

# ОБЗОР МЕЖДУНАРОДНОЙ ДЕЯТЕЛЬНОСТИ В СФЕРЕ РАЗРАБОТКИ И ИССЛЕДОВАНИЯ МЕТОДОВ ОБРАБОТКИ ДАННЫХ ОТ УСТРОЙСТВ ИНТЕРНЕТА ВЕЩЕЙ

Е. М. Окулов<sup>1</sup>, Р. В. Киричек<sup>1\*</sup>

<sup>1</sup> СПбГУТ, Санкт-Петербург, 193232, Российская Федерация

\* Адрес для переписки: [kirichek@sut.ru](mailto:kirichek@sut.ru)

## **Аннотация**

**Предмет исследования.** Статья посвящена обзору методов обработки данных, поступающих от различных устройств Интернета Вещей. **Метод.** В качестве исследования рассматривается метод анализа различных технологий обработки данных от устройств Интернета вещей. **Основной результат.** В рамках данной работы было показано, что, используя единый язык программирования для различных модулей и сред можно сделать новый расширяемый сервис, который позволит рациональнее использовать ресурсы, в том числе машин с низким уровнем производительности. **Практическая значимость.** На основе данного проекта есть возможно создать временнную сеть сенсоров, состоящую из модулей, для предупреждения случаев стихийных бедствий или чрезвычайных ситуаций. Кроме того, путем распределения функций защиты можно повысить безопасность работы всех задействованных систем.

## **Ключевые слова**

сети связи, беспроводная технология, методы распределенной обработки данных, Интернет Вещей, влияние.

## **Информация о статье**

УДК 004.9

Язык статьи – русский.

Поступила в редакцию 01.11.16, принята к печати 25.11.16.

**Ссылка для цитирования:** Окулов Е. М., Киричек Р. В. Обзор международной деятельности в сфере разработки и исследования методов обработки данных от устройств Интернета Вещей // Информационные технологии и телекоммуникации. 2016. Том 4. № 4. С. 58–66.

# REVIEW OF INTERNATIONAL ACTIVITY IN THE FIELD OF DEVELOPMENT AND RESEARCHING METHODS TO PROCESSING DATA FROM IoT DEVICES

E. Okulov<sup>1</sup>, R. Kirichek<sup>1\*</sup>

<sup>1</sup> SPbSUT, St. Petersburg, 193232, Russian Federation

\* Corresponding author: kirichek@sut.ru

**Abstract—Research subject.** The article is devoted to the review of methods of data processing coming from various IoT devices. **Method.** As a researching method, article shows method of distributed data processing for the next generation of IoT systems. **Core result.** In the framework of this article it was shown that, using a single programming language for various modules and environments, it is possible to obtain a new enhanced service that allows rational use of resources, including machines with low performance. **Practical relevance.** Based on this project, it is possible to create a temporary network of sensors, consisting of modules, to prevent natural disasters or emergencies. In addition, by distributing the protection functions, it is possible to increase the safety of all the systems involved. **Keywords**—Communication networks, wireless technology, methods of distributed data processing, Internet of things, influence, usefull, researching methods.

## Article info

Article in Russian.

Received 01.11.16, accepted 25.11.16.

**For citation:** Okulov E., Kirichek R.: Review of International Activity in the Field of Development and Researching Methods to Processing Data from IoT Devices // Telecom IT. 2016. Vol. 4. Iss. 4. pp. 58–66 (in Russian).

## Введение

Беспроводные сенсорные сети (БСС) являются базовой основой для Интернета вещей. В настоящее время БСС получили бурное развитие, особенно в связи с исследованиями технологий микроэлектронных механических систем (MEMS – *Micro Electro-Mechanical Systems*), которые облегчили разработку и внедрение умных датчиков [1]. Типовая БСС состоит из большого количества беспроводных датчиков, которые осуществляют сбор, хранение, обработку информации, взаимодействуя с соседними узлами. В прошлом датчики соединялись проводами, но с развитием энергоэффективных беспроводных технологий датчики объединяются в самоорганизующиеся сети по беспроводным каналам, что приводит к еще большему удобству реализации и использования таких сетей в дальнейшем.

