухають повідомлення. У випадку з IUPlugin – з TCP з'єднання, встановленого за допомогою IU, а у випадку з BluetoothPlugin – від RFCOMM Bluetooth з'єднання до служби, яка також працює в агентній платформі. Випадок MulticastPlugin є особливим, тому що це потік лише для прослуховування повідомлень, орієнтованих на групові розсилки. Як зазначено, агенти можуть попросити платформу про приднання до групи, і це робиться через IUAPIInterface. Якщо підключення здійснюється через протокол IP, то платформа відповідає на запит IP-адресою багатоадресної розсилки та транспортним портом. За допомогою цієї інформації створюється новий MulticastPlugin і додається до архітектури.

Як уже згадано, розсилання повідомлень групам також реалізується як інший тип зв'язку, представлений класом GroupConnection.

Екземпляр GroupConnection має всередині хеш-таблицю реалізацій SenderConnection. Причина такого рішення полягає в тому, що, хоч найкращим способом надсилання групових повідомлень є використання мультикаст IP-адреси, ми не можемо бути впевненими, що всі пристрої в групі підтримують багатоадресні TCP/IP або UDP. Таким чином, хоча платформа визначає адресу багатоадресної розсилки для ідентифікації кожної групи і використовує сокети UDP для надсилання багатоадресних повідомлень, вона також гарантує, що члени груп, які не підтримують IP-мультикаст, також можуть отримувати групові повідомлення. Наведена нижче UML діаграма послідовності демонструє це на прикладі надсилання повідомлення від викладача до студентів, що використовують різні типи підключення. Тут агент надсилає повідомлення через BluetoothPlugin. Коли повідомлення надходить до IU, він далі відправляє повідомлення за допомогою відповідного GroupConnection.

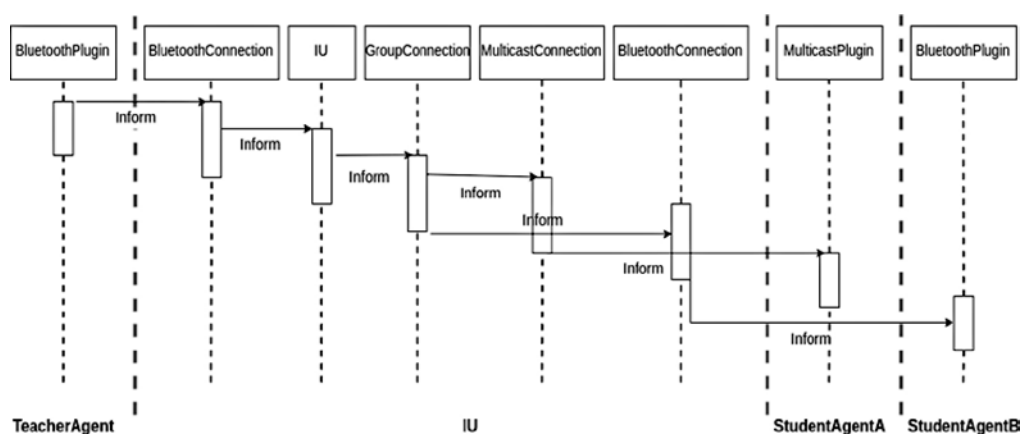


Рис. 4. Діаграма послідовності відправки повідомлення від викладача до студентів

Платформа підтримує протоколи зв'язку кожного пристрою (наприклад, ZigBee, WiFi) і виступає як шлюз, виконуючи певні функції для забезпечення сумісності. На рівні агентів ми надаємо агентам можливість самоконфігурації своєї внутрішньої архітектури, залежно від потреб для використання різних комунікаційних протоколів з урахуванням контексту та потреб програми. Ця гнучкість у рамках внутрішнього дизайну агентів також полегшує одночасне використання різних механізмів поширення повідомлень.

Висновки

У теоретичній частині роботи розглянуто базові положення концепції Інтернету речей та вимоги, що висувають до систем IoT. Описано

основні можливості та переваги використання програмних агентів та мультиагентних систем у реалізації різних IoT.

У практичній частині запропоновано архітектурне рішення на основі мультиагентної системи для підтримки та управління комунікацією у мережі IoT обладнання.

Наведене рішення здатне розв'язати проблеми неоднорідності пристроїв та комунікації, побудови гнучкої інфраструктури комунікації та інфраструктури групової комунікації в агентному середовищі, а також на рівні агента. Було запропоновано агентну платформу, що полегшує комунікацію агентів, які працюють на різних пристроях (датчики SunSpot, смартфони) і використовують різні комунікаційні протоколи,

а також надає можливість формувати агентів у групи для ефективного доставлення групових повідомлень.

