Journal of Information Processing Vol.24 No.2 195–202 (Mar. 2016)

[DOI: 10.2197/ipsjjip.24.195]

Invited Paper

Survey of Real-time Processing Technologies of IoT Data Streams

Keiichi Yasumoto[1,a) Hirozumi Yamaguchi[2] Hiroshi Shigeno[3]

Received: September 2, 2015, Accepted: December 1, 2015

Abstract: Recently, Internet of Things (IoT) has been attracting attention due to its economical impact and high expectations for drastically changing our modern societies. Worldwide by 2022, over 50 billion IoT devices including sensors and actuators are predicted to be installed in machines, humans, vehicles, buildings, and environments. Demand is also huge for the real-time utilization of IoT data streams instead of the current off-line analysis/utilization of stored big data. The real-time utilization of massive IoT data streams suggests a paradigm shift to new horizontal and distributed architecture because existing cloud-based centralized architecture will cause large delays for providing service and waste many resources on the cloud and on networks. Content curation, which is the intelligent compilation of valuable content from IoT data streams, is another key to fully utilize and penetrate IoT technologies. In this paper, we survey the emerging technologies toward the real-time utilization of IoT data streams in terms of networking, processing, and content curation and clarify the open issues. Then we propose a new framework for IoT data streams called the Information Flow of Things (IFoT) that processes, analyzes, and curates massive IoT streams in real-time based on distributed processing among IoT devices.

Keywords: IoT, data stream, real-time processing, distributed processing, on-line learning, content curation

1. Introduction

Recently, Internet of Things (IoT) technology, which connects various physical things to the Internet, has been attracting considerable attention. In a white paper [1], Cisco predicted that by 2022, 50 billion things will be connected to the Internet that will produce 14.4 trillion dollars in revenue. IDC predicted that 28 billion IoT devices will be installed by 2020, and the annual market revenue will reach 700 billion dollars [2]. IoT was ranked as the top expected technology in Gartner's hype cycle 2014 [3] and continues to top the 2015 hype cycle. METI in Japan asserts that IoT will fuel a data-driven society where the digital data collected by IoT will acquire added value and benefit society [4]. IoT technology not only will hugely impact markets but also has large potential to drastically change society.

Many IoT research projects are working on their own respective purposes. For example, IoT-A [5] aims to establish IoT architecture, ClouT [6] is integrating cloud computing and IoT technologies, iCore [7] is establishing cognitive management frameworks, IoT6.eu [8] is applying IPv6 to IoT, and IERC [9] is integrating the results from different projects. Among the many research challenges in IoT, (1) *Heterogeneity*, (2) *Scalability*, (3) *Interoperability*, and (4) *Security and Privacy* have been identified as the most important challenges.

To tackle these challenges, various IoT platforms have been designed and implemented to interconnect IoT devices and process/merge data streams. Arkessa [10], Axeda [11], ThingSquare [12], Thingworx [13], WoTkit [14] and Xively [15] are examples of such mashup services. Most of these IoT platforms employ architectures based on a type of cloud computing called Platform as a Service (PaaS). Generated IoT data streams are placed in cloud storage as big data, and off-line analysis is applied with generous computation power and time to extract "intelligence" or patterns beneficial for business.

Since IoT data streams reflect current, real world situations, their *real-time utilization* is anticipated. For example, if live street view video of every place in a city can be made from videos captured in real-time by multiple mobile cameras carried by people and vehicles, tourism, economics, and security will benefit. For such a service, however, existing cloud-based approaches must introduce non-negligible delays until the service is provided, and this may reduce service quality and/or rapidly increase service costs due to wasting cloud computing resources and communication bandwidth. For the actual penetration of IoT, *content curation*, that is, creating valuable content from IoT data streams, is important. Curating content from various IoT data streams requires a *recipe* that consists of the following three steps:

- which data streams to use,
- how to process and analyze them and
- how to integrate multiple analysis results to form content.

Since the number of possible recipes will generally be enormous, *intelligence* is needed that automatically finds good recipes. Thus, we add the following two research challenges faced by IoT: (5) *timeliness* (real-time utilization of IoT data streams), and (6) *intelligence* (content curation from IoT data streams).

This paper is organized as follows. Section 2 provides IoT use

[1] Nara Institute of Science and Technology, Ikoma, Nara 630–0192, Japan
[2] Osaka University, Suita, Osaka 565–0871, Japan
[3] Keio University, Yokohama, Kanagawa 223–8522, Japan
[a)] yasumoto@is.naist.jp

cases that require timeliness and intelligence. Section 3 surveys existing technologies for IoT, and Section 4 clarifies unsolved problems. In Section 5, we propose a new IoT platform called IFoT that deals with timeliness and intelligence. Finally, we conclude the paper in Section 6.

2. Use Cases of Real-Time Utilization of IoT Data Streams and Key Challenges

In this section, we describe several use case scenarios for the real-time utilization of IoT data streams and pose key challenges.