В настоящее время Интернет вещей сделал качественный рывок, став одной из самых обсуждаемых тем в сфере технологий. Такое растущее признание происходит из-за воздействия Интернета Вещей на сбор и обработку данных, в том числе потенциал, который все еще остается неиспользованным до конца. Каждый день новые машины, датчики, и устройства, подключенные к сети IoT, передают

все больше и больше информации для последующей обработки. Компании внедряют новые идеи в свои IoT проекты, которые позволяют лучше управлять увеличенными объёмами информации. Традиционные базы данных и архитектуры аналитики всегда будут жизненно необходимы, но IoT дает новые возможности по обработке разнообразных данных, потоком передающихся от огромного числа источников. Совокупность данных от IoT-устройств огромна и очень быстро распространяется. Данная статья носит обзорный характер по методам управления, хранения и обработки данных от устройств и приложений Интернета вещей.

### **Текущее состояние дел в области обработки данных от IoT-устройств**

Аналитическая компания Aberdeen Group провела обзор организации, которые собирают, интегрируют и анализируют данные, создаваемые устройствами Интернет вещей. Как показал обзор, IoT компании стремятся эффективно использовать перенасыщение информацией, которые в свою очередь поступают от различных устройств, а также разных систем, для совершенствования алгоритмов обработки данных и улучшения производительности. Aberdeen Group исследовала около 68 IoT компаний, выявив проблемы, которые надо оптимизировать в ближайшее время. Среди проблем были выделены следующие:

- средний объем IoT трафика вырос на 30 % за прошедший год;
- 54 % компаний сообщили, что существующих возможностей для анализа данных недостаточно;
- 50 % IT организаций не удалось сократить время на обработку трафика за прошедший год.

Датчики становятся более доступными чем когда-либо прежде и цены на подключение устройств к Интернет продолжают понижаться [2].

Недавние исследования Cisco и IBM показывают, что пользователи генерируют 2,5 квинтилионов ( $10^{18}$ ) байт в день, которые к 2020 году увеличатся до 40 петабайт ( $40 \cdot 10^{15}$ ), что означает 5,200 гигабайта на каждой человек на земле. Большая часть этих данных будет сгенерирована устройствами и датчиками Интернета вещей (IoT). На сегодняшний день IoT включает в себя миллиарды подключенных к Интернету устройств или вещей, каждый из которых распознается, взаимодействует, вычисляет и приводиться в действие, а также обладает искусственным интеллектом, мульти модульными интерфейсами, физическими/виртуальными идентификационными данными и атрибутами. Подключенные к Интернету могут быть сенсоры, RFIDs, социальные медиа, clickstreams, различные транзакции, приводы (к примеру, оборудование для добычи полезных ископаемых, разведки нефти или производственных операций), лабораторные инструменты (к примеру, физический синхротрон повышенной энергии), а также различные гаджеты (ТВ, смартфоны и т. д.).

Концепция Интернет Вещей позволяет подключаться к вещи в любое время, в любом месте, каждому с любого устройства, в идеале используя любую сеть, технологию и сервис. Такое представление привело к IoT приложениям с большим количеством данных, которые способны обрабатывать миллиарды потоков и десятки лет исторических данных для своевременного принятия решения. Такие приложения должны обрабатывать и управлять потоковыми и многомерными

данными из географически распределённых источников данных, которые доступны в различных форматах, иметь различное местонахождение и надежны на разных уровнях [3].

### **Требования к приложениям IoT с большим количеством потребляемых данных**

Настоящее поколение приложений для больших объемов данных IoT (интеллектуальное управление цепочками поставок в логистике, умные-энергетические сети) сочетают в себе несколько независимых аналитических моделей данных, хранилища данных, а также обработка потоков информации в режиме реального времени, которые доступны через географически распределенные центры обработки данных (как частные, так и общедоступные). Например, интеллектуальное управление цепочками поставок обеспечивает инновационный подход для всей отрасли, что является сложной задачей, поскольку:

- объёмы данных охватывают несколько континентов и независимо друг от друга управляются сотнями поставщиков и дистрибуторов;
- данные обновляются в режиме реального времени на основе информации поступающей от датчиков, прикрепленных на транспортных средствах и контейнерах;
- пользователи оставляют отзывы через социальные медиа, порталы и блоги, что влияет на количество клиентов.

