

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**  
**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**  
**Кафедра информационных технологий**

**ДОКЛАД**  
**по дисциплине «Агентная парадигма программирования»**  
**на тему «АНАЛИЗ АГЕНТНЫХ ПЛАТФОРМ»**

Работу выполнил студент группы 209

Брезицкий С.О.

Работу принял доц. каф. ИТ

Полетайкин А.Н.

Краснодар-2025

## **1 Введение**

SPADE (Smart Python Agent Development Environment) — открытая платформа для создания мультиагентных систем (МАС), разработанная на языке Python и использующая протокол XMPP для обмена сообщениями.

## **2 Общие свойства платформы SPADE**

### **2.1 Область применения**

SPADE применяется в различных областях, включая:

- Моделирование и симуляция: используется для создания симуляторов виртуальных рынков и других экономических моделей.
- Интернет вещей (IoT): обеспечивает взаимодействие агентов с IoT-устройствами через концепцию "артефактов".
- Образование и исследовательская деятельность: применяется в учебных курсах и научных проектах для демонстрации принципов МАС.

### **2.2 Технологии, лежащие в основе платформы**

Язык программирования: Python (версия 3.8 и выше).

Протокол обмена сообщениями: XMPP (eXtensible Messaging and Presence Protocol), обеспечивающий стандартизованную и расширяемую коммуникацию между агентами.

Асинхронная архитектура: использование библиотеки asyncio для реализации асинхронного поведения агентов.

Соответствие стандартам: поддержка стандартов FIPA (Foundation for Intelligent Physical Agents), включая использование метаданных и форм данных XMPP (XEP-0004).

#### **2.2.1 XMPP**

XMPP (англ. eXtensible Messaging and Presence Protocol «расширяемый протокол обмена сообщениями и информацией о присутствии») — открытый, основанный на XML, свободный для использования протокол для мгновенного обмена сообщениями и информацией о присутствии (см. список контактов) в режиме, близком к режиму реального времени.

## **2.3 Сообщество разработчиков**

SPADE является проектом с открытым исходным кодом, размещённым на GitHub. Основным разработчиком является Хавьер Паланса (Javier Palanca). Платформа активно используется в академических кругах, особенно в Политехническом Университете Валенсии (UPV), где она была первоначально разработана.

## **2.4 Расширяемость платформы**

SPADE поддерживает расширение функциональности через плагины:

- spade\_bdi: реализация агентов с архитектурой BDI (Belief-Desire-Intention) с использованием языка AgentSpeak.
- spade\_pubsub: поддержка протокола публикации/подписки (PubSub) для обмена сообщениями между агентами.
- spade\_artifact: управление артефактами в среде SPADE, что особенно полезно для интеграции с IoT-устройствами.

## **2.5 Интеграция**

Благодаря использованию протокола XMPP, SPADE обеспечивает лёгкую интеграцию с другими системами и платформами, поддерживающими этот протокол. Это позволяет агентам SPADE взаимодействовать с внешними сервисами и агентами, реализованными на других платформах, таких как JADE.

## **2.6 Документация**

Официальная документация SPADE доступна на ReadTheDocs и включает подробные руководства по установке, настройке и разработке агентов. Документация регулярно обновляется и содержит примеры кода, что облегчает освоение платформы для новых пользователей.

## **2.7 Лицензия**

SPADE распространяется под лицензией MIT, что предоставляет широкие возможности для использования и модификации платформы в различных проектах, включая коммерческие.

### **3 Свойства платформы, связанные с разработкой МАС**

#### **3.1 Поддержка методологии разработки**

SPADE предоставляет инструменты для всех этапов разработки МАС:

- Анализ и проектирование: использование архитектуры BDI и поведения агентов для моделирования сложных систем.
- Реализация: разработка агентов на Python с использованием предоставленных классов и методов.
- Тестирование и отладка: возможность запуска агентов в изолированной среде и мониторинг их поведения через веб-интерфейс.
- Развёртывание: поддержка различных XMPP-серверов и возможность масштабирования системы.