2.1 Use Case Scenarios

- (S1) Participatory live street view

The first use case scenario is *participatory live street view,* shown in **Fig. 1**. This service allows a user to watch live street view video of any place in the city from any angle. To realize this service, based on user requests, relevant videos must be captured by the mobile cameras of people/vehicles and/or fixed cameras installed in the street, gathered, and processed to create/provide requested videos in real-time. If such a service is realized, it will benefit various purposes, such as tourist navigation (sightseeing, shopping, dining), economic activities (taxi/bus allocation, customer attraction), and security (surveillance, finding lost children).

However, implementing such a service on top of current cloud-based systems will be difficult, because such a huge number of uploads and downloads of video streams as well as their processing will exhaust both network bandwidth and cloud computation resources. Thus, the system does not scale or need huge cost for enhancing cloud systems. Unlike cloud-based architecture where all the streams converge to a single point, we need a new architecture that allows multiple data streams to directly flow between producers and consumers in parallel.

- (S2) Ultra-realistic live sports broadcasts based on UGC

Recently, User Generated Contents (UGC), which are uploaded to video/photo sharing websites, SNSs, and BBSs, are becoming more popular. Live video streaming applications such as Meerkat [16] and Periscope [17] allow users to share live video streams captured by smartphones.

Our second use case scenario is ultra-realistic live sports broadcasting services that target such popular sports as soccer, baseball, Olympic games, and World cup matches using UGC. This service collects information in and around stadiums including cheers, weather conditions, atmosphere and tweets (SNS messages) as well as video streams from the mobile/wearable devices of spectators and audience members and creates video content for broadcast by integrating the collected information. Such a service can create new experiences for sport fans who cannot attend live sporting events.

A key challenge to realize this service is how to create content that has value to prospective audiences by intelligently selecting, processing, and integrating data streams. This intelligent content creation task is called *content curation*. For content curation, we need techniques for capturing and handling various data streams in a unified way as well as efficiently identifying relevant data streams.

- (S3) City-wide real-time pedestrian flow tracking

Our third use case scenario is real-time pedestrian flow tracking in crowded city areas. It is desirable to grasp pedestrian flow between points and/or areas for various purposes, for example, providing transportation options for smooth transitions and smooth evacuation guidance during emergencies.

When a time series of the position data of tens of thousands to millions of pedestrians is uploaded to a cloud and processed/analyzed there, it will be difficult to track such pedestrian-flow changes in real-time; many cellular bandwidth and cloud resources will also be wasted.

We believe that the required spatio-temporal granularity of people flow depends on where the information is used. That is, the granularity of pedestrian-flow information near the requesting place must be fine, but the information far from the place may be coarser. The insight from this use case is that data streams must be processed and aggregated near their generation sources to reduce them and propagate them in far places in a scalable manner.

- (S4) Real-time anomaly detection for seniors living alone

In recent years, the solitary deaths of elderly people who are living alone is becoming a big social problem [18]. We must realize elderly monitoring services that detect anomalies in real-time and timely notify caretakers and families.

Many elderly monitoring services have been provided that use electric pots, electricity/gas remote meters and sensors attached to toilets and refrigerators. These systems, however, are time-consuming for detecting anomalies and may not be effective for lifesaving purposes. Camera-based monitoring systems can quickly detect anomalies like falls, but they violate privacy. Moreover, most existing monitoring services are constructed as cloud services. The cost of continuously using cloud communication and processing is non-negligible, and leakage risks are induced by storing privacy data on the cloud.

Ueda et al. [19] proposed an in-home living activity recognition method, where 11 different activities are recognized with more than 90% accuracy using indoor position sensors and power meters. However, the method is based on off-line learning; no real-time activity recognition is provided. We must realize a low-cost system as well as one that protects privacy and processes private data streams near sources and sends only aggregated information

Fig. 1 Live street view.

to the cloud.

2.2 Key Challenges

The following four key challenges arise from the use case scenarios.

C1: Creation technology for IoT data streams: capturing various real world events anywhere and anytime in a unified manner.
C2: Networking technology for IoT data streams: enabling direct flow between producers and consumers in parallel.
C3: Processing technology for IoT data streams: processing and aggregating data streams near their sources.
C4: Content curation technology: intelligently selecting necessary streams and processing and integrating them into valuable content, based on the interests of prospective users.

3. Enabling Technologies

In this section, we survey the technologies that enable the challenges discussed in Section 2.

3.1 Sensor as a Data Stream Generator

A variety of sensors can sense the mechanical, thermal, biological, chemical, optical, and magnetic properties of physical environments. Due to the progress of Micro-Electro-Mechanical Systems (MEMS) technologies, sensor nodes, on which sensors are equipped for communication and processing capabilities, are becoming smaller. These sensor nodes are organized into Wireless Sensor Networks (WSNs) [20], which are deployed on farms and in factories to monitor them. Sensors are embedded not only in wearable devices such as eyeglasses and watches but also in cups [21], furniture and sporting equipment like tennis rackets [22] and basketballs [23], door locks [24], and even fish finders [25]. They are connected by the Internet to cloud servers, and smartphones are common ways to access, control, and visualize sensing data.