Компании должны объединять и анализировать распределённые данные вместе с обычными факторами, такими как прогноз погоды и ценовые позиции, чтобы установить какие факторы влияют на спрос конкретных продуктов, а затем быстро принимать меры по адаптации в конкурентной и развивающейся среде. По аналогии приложения IoT могут собирать информацию в реальном времени через социальные сети, департаменты здравоохранения, больницы, сайты амбулаторного обслуживания для выявления вспышек смертельных заболеваний. Очевидно, что такие приложения IoT используют большие наборы данных, которые не могут быть переданы через Интернет для обработки на централизованных государственной или частной центрах обработки данных. Основными причинами этого является:

- наборы данных регулируются политикой конфиденциальности, и содержат нормативно – правовые ограничения которые запрещают передачу их за пределы родительский домена;
- потоки данных слишком большие по объёму и слишком быстрые по скорости, чтобы быть обработанными ЦОД, поскольку это может привести к телекоммуникационным сбоям;
- аналитические модели должны быть доступны по всей территории распределения.

Несмотря на требования IoT для приложений по работе с большими данными, сегодняшние технологии и вычислительная инфраструктура очень ограничена, такие приложения могут вычислять и хранить ресурсы в рамках централизованной локальной сети, один кластер в ЦОД. Они не могут обеспечить безопасность и приватность хранящейся информации по обработке гетерогенных данных, управляемых гетерогенной политикой и правилами контроля доступа.

Для управления данными в реальном времени нужны как новые алгоритмы, так и вычислительные средства, основанные на новой аппаратной базе. Таким образом, успешная стратегия работы с данными в инфраструктуре IoT подразумевает:

- системы сбора данных в реальном времени;
- потоковую передачу данных;
- хранение данных;
- концентраторы событий;
- анализ больших данных, в особенности с использованием самообучающихся систем;
- защиту данных, например, путем шифрования;
- управление доступом;
- обнаружение вторжений;
- новые виды вычислительных машин<sup>1</sup>.

### **Облачные вычислительные центры обработки данных**

Существующие технологии обработки большого потока информации, и инфраструктура ЦОД обладают разнообразными возможностями для распределенных систем обработки данных. Одним из таких примеров являются облачные вычислительные центры обработки данных

Коммерческие центры обработки данных такие как Amazon Web Services и Microsoft Azure предоставляют ресурсы по вычислению, хранению и программной поддержке данных в концепции облачного сервиса, работающих по принципу виртуальных стеков. Примеры виртуальных машин для управления являются Eucalyptus и Amazon Elastic Compute Cloud (EC2); утилиты по управлению – Future Grid Image repository; хранилища информации – Google File System (GFS), The Hadoop распределенная файловая система и Amazon Simple Storage Service (S3); а также фреймворки для быстрой обработки данных – Amazon Elastic MapReduce.

С другой стороны, частные data-центры как правило строят базовую инфраструктуру путем объединения доступного ПО, инструментов и услуг. Такое программное обеспечение включает в себя системы управления кластерами, такие как Oscar, и простые утилиты Linux для управления ресурсами (Slurm); системы с параллельным файловым хранилищем таким как SAN/NAS (*storage area network / network-attached storage и Lustre*); а также системы управления данными – Berkeley Storage Manager (BeST-Man) и dCache. В дополнении, некоторое число частных data-центров с поддержкой распространения ресурсов (*sharing*) сеточного распределения, например, Globus tool-kits, Unicore и Lightweight Middleware for Grid Computing (gLite).

