

Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Факультет математики, механики и компьютерных наук

Направление подготовки 010400
«Прикладная математика и информатика»

А. А. Тактаров

РЕАКТИВНЫЙ ФРЕЙМВОРК ДЛЯ ОРГАНИЗАЦИИ МУЛЬТИАГЕНТНЫХ
РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ

Магистерская диссертация

Научный руководитель:
старший преподаватель
В. Н. Брагилевский

Рецензент:
доцент, к. ф.-м. н.
С. А. Гуда

Ростов-на-Дону

2014

Содержание

Введение	3
1. Постановка задачи	5
Список литературы	6

Введение

Стремительный рост возможностей технологий беспроводной передачи данных, а также широкое распространение мобильных и встраиваемых устройств способствовали появлению концепции так называемого «Интернета вещей» (англ. «*Internet of Things*»)[1], которая заключается в объединении всех окружающих нас вещей в огромную вычислительную сеть. Участниками (*агентами*) такой сети являются устройства, которые способны собирать информацию о физической среде, обрабатывать ее и реагировать на изменение состояния других агентов и всей системы в целом. Стабильное функционирование такой сети позволит с огромной скоростью внедрять и использовать такие технологии, как «умные» датчики[2], носимые устройства (англ. *wearable devices*), а также системы автоматизированного управления домом. Кроме того, становление «Интернета вещей» влечет за собой появление принципиально новых потоков информации, тщательный анализ которых позволит улучшать существующие системы здравоохранения, безопасности и контролировать состояние окружающей среды.

Однако, создание подобного рода сети невозможно без наличия функционирующей инфраструктуры, которая бы позволила быстро и эффективно интегрировать новые компоненты. Исходя из распределенной природы описываемой сети, сформулируем необходимые для этого требования:

1. Соблюдение принципа системности при разработке[3]. Продукт должен быть представлен в виде целой системы компонентов, каждый из которых обладает определенной функцией. Такие компоненты автоматически становятся автономными участниками сети.
2. Однородность среды. Компоненты сети должны иметь возможность взаимодействовать между собой, используя стандартизированные протоколы и схемы. Задачи идентификации, обеспечения целостности, конфиденциальности передаваемых данных должны по возможно-

сти быть решены этими протоколами.

3. Открытость используемых технологий. Применение как программных, так и аппаратных решений, которые имеют открытую документацию, лояльные условия использования и одновременно поддерживаются разными разработчиками (обычно целым сообществом), позволяет в определенных случаях решить проблему интеграции компонентов и сократить разрыв между разработкой и запуском в производство. Кроме того, открытые платформы предоставляют широкие возможности для начинающих команд разработчиков, что является благоприятным для формирования рынка.

В данной работе описан процесс реализации и интеграции мультиагентной системы на примере задачи распределенной печати фотографий. В рамках разработанной системы устройство, печатающее фотографию, рассматривается как автономный агент, который обладает состоянием и способен принимать и исполнять задания. Принципы, сформулированные выше, были использованы в качестве основополагающих на этапах проектирования и разработки данного продукта.

1. Постановка задачи

Целью работы является разработка и развертывание системы, позволяющей организовать моментальную печать фотографий пользователей социальной сети Instagram, распределяя задания печати среди подключенных к системе агентов – *печатных станций*, в состав которых входит печатное устройство – принтер.

«Здесь краткое описание цикла работы»

Сформулируем основные требования, предъявляемые к системе:

1. Печатные станции могут быть физически отделены друг от друга, кроме того они могут находиться в разных сегментах сети. Необходим способ организации канала связи между агентами и контроль жизнеспособности этого канала.
2. Необходим интерфейс управления печатными станциями и заданиями печати.
3. Система должна иметь минимальный отклик и максимально быстро реагировать на изменение состояния компонентов. Изменение статуса задания печати (печать может завершиться успешно, а может закончиться неудачей в результате обрыва соединения) должно моментально отражаться в интерфейсе управления заданиями.

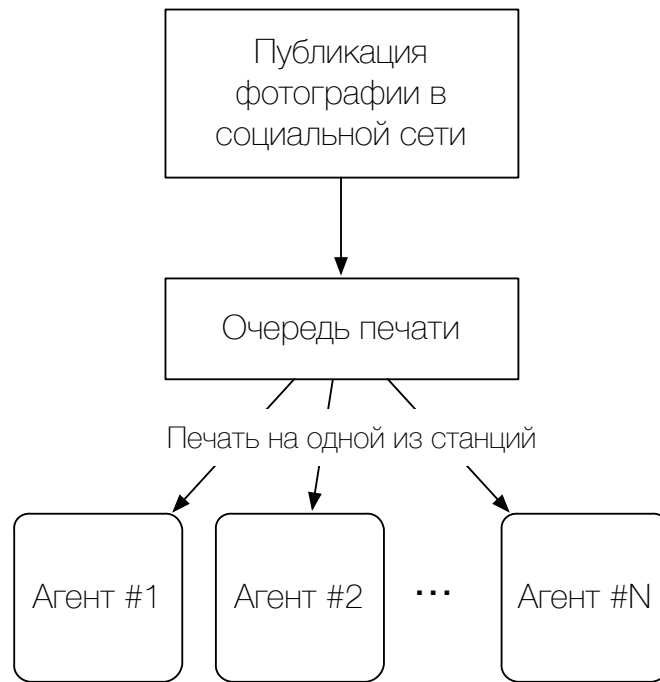


Рис. 1. Архитектура разработанного приложения

Список литературы

1. B. Wasik. In the programmable world, all our objects will act as one. <http://www.wired.com/2013/05/internet-of-things-2/>, 2013.
2. Nest world's first learning thermostat. <https://nest.com/>.
3. М. Слюсаренко, И. Слюсаренко. Системологический подход к декомпозиции в объектно-ориентированном анализе и проектировании программного обеспечения. 2010.