Participatory sensing and opportunistic sensing [26] also exploit sensor nodes, which mainly rely on mobile agents like humans and vehicles. Well-known or recent projects include EarPhone [27], GreenGPS [28], and SakuraSensor [29]. In particular, incentive mechanisms and gamification to promote user participation are recent challenging topics [30].

As seen above, a variety of sensors may generate a number of IoT data streams. Nevertheless, they are basically utilized by dedicated software or platforms due to the proprietary aspects of devices and a lack of standard platforms that enable many developers to easily obtain, analyze, and combine data streams in the context of applications and services that are provided to users. In other words, content-centric (or dependent) stream processing is required to migrate service-level processing tasks from cloud servers to distributed components in networks to mitigate the data processing cost, which is high in terms of delay, server load, and throughput.

3.2 Networking Technologies for IoT data Streaming

A number of protocols have been proposed for academic research, industrial use, and the standardization of IoT/M2M communication. For lightweight communications where IoT sensors are restricted due to limited energy sources or processing capabilities, several protocols have been proposed, such as Message Queue Telemetry Transport (MQTT) [31] and Constrained Application Protocol (CoAP) [32]. MQTT is a lightweight publish/subscribe messaging transport protocol based on a client-server model designed for M2M and IoT applications with constrained networks. CoAP is a simplified web transfer protocol that is specialized for use with constrained nodes and networks such as in M2M applications. It provides asynchronous request/response interactions between clients and servers over UDP, which can easily interface with HTTP.

Other protocols have also been proposed, including Websocket [33] and IPv6 over Low power Wireless Personal Area Networks (6LowPAN) [34]. WebSocket is a full-duplex protocol over a TCP connection that typically provides bidirectional communication between web browsers and web servers. 6LoWPAN defines an IPv6 header compression format for IPv6 packet delivery in low-power wireless personal area networks, i.e., IEEE 802.15.4. These protocols are designed for peer-to-peer communication, client-server communication on the Internet, or communication in WPANs. The peer-to-peer style of communication cannot provide real-time services. Instead, it is more feasible and reasonable to connect heterogeneous IoT devices, often directly through heterogeneous access networks or local cloud servers.

This concept leads to *edge-heavy computing*. EdgeComputing [35] and Fog Computing [36] are based on such paradigms where data processing is executed on those components in or on the edge of networks to mitigate server load. The demerit of these approaches is the need for investment to replace such network constituents like Information-Centric networks (ICNs). Edge-Centric Computing [37], which seeks a more practical solution by extending EdgeComputing and Fog Computing, delegates the processing tasks of cloud servers to other distributed systems like P2P to realize service components such as proximity, intelligence, trust and control outside the cloud.

As recent work on IoT, MINA [38] is an integrated network system that provides seamless unification of different wireless access technologies like cellular, WiFi, ZigBee, and Bluetooth and multi-hop communication technologies like MANET. It uses software defined networks (SDNs) and flexibly delivers data streams among devices. To cope with the issue of the heterogeneity of available resources in such multinetwork environments, MINA, which is composed of SDN controllers on four different layers, monitors the available resources and schedules the data streaming considering the QoS required by the services.

3.3 IoT Data Stream Processing

Unlike conventional DBMS that assume all data are stored in DBs before analysis, more than a few IoT systems assume that data from IoT devices are streams, where data elements have temporal relationships and require real-time processing. Most are also redundant with respect to the data's value.

Following its definition by Ref. [39], data stream processing is represented, at a high level, as a graph of FIFO queues that correspond to data streams and operators that may take multiple inputs/outputs from/to those FIFO queues. Operators are continuous stream transformers, which must contain activation, initialization, and output data rate policies. For example, the function of averaging the input data streams is a simple stream transformer.

In the context of databases, such a system that deals with data streams is often called a *Data Stream Management System* (*DSMS*). In contrast to DBMS, which issues a single query to an entire dataset, DSMS updates the result whenever new data arrive, but it can be seen as an extension of DBMS since it is a query-based system.

Complex Event Processing (CEP) [39], [40] is another well-known technique to detect events that satisfy given conditions over different streams. Historically, such single event processing as attribute-based data and interest filtering has been employed in publisher/subscriber systems, but these systems with multiple sensors must deal with more complex conditions. CEP has been exploited in many fields like distributed information systems, business process automation, control systems, network monitoring, and sensor networks. CEP supports a variety of languages (Java, Python, and R) to specify conditions, but SQL-based ones are popular in many systems. For example, Continuous Query Language (CQL) [41] was initially developed by Stanford for their STREAM [42] system, and Oracle, uCosminexus Stream Data Platform (Hitachi Co. LTD.), and others have implemented CQL in their systems. Event Processing Language (EPL) is another common language.