### **Метод распределенной обработки данных для следующего поколения IoT систем**

Рассмотрим метод распределенной обработки данных для Интернета Вещей использующий облачные вычисления и множество сенсоров для данных с изоб-

---

<sup>1</sup> Семь основных концепций и четыре шага на пути к IoT. URL: [www.ibm.com/developerworks/ru/library/iot-key-concepts](http://www.ibm.com/developerworks/ru/library/iot-key-concepts)

ражениями. В частности, способ реализации распределенного выполнения алгоритмов обработки данных изображения, в том числе рассмотрим прототип системы. Результатом окажется возможность оценить масштабируемую систему обработки на малых и средних характеристиках аппаратных устройств. Все устройства IoT, использованные в проекте, имеют относительно средние характеристики и недорогие в цене.

В последние годы, сети и сенсорные технологии стали быстро развиваться, и воспринимают все больше данных [4]. Дальнейшие исследования расширяют возможности машинного обучения, включая глубокое которое взаимодействует напрямую с Интернетом Вещей. Еще недавно хватало простой информации, такой как сбор температурных показателей, влажности и ускорения было достаточно. Теперь все большее количество данных (например, изображения и аудиоданные), с помощью ресурсов процессора, требует обработки. Японские разработчики показали среду, где облачный сервер управлял несколькими IoT-устройствами, а также среду для обмена данными между устройствами IoT под OpenCv-Python. В данном проекте был предложен метод для Интернета Вещей, имеющий окружение по функционалистке сопоставимый с PC [5].

Во многих традиционных встраиваемых устройствах ПО реализовано на одном аппаратном средстве и одна задача запускается на независимом устройстве (рис. 1, а).

В соответствии эволюцией технологии LSI, недавно выпущенные устройства Интернет Вещей являются многофункциональными, выполняя функции получения и передачи данных. Тем не менее, из-за ограниченного энергопотребления и низкой стоимости, возможности по обработке данных на IoT-устройствах составляют от  $\frac{1}{4}$  до  $\frac{1}{100}$  от возможностей современных компьютеров. Параллельные вычислительные системы, использующие железо с низкими характеристиками, могут использовать метода высокоскоростной передачи данных, однако необходима программная модель такого алгоритма, которая подразумевает раздельную программную разработку для каждого устройства (рис. 1, б). В большинстве случаев программы для параллельной системы обработки сложны. Кроме того, из-за долгого периода разработки, отладки и высоких затрат, реализация параллельной системы обработки данных была затруднена.

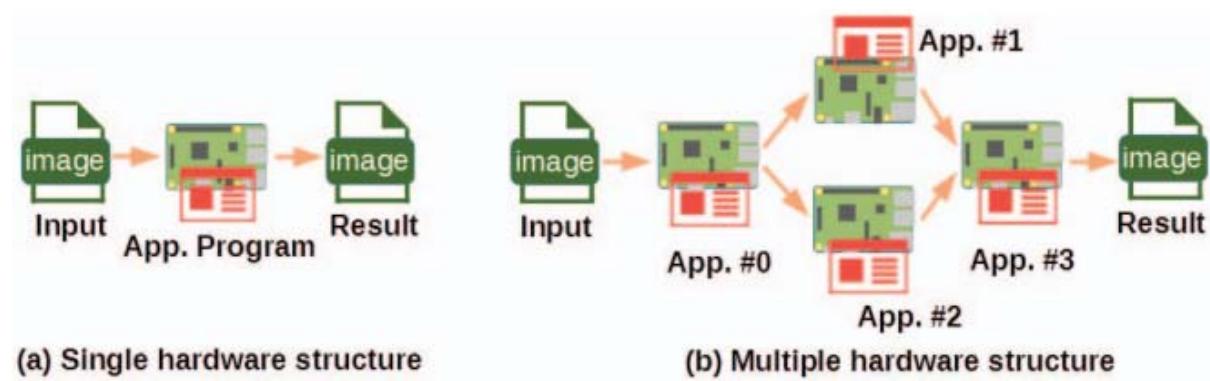


Рис. 1. Однозвенная и многозвенная структура устройств

## Метод распределенный вычислений для IoT устройств

Разработчики с портала IEEE Xplore, Tetsuo Furuichi и Hiroshi Mineno, реализовали вычислительную инфраструктуру для Интернет Вещей, которая выполняется на моно-процессе:

### 1. Единый язык программирования.

Для того чтобы уменьшить сложность разработки, был выбран один компьютерный язык с приоритетом по унификации операционной системы. Во встроенных системах обычно используется язык C/C++. Также был выбран Python для написания скриптов с подключенной библиотекой численных алгоритмов OpenCV с открытым кодом.

### 2. Клиент-серверная модель.

Существует большое количество распределенных процессов вычисления. Была выбрана клиент-серверная модель, которая была развернута на облачном сервисе, в котором устройства Интернет Вещей управляются как клиентские системы.

### 3. Единое управление на удаленном сервере.

Серверная система на облачном сервисе функционирует как удаленный сервер и кроме того непосредственно запускает Python-скрипты, команды каждого клиента, весть этот набор операций в клиентской системе может управляться одним скриптовым сценарием.

### 4. Передача данных между устройствами.

Удаленный сервер может контролировать напрямую выполнение алгоритмов системы. Объекты, до и после выполнения алгоритма обмениваются между клиентами без участия сервера. В частности, клиент отправителя и клиент получателя обмениваются данными напрямую.

### 5. Маршрутизация на основе устройств и сети исполнения.

Перед выполнением операций, сервер маршрутизирует процесс в соответствии с сценарием основанном на вычислительных характеристиках (производительность процессора, объем памяти) и коммуникационными характеристиками между клиентами (задержка, пропускная способность).

Для того чтобы оценить эти основные функции для за исключением того автоматическая маршрутизация Е, мы разработали прототип системы

Был разработан прототип системы для того чтобы оценить основные функции, за исключением автоматической маршрутизации (рис. 2).

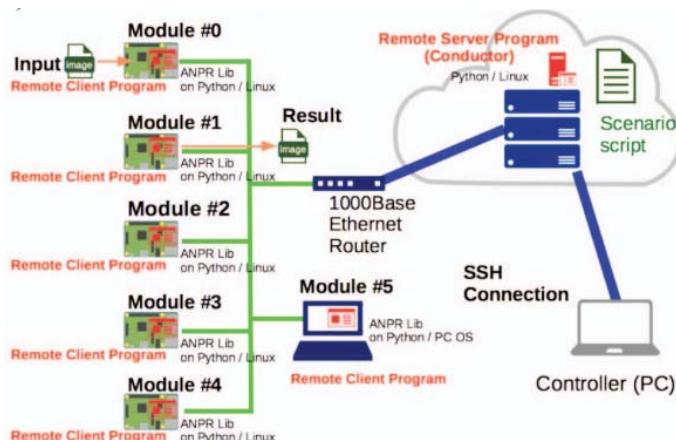


Рис. 2. Прототип системы

Прототип системы включает в себя один VPS сервер (*Virtual Private Server*), 5 встроенных Linux модулей (#0, #1, #2, #3, #4) обладают различной производительностью, и один PC(#5). Также был разработана «Программа доступа к удаленному серверу» на VPS сервере. На клиентской машине развернута «Автоматическая программа по распознаванию номера с фотографии» (ANPR) на основе OpenCV, переписанная на Python и «Программа доступа к удаленному клиенту» которая получает команды от «Программа доступа к удаленному серверу» на VPS сервере.