Продемонстровано переваги такого рішення на прикладі реалізації «Розумного Університету»

(системи підтримки та управління навчальним процесом, що складається з різних типів датчиків та персональних смартфонів учасників навчального процесу) за використання розробленої мультиагентної платформи.

Список літератури

1. Гороховський С. С. Агентні технології: спроба критичного огляду [Електронний ресурс] / Семен Самуїлович Гороховський. – Режим доступу: http://ekmair.ukma.edu.ua/bitstream/handle/123456789/10630/Horokhovs%27kyy_Ahentni_tekhnolohiyi_sproba.pdf. – Назва з екрана.
2. Anas M Mzahn. Agents of Things (AoT): An Intelligent Operational Concept of the Internet of Things (IoT) [Electronic resource] / Anas M Mzahn, Alicia Y.C. Tang, Mohd Sharifuddin Ahmad. – Mode of access: <https://ieeexplore.ieee.org/document/6920728/>. – Title from the screen.
3. A Software Product Line Process to Develop Agents for the IoT [Electronic resource] / Inmaculada Ayala, Mercedes Amor, Lidia Fuentes, José M. Troya. – Mode of access: <https://www.ncbi.nlm.nih.gov/pubmed/26140350>. – Title from the screen.
4. McEwen A. Designing the Internet of Things / A. McEwen, H. Cassimally. – London : TJ International, 2013. – 324 c.
5. RFC 7452 Architectural Considerations in Smart Object Networking [Electronic resource]. – Mode of access: <https://tools.ietf.org/html/rfc7452>. – Title from the screen.
6. The Internet of Think Overview [Electronic resource]. – Mode of access: <https://www.internetsociety.org/resources/doc/2015/iot-overview>. – Title from the screen.
7. Y.2060 : Overview of the Internet of things Electronic resource]. – Mode of access: <https://www.itu.int/rec/T-REC-Y.2060-201206-I>. – Title from the screen.

References

- Anas M Mzahn, Tang, Alicia Y. C., & Mohd, Sharifuddin, Ahmad. *Agents of Things (AoT): An Intelligent Operational Concept of the Internet of Things (IoT)*. Retrieved from <https://ieeexplore.ieee.org/document/6920728/>.
- Ayala, Inmaculada, Amor, Mercedes, Fuentes, Lidia, & Troya, José M. *A Software Product Line Process to Develop Agents for the IoT*. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/26140350>.
- Gorokhovskiy S. *Ahentni tekhnolohii: sproba krytychnoho ohliadu*. Retrieved from <http://ekmair.ukma.edu.ua/bitstream/handle/123456789/10630/Horokhovs%27kyyHYPERLINK> “http://ekmair.ukma.edu.ua/bitstream/handle/123456789/10630/Horokhovs%27kyy_Ahentni_tekhnolohiyi_sproba.pdf”_Ahentni_tekhnolohiyi_sproba.pdf.
- McEwen, A., & Cassimally, H. (2013). *Designing the Internet of Things*. London: TJ International.
- RFC 7452 Architectural Considerations in Smart Object Networking. Retrieved from <https://tools.ietf.org/html/rfc7452>.
- The Internet of Think Overview. Retrieved from <https://www.internetsociety.org/resources/doc/2015/iot-overview>.
- Y.2060: Overview of the Internet of things. Retrieved from <https://www.itu.int/rec/T-REC-Y.2060-201206-I>.

S. Gorokhovskiy, O. Yamkoviy, O. Kyriienko, O. Pyechkurova

AGENT-BASED IOT PLATFORM FOR CONTROLLING THE LEARNING PROCESS

The large-scale implementation of the IoT promises to change many aspects of our lives for the better. The IoT technology provides the opportunity to improve the agriculture, industry, and the production and distribution of energy by increasing the availability of information on the production chain with the help of network sensors. However, IoT raises many issues that need to be addressed and resolved in order to ultimately realize potential benefits.

The theoretical part of the paper considers the basic provisions of the concept of the Internet of Things, and describes the requirements for the IOT systems. Here are the main features and benefits of using software agents and multi-agent systems in the implementation of various IOT. The practical part offers an architectural solution based on a multi-agent system for maintaining and managing communications in the IOT equipment network.

This solution can solve the problems of device heterogeneity and communication, building a flexible communications infrastructure, and building a group communication infrastructure in the agent environment, as well as at the agent level. To solve the above problems, an agent platform was proposed that facilitates the communication of agents working on various devices (SunSpot sensors, smartphones), uses various communication protocols, and provides the ability to form agents in groups for efficient delivery of group messages.

The advantages of such a solution are demonstrated on the example of the implementation of the “Intelligent University” (a support system and management of educational process consisting of different types of sensors and personal smartphones of participants of the educational process) for the use of the developed multi-agent platform.

Keywords: Internet of Things (IoT), communication models, multiagent system, software agent.