Several platforms exist for stream processing, including IBM Infosphere Stream [43] (which uses IBM Streams Processing Language (SPL)), SAP Sybase Event Stream Processor, StreamBase [44], SQLstream Blaze, Amazon AWS IoT, Yahoo!S4, TIBCO Business Events, Microsoft StreamInsight, Apache Storm [45], and Apache Spark [46]. These platforms are tuned for high-speed real-time processing of a massive amount of temporal data. For example, Spark has a function called Resilient Distributed Dataset (RDD) that hides parallel and distributed operations over multiple streams and provides seamless access to service users. More academic issues have been discussed in scientific research. For example, CLARO [47] designed stochastic query processing when the input data are uncertain and have errors.

Machine learning and statistics are considered part of IoT data streaming frameworks. Feature selection and principal component analysis (PCA) help reduce the data dimensions to maintain the capability of representing data characteristics. Machine learning also involves operations to create such classifiers as Support Vector Machine (SVM) or unsupervised learning (like clustering) and to train various models, including linear function, k-nearest neighbor, and logistic model by regression analysis. These operations can be applied to data streams slotted by a certain time window or revised for incremental updates of models whenever new data arrive. This is often called *on-line machine learning*, whose popularity is rising due to increased attention to IoT and big data. Supervised on-line machine learning represents the errors between the true and estimated values by models and updates the model parameters to minimize the errors whenever data arrive. Assuming that the error functions are represented as probability distribution functions, the parameters are updated to minimize the expected values of errors. Stochastic Gradient Descent (SGD), a well-known method for this purpose, was originally designed for randomly picked data from a dataset; but by assuming the incremental feeding of data to the procedure, it can be applied to data streams with limited memory space. Similarly, many algorithms for batch (or off-line) machine learning algorithms can be converted to on-line versions if we control references to training data and model updates.

Many of the IoT platforms introduced above support on-line learning schemes. For example, AWS and StreamInsight provide Amazon Machine Learning [48] and Azure Machine Learning [49] to enable data processing and analysis over data streams. However, both systems need cloud servers, which are often expensive for many types of applications and services in terms of performance. Jubatus [50] supports distributed on-line machine learning, but it does not focus on distributed stream fusion based on service-level context.

3.4 Service Composition from IoT Data Streams

Some programming models and tools have been developed and provided for creating contents (composing services) by collecting and merging various IoT data streams. Web of Things [51] is a programming model where services can be easily mashed-up by associating objects with web components using web 2.0 technology. WotKit [52] is a visual programming tool for service mashups. IBM also provides a visual programming tool called Node-RED [53] where a new service can be created just by drawing lines among IoT devices, APIs, and services. Another study [54] extends Node-RED to treat distributed data streams. Mobile Fog [55] is a programming model that constructs large-scale IoT services.

Although these programming models and tools facilitate users to easily and intuitively compose services and/or create contents, they still need to manually design output layouts of the contents and specify the processing sequence of the streams until the content is derived. Therefore we need automated content creation adaptive to the availability of data streams and their dynamism.

Fujisawa et al. proposed a video curation system [56] that targets baseball games and automatically creates real-time video content with high values from multiple video streams captured by spectator cameras in different places and at different angles and zoom levels. In their study, assuming that video contents with similar camera switching patterns (i.e., which camera's video is used in the broadcasted content and when) to the TV broadcast have high values, machine learning algorithms are constructed using the camera switching patterns of TV broadcasts as training data.

3.5 International Activities on IoT Framework Design and Standardization

Many IoT-related organizations, consortiums, and projects exist. For example, Open Interconnect Consortium (OIC), which was founded by Intel, Samsung, and others, released IoTivity [57], an open source software framework that enables seamless device-to-device connectivity to address the emerging needs of the Internet of Things (IoTivity 1.0 was released in

October 2015). AllSeen Alliance from the Linux Foundation released Alljoyn [58], an open-source IoT framework by Qualcomm. OneM2M [59], which developed technical specifications for M2M services, releases standards to create a foundation platform for IoT devices and applications. Industrial Internet Consortium (IIC) [60] focuses more on industrial applications for IoT; Intel, GE and some others are leading this consortium. Though their goals are different, they basically share a common vision for IoT where everything is connected by the Internet to support next generation applications and services that have deeply penetrated our social lives, societies, industries, and infrastructures. The IPSO Alliance [61] established the Internet Protocol as the basis for the connection of Smart Objects. The European Commission 7th Framework program (EU-FP7) sponsors the IoT European Research Cluster (IERC) [9] that addresses the large potential for IoT-based capabilities in Europe involving international partners from Europe, USA, Japan, China, and Korea.

Many platforms and related projects have also been developed. Some platforms for developing IoT applications are now available on the market and primarily focus on processing large-scale real-time streams. Research-based projects share this goal. For example, the concept of the EU-Japan funded project ClouT [6] is leveraging cloud computing as an enabler to bridge things, people, and services by the Internet. Another EU-Japan project, FESTIVAL [62], connects and unites European and Japanese IoT testbeds to provide IoT experimentation platforms for homogeneous access APIs with an Experimentation as a Service (EaaS) model for experimenters. Many other EU-funded projects exist: CASCADAS2, VITAL-IoT [63], and IOT-I & IOT-A [5].

Finally, smart city projects are closely related with IoT technologies. In the EU, many cities are now interested in making them smarter with respect to such infrastructure-related issues as energy, mobility, government services, and health. The following are well-known smart city projects: Santander supported by Future Internet Research (FIRE), BCN Smart City (Barcelona), Valencia Smart City, and Smart Beehive Project (Ireland). For smaller-scale networks (in comparison with smart cities) that are deployed in homes, buildings, and offices, several platforms are available. For example, HomeKit and Brillo are provided by Apple and Google, IoTivity by OIC, and AllJoyn by AllSeen.

4. Open Issues

This section reflects on the use case scenarios in Section 2 and key challenges C1-C4 and clarifies the open issues for the real-time utilization of IoT data streams.

C1 (Creation technology for IoT data streams): Scenarios S1-S4 utilize different sensors/data streams, such as those from cameras, microphones, accelerometers, ambient sensors, position sensors, power meters, vital sensors, and SNSs. Thus, C1 requires a common representation format of various IoT data streams because they are processed and combined to form new streams. The common format must also be able to represent higher-level streams. Hereafter, we use the term *flows* to refer to both raw data streams and higher-level streams after processing. The common format should include metadata that help devices/servers easily search for necessary flows, aggregate/summarize flows, and obtain knowledge or patterns. Existing IoT technologies do not have a common representation format for handling different types/levels of flows in a unified manner.

C2 (Networking technology for IoT data streams) and C3 (Processing technology for IoT data streams): Since scenarios S1-S4 suppose real-time stream distribution among devices and real-time processing and analysis of flows, they require an adaptive processing mechanism to meet real-time constraints by adaptively allocating computation resources and/or a granularity adjustment mechanism to satisfy the bandwidth constraint. On-line learning is another issue. Flows with high dynamism must be analyzed in real-time, and knowledge or patterns such as contexts and objects must be detected on-line so that the learned tags are attached to flows for real-time utilization. Few existing IoT platforms consider both network/computation resources and data granularity adjustments to achieve real-time distribution of flows.

C4 (Content curation technology): Scenarios S1-S4 suppose real-time content curation from multiple flows. There are two kinds of curators: human and machine. For human curators, support for the selection of relevant flows and for understanding them is essential (C5). There are three functions for human curators: intelligent flow search that considers the curator's value, flow visualization for understanding the content, and flow prediction for understanding temporal changes. Realizing machine curators is a very interesting but challenging issue. Predicting content values for prospective audiences is one part of the key challenges. Few existing IoT platforms have developed automated curators or functions to support human curators to manage real-time contents. Machine learning-based automatic video curation from spectator mobile cameras that target baseball games was proposed [55], but TV broadcasts are used as training data

Table 1 Challenges and technical issues for real-time utilization of IoT flows.

Challenges	Purposes	Issues
C1 Creation	generate flows in common format	metadata format, resource allocation, granularity adjustment
C2 Networking	distribute flows in real-time and scalable manner	real-time constraints, scalability
C3 Processing and analysis	attach tags, aggregate/summarize/analyze flows	adaptive resource allocation, analysis, context/object recognition, on-line learning
C4 Content curation and presentation	create content by integrating multiple flows, present content effectively	visualization, user interfaces, automated curation
C5 Understanding	support human curators to understand flows	intelligent search, visualization, predictions of changes
C6 Security and privacy	make people feel secure when they provide and/or use flows	privacy preservation, prevention of falsification

by assuming that they have high values. Since preparing training data that only contain good curations is difficult, a method is required for estimating the user values of a given curation.

Another big challenge is related to security and privacy issues (C6). Scenarios S1-S4 utilize privacy-sensitive data flows such as location data, vital signs, and video/audio data. For the wide penetration and the utilization of IoT flows, functions are necessary that make people feel secure and safe when distributing their flows and/or using the flows of others. Many security studies treat individual data types like location data. Security architecture for the cyber-physical-social world was proposed [64], but few treat both security and privacy issues in the IoT context such as data heterogeneity and real-time distribution.

Table 1 summarizes the main challenges, their purposes, and the remaining technical issues.

5. IFoT: Real-Time Information Flow Processing Framework

As discussed above, most existing IoT platforms do not fully support both distributed and on-site processing. Even for local services, we need to set up a cloud server and collect/process data streams in servers far from the data sources. Such architecture not only limits communication and computation capacity but also requires additional efforts for handling privacy-sensitive data, creating barriers to the real-time utilization of IoT big data.

In this section, we propose the *Information Flow of Things* (*IFoT*), a new framework for processing, analyzing, and curating IoT data streams in real-time and in a scalable manner based on distributed processing among IoT devices. In IFoT, both raw data streams and higher-level streams after processing/aggregating/merging are called *information flows* (or *flows*) and treated identically.

IFoT aims to solve the following three technical issues: (1) handling various information flows in a unified manner, (2) processing and analyzing flows in their proximity and distributing them directly between devices in real-time and in a scalable manner, and (3) intelligently integrating different flows into content (as a higher-level flow) and providing it in real-time. These issues are solved by three different layered components: *IFoT-Neuron*, *IFoT-PO3-Engine*, and *IFoT-Curator* (**Fig. 2**).

5.1 IFoT-Neuron

IFoT-Newron, which is an abstraction of an intelligent sensing device and a flow source, captures and processes data in the real world and sends them out as a flow(s). For this purpose, it incorporates functions for creating relevant information flows (C1 in Table 1), and processing/analyzing flows (C3), and handling security and privacy issues (C6). IFoT-Neuron is expected to be installed in every IFoT compatible device (called an *IFoT node*) as a software library or a hardware module.

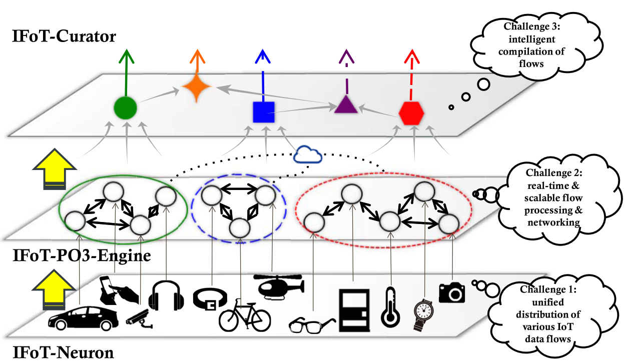

Fig. 2 IFoT: challenges and approaches.

IFoT-Neuron has communication capabilities with nearby IFoT nodes, connections to the Internet (optional), and processing flows such as attaching tags, basic stream processing, and anonymization.

To tackle challenge C1, for different flows, we define a common metadata format that consists of data type, granularity, location information, and a set of tags for each time interval of the flows.

For challenge C2, tags (i.e., contexts, identified objects/events, etc.) are derived through a learning algorithm implemented in the IFoT-PO3-Engine and automatically attached to flows. The metadata associated with each flow facilitate efficient searches and further processing of the flows by other IFoT nodes. Moreover, for the real-time distribution of flows, it offers a *dynamic granularity adjustment function* that reduces the data granularity, as requested by the IFoT-PO3-Engine.

For the wide penetration of IFoT compatible devices, IFoT-Neuron should be implemented as a small, low-cost, and low-power hardware component with sufficient computation power. Toward zero-energy operation, MEMS and energy-harvesting technologies should also be employed.

5.2 IFoT-PO3-Engine

To distribute flows between sources and users without stagnation, collecting low-level (raw) flows in clouds is not a good idea because of the bandwidth waste in paths to the cloud and the imposition of large delays. Instead, it is desirable to process, analyze, and aggregate flows near their sources to reduce the required bandwidth between sources and destinations. We call this concept "*Process On Our Own*," or *PO3* in short.

For a concrete shape of this concept, we designed an IFoT-PO3-Engine that offers functions for executing high-load tasks including complex event processing and on-line learning among IFoT nodes in a distributed and cooperative manner and efficiently distribute resulting higher-level flows to remote IFoT nodes.

For distributed and cooperative processing, predicting processing time for a heavy task is required. If the predicted time does not satisfy the time constraint, the task is divided into sub-tasks, which are sent to nearby IFoT nodes for execution. Division into sub-tasks and allocating them are dynamically done by considering the available bandwidth in the network and the computation power of the nearby IFoT nodes.

Real-time flow distribution among remote IFoT nodes is another issue to be solved. The IFoT-PO3-Engine searches for multiple routes to a destination node (including routes through cellular networks), measures or estimates delays and available bandwidth on each route, and establishes a multi-path route to deliver a flow in real-time in cooperation with the dynamic granularity adjustment function of IFoT-Neuron.

5.3 IFoT-Curator

IFoT aims to realize the real-time utilization of information flows by providing users with content curated from multiple flows in real-time. To this end, we need to define a language to describe a *curation recipe* (e.g., a task graph to create content with the data's required spatio-temporal granularity) and its execution system. When a curation recipe is submitted to an IFoT node, it is executed among nearby IFoT nodes in a distributed manner with an API provided by the IFoT-PO3-Engine and the IFoT-Neuron. Therefore, the execution system is designed and implemented as middleware with such functions as code/data migration and distributed/cooperative task processing.

IFoT-Curator aims not only for the execution of human-edited recipes but also for the support for recipe-editing work (C5 in Table 1) and the further automatic creation of recipes (C4 in Table 1). For the challenge C5, it is required for a human curator to be able to acquire only a special subset of massive flows that match his/her interests, visualize dynamism in flows and predict their future change for their better understanding.

For automated curation, it is also needed to realize a function to measure the value of a content's value for its prospective audience and a function to predict new content's expected value created with each possible recipe. Even though the latter is challenging, this function must be achieved.

6. Conclusion

IoT technologies offer the potential to drastically change our societies. The keys are the real-time utilization of IoT data streams and intelligent content creation (content curation) from these data streams. However, existing network and cloud computing architectures may not be able to accommodate the massive data streams generated by as many as a trillion IoT devices in real-time. Thus, a paradigm shift is essential for new information processing architecture that allows data streams to flow in the required form among places.

In this paper, we surveyed the existing and emerging technologies toward real-time IoT data stream utilization and content curation, clarified open problems, and proposed a new framework called Information Flow of Things (IFoT) for processing IoT data streams in real-time. To realize IFoT, many challenging issues need to be solved. We hope this paper spurs prospective researchers in related fields to advance their research toward the realization of data-driven societies.

Acknowledgments This study was supported in part by JSPS Grant in Aid for Scientific Research 26220001, 25280031 and 15H02690.

References

[1] Bradley, J., Barbier, J. and Handler, D.: Embracing the Internet of Everything To Capture Your Share of $14.4 Trillion (White Paper), available from ⟨http://www.cisco.com/web/about/ac79/docs/innov/IoE_Economy.pdf⟩ (accessed 2015-11-08).

[2] IDC Market in a Minute: Internet of Things, available from ⟨http://www.idc.com/downloads/idc_market_in_a_minute_iot_infographic.pdf⟩ (accessed 2015-11-08).

[3] Gartner's 2014 Hype Cycle for Emerging Technologies Maps the Journey to Digital Business, available from ⟨http://www.gartner.com/newsroom/id/2819918⟩ (accessed 2015-11-08).

[4] Ministry of Education, Trade and Industry: Changes in response to the arrival of a data-driven society using CPS, available from ⟨http://www.meti.go.jp/committee/sankoushin/shojo/johokeizai/pdf/report01_04_00.pdf⟩

[5] IoT-A, Internet of Things – Architecture, available from ⟨http://www.iot-a.eu/public⟩ (accessed 2015-11-08).

[6] Cloud of Things for empowering the citizen clout in smart cities, available from ⟨http://clout-project.eu/⟩ (accessed 2015-11-08).

[7] iCore Project: available from ⟨http://www.iot-icore.eu/⟩ (accessed 2015-11-10).

[8] Researching IPV6 potential for the Internet of Things, available from ⟨http://iot6.eu/⟩ (accessed 2015-11-08).

[9] IERC, European Research Cluster on the Internet of Things, available from ⟨http://www.internet-of-things-research.eu/⟩ (accessed 2015-11-08).

[10] Arkessa, available from ⟨http://www.arkessa.com/⟩

[11] Axeda, available from ⟨http://www.axeda.com/⟩

[12] ThingSquare, available from ⟨http://www.thingsquare.com/⟩

[13] Thingworx, available from ⟨http://www.thingworx.com/⟩

[14] Blackstock, M. and Lea, R.: IoT mashups with the WoTKit, *Proc. IEEE Internet of Things* (*IOT*), pp.159–166 (2012).

[15] Xively, available from ⟨http://xively.com⟩

[16] Meerkat, https://meerkatapp.co/ (Accessed 2015-11-11).

[17] Periscope, available from ⟨https://www.periscope.tv/⟩ (accessed 2015-11-11).

[18] Cabinet Office: Annual Report on the Aging Society: 2014 (Summary), available from ⟨http://www8.cao.go.jp/kourei/english/annualreport/2014/2014pdf_e.html⟩ (accessed 2015-11-30).

[19] Ueda, K., Suwa, H., Arakawa, Y. and Yasumoto, K.: Exploring Accuracy-Cost Tradeoff in In-Home Living Activity Recognition based on Power Consumptions and User Positions, *Proc. 14th IEEE Int'l. Conf. on Ubiquitous Computing and Communications* (*IUCC 2015*), pp.1131–1137 (2015).

[20] Yick, J., Mukherjee, B. and Ghosal, D.: Wireless sensor network survey, *Computer Networks*, Vol.52, pp.2292–2330 (2008).

[21] Intel shows off a light-up smart mug, because why not?, available from ⟨http://www.engadget.com/2014/01/07/intel-smart-mug-concept/⟩ (accessed 2015-11-09).

[22] Smart Tennis Sensor for Tennis Rackets, available from ⟨http://www.sony.com/electronics/smart-devices/sse-tn1w⟩ (accessed 2015-11-09).

[23] 94FiFty, available from ⟨http://www.94fifty.com/⟩ (accessed 2015-11-09).

[24] GOji, available from ⟨http://gojiaccess.com/⟩ (accessed 2015-11-09).

[25] Deeper, Smart Fishfinder, available from ⟨https://buydeeper.com⟩ (accessed 2015-11-09).

[26] Higuchi, T., Yamaguchi, H. and Higashino, T.: [Invited Paper] Mobile Devices as an Infrastructure: A Survey of Opportunistic Sensing Technology, *Journal of Information Processing*, Vol.23, No.2, pp.94–104 (2014).

[27] Rana, R.K., Chou, C.T., Kanhere, S.S., Bulusu, N. and Hu, W.: Earphone: An end-to-end participatory urban noise mapping system, *Proc. 9th ACM/IEEE Intl. Conf. on Information Processing in Sensor Networks* (*IPSN 2010*), pp.105–116 (2010).

[28] Ganti, R.K., Pham, N., Ahmadi, H., Nangia, S. and Abdelzaher, T.F.: GreenGPS: A participatory sensing fuel-efficient maps application, *Proc. 8th Intl. Conf. on Mobile Systems, Applications, and Services* (*MobiSys 2010*), pp.151–164 (2010).

[29] Morishita, S., Maenaka, S., Nagata, D., Tamai, M., Yasumoto, K., Fukukura, T. and Sato, K.: SakuraSensor: Quasi-Realtime Cherry-Lined Roads Detection through Participatory Video Sensing by Cars, *Proc. 2015 ACM Intl. Joint Conf. on Pervasive and Ubiquitous Computing* (*UbiComp 2015*), pp.695–705 (2015).

[30] Arakawa, Y. and Matsuda, Y.: [Invited Paper] Gamification mechanism for enhancing a participatory urban sensing: survey and practical results, *Journal of Information Processing*, Vol.24, No.1, pp.31–38 (2016).

[31] OASIS Standard, MQTT version 3.1.1, available from ⟨http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.doc⟩ (2014).

[32] Shelby, Z., Hartke, K. and Bormann, C.: Request for Comment 7252, The Constrained Application Protocol (CoAP), available from ⟨http://tools.ietf.org/rfc/rfc7252.txt⟩ (2014).

[33] WebSocket, available from ⟨https://www.websocket.org/⟩

[34] Shelby, Z. and Borman, C.: 6LoWPAN: The Wireless Embedded Internet, John Wiley & Sons (2011).

[35] Davis, A., Parikh, J. and Weihl, W.E.: Edgecomputing: Extending enterprise applications to the edge of the internet, *Proc. 13th International World Wide Web Conference on Alternate Track Papers and Posters*, pp.180–187 (2004).

[36] Bonomi, F., Milito, R., Zhu, J. and Addepalli, S.: Fog Computing and Its Role in the Internet of Things, *Proc. 1st Edition of the MCC Workshop on Mobile Cloud Computing* (*MCC'12*), pp.13–16 (2012).
[37] Lopez, P.G., Montresor, A., Epema, D., Datta, A., Higashino, T., Iamnitchi, A., Barcellos, M., Felber, P. and Riviere, E.: Edge-centric Computing: Vision and Challenges, *ACM SIGCOMM Computer Communication Review*, Vol.45, No.5, pp.37–42 (2015).
[38] Qin, Z., Denker, G., Giannelli, C., Bellavista, P. and Venkatasubramanian, N.: A Software Defined Networking Architecture for the Internet-of-Things, *Proc. IEEE Network Operations and Management Symposium* (*NOMS*), pp.1–9 (2014).
[39] Hirzel, M., Soulé, R., Schneider, S., Gedik, B. and Grimm, R.: A catalog of stream processing optimizations, *ACM Comput. Surv.*, Vol.46, No.4, Article 46, pp.1–34 (2014).
[40] Cugola, G. and Margara, A.: Processing flows of information: From data stream to complex event processing, *ACM Comput. Surv.*, Vol.44, No.3, Article 15, pp.1–62 (2012).
[41] Arasu, A., Babu, S. and Widom, J.: The CQL continuous query language: Semantic foundations and query execution, *The VLDB Journal*, Vol.15, No.2, pp.121–142 (2006).
[42] Arasu, A., Babcock, B., Babu, S., Datar, M., Ito, K., Motwani, R., Nishizawa, I., Srivastava, U., Thomas, D., Varma, R. and Widom, J.: STREAM: The stanford stream data manager, *IEEE Data Eng. Bull.*, Vol.26, No.1, p.665 (2003).
[43] Gedik, B. and Andrade, H.: A model-based framework for building extensible, high performance stream processing middleware and programming language for IBM InfoSphere streams, *Softw. Pract. Exp.*, Vol.42, No.11, pp.1363–1391 (2012).
[44] StreamBase Systems, available from ⟨http://www.streambase.com⟩ (2012).
[45] Storm project, available from ⟨http://storm-project.net/⟩ (2012). Retrieved May 2012.
[46] Apache Spark-Lighting-fast cluster computing, available from ⟨http://spark.apache.org/⟩
[47