### Запуск прототипа и результат

В результате, была реализована часть ANPR программы, поддерживающей библиотеку OpenCV написанную на C++ для Python скриптов. Такой алгоритм имеет три стадии (pre, rect, svm), где 'pre' стадия включает пиксельные операции на кадровой памяти, 'rect' стадия включает операции по пиксельному поиску и 'svm' включает операции машинного обучения. Затем, некоторые части, функции 'rect', разделились на подфункции для параллельных вычислений. Разработчики подготовили три распределительные системы с разной скоростью, (S) – низкая скорость, (M) – средняя скорость и (F) высокая скорость. Было просчитано время выполнения с помощью комбинирования различных скоростных режимов на каждой стадии выполнения алгоритма (рис. 3). К примеру, модель "S-[SS]-M" означает что стадия 'pre' выполняется процессором ARM11, 'rect' выполняется двумя модулями ARM11 и 'svm' модулем Cortex-A7. В результате параллельная обработка с использованием нескольких модулей стадия 'rect', обычно занимающая много времени на поиск пикселей, позволила значительно сократить время работы.

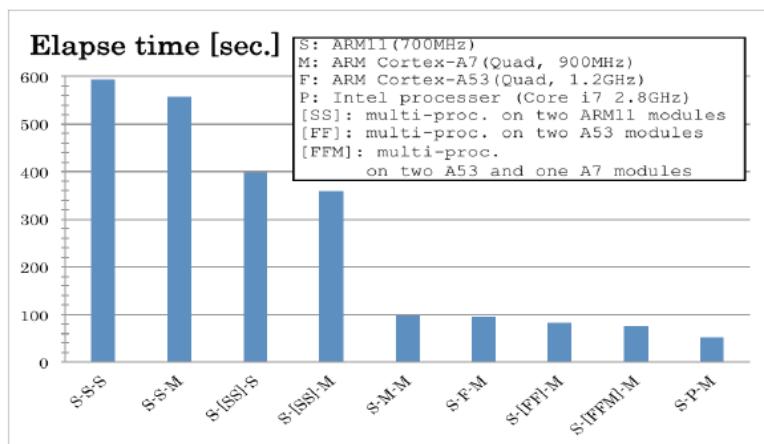


Рис. 3. Результаты испытаний

### Заключение

В данной статье установлено, что на основании использованного прототипа системы, проводящий сервер может запускать распределенные скрипты без специальных распределенных программ для каждого модуля. Кроме того, была улучшена производительность программы распознавания изображений для устройств

IoT с низкой спецификации. Используя единый язык программирования различных модулей и среды, в которой модули тесно связаны друг с другой в Интернете, можно сделать новый расширяемый сервис. На основе данного проекта есть возможно создать временную сеть сенсоров, состоящую из модулей, для предупреждения случаев стихийных бедствий или чрезвычайных ситуаций. Кроме того, путем распределения функция защиты повысит безопасность системы.

## Литература / References

1. Akyildiz I. F., Su W. Wireless Sensor Networks: A Survey // Computer Networks. 2002. Vol. 38. Iss. 4. pp. 393–422.
2. Kirichek R., Golubeva M., Kulik V., Koucheryavy A. The Home Network Traffic Models Investigation // 18<sup>th</sup> International Conference on Advanced Communication Technology (ICACT). 2016. pp. 97–100. DOI: 10.1109/ICACT.2016.7423280.
3. Wang L., Ranjan R. Processing Distributed Internet of Things Data in Clouds // IEEE Cloud Computing. 2015. Vol. 2. Iss. 1. pp. 76–80.
4. Bradley J., Loucks J., Macaulay J., Noronha A. Internet of Everything (IoE) Value Index. Cisco Systems. 2012.
5. Furuchi T. Lightweight Image Sensor Node for Next Generation IoT // IEEE 4<sup>th</sup> Global Conference on Consumer Electronics (GCCE). 2015. pp. 36–37.

**Окулов Евгений Михайлович**

– магистрант, СПбГУТ, Санкт-Петербург, 193232, Российская Федерация, zTiff0@gmail.com

**Киричек Руслан Валентинович**

– кандидат технических наук, доцент, СПбГУТ, Санкт-Петербург, 193232, Российская Федерация, kirichek@sut.ru

**Okulov Evgeniy**

– undergraduate, SPbSUT, St. Petersburg, 193232, Russian Federation, zTiff0@gmail.com

**Kirichek Ruslan**

– Ph.D., associate professor, SPbSUT, St. Petersburg, 193232, Russian Federation, kirichek@sut.ru