#### **3.2 Наличие инструментария разработки агентов и МАС**

Классы агентов и поведения: предоставление базовых классов для создания агентов и определения их поведения (циклическое, периодическое, одноразовое и т.д.).

Веб-интерфейс: интерактивный интерфейс для мониторинга состояния агентов, их поведения и взаимодействий.

Интеграция с внешними библиотеками: возможность использования сторонних библиотек Python для расширения функциональности агентов.

### **4 Апробация платформы**

На платформе SPADE была создана МАС для 3 лабораторной работы по АПП, а именно было реализовано мультиагентное приложение с реактивной архитектурой для предметной области «Размещение заказов на оборудование».

На рисунке 1 показано взаимодействие агентов через вывод в консоль.

```
[Photon] запущен, предложение={'cost': 5, 'productivity': 120, 'reliability': 9}
[Specdetal] запущен, предложение={'cost': 8, 'productivity': 200, 'reliability': 7}
[Stankostroy] запущен, предложение={'cost': 5, 'productivity': 100, 'reliability': 8}
[Atom] запущен, предложение={'cost': 7, 'productivity': 150, 'reliability': 6}

[firm1] запущен, требования={'cost': 5, 'productivity': 100, 'reliability': 8}
```

```
[firm2] запущен, требования={'cost': 6, 'productivity': 150, 'reliability': 5}
Все агенты запущены
```

```
[firm1] отправка CFP
[firm1] CFP отправлен к photon@localhost
[firm1] CFP отправлен к specdetal@localhost
[firm1] CFP отправлен к stank@localhost
[firm1] CFP отправлен к atom@localhost
[firm1] Начало сбора предложений
```

```
[Photon] CFP получен от firm1@localhost
[Photon] PROPOSE отправлен к firm1@localhost: {'cost': 5, 'productivity': 120, 'reliability': 9}
[Specdetal] CFP получен от firm1@localhost
[Specdetal] Требования не выполнены, PROPOSE не отправлен
[Stankostroy] CFP получен от firm1@localhost
[Stankostroy] PROPOSE отправлен к firm1@localhost: {'cost': 5, 'productivity': 100, 'reliability': 8}
[Atom] CFP получен от firm1@localhost
[Atom] Требования не выполнены, PROPOSE не отправлен
```

```
[firm1] PROPOSE получен от Photon: {'cost': 5, 'productivity': 120, 'reliability': 9}
[firm1] PROPOSE получен от Stankostroy: {'cost': 5, 'productivity': 100, 'reliability': 8}
```

```
[firm1] Сбор завершён, предложений: 2
[firm1] Выбрано: Photon с {'cost': 5, 'productivity': 120, 'reliability': 9}
[firm1] ACCEPT-PROPOSAL отправлен к Photon
[firm1] REJECT-PROPOSAL отправлен к Stankostroy
[firm2] Сбор завершён, предложений: 0
[firm2] Нет подходящих предложений
Все агенты остановлены
```

Рисунок 1 – Взаимодействие агентов

## 5 Примеры реальных проектов

Симулятор виртуальных рынков: разработка симулятора для моделирования поведения агентов в виртуальных экономических системах.

Ссылка: <https://riunet.upv.es/bitstream/handle/10251/185861/Barbas%20-%20Multi-agent%20Generic%20Market%20Simulator%20for%20Virtual%20Markets%20using%20SPADE.pdf>

Интеграция с IoT: создание агентов, взаимодействующих с IoT-устройствами через артефакты, что позволяет моделировать сложные сценарии в умных средах.

Ссылка:

[https://www.researchgate.net/publication/358766828\\_Developing\\_IoT\\_Artifacts\\_in\\_a\\_MAS\\_Platform](https://www.researchgate.net/publication/358766828_Developing_IoT_Artifacts_in_a_MAS_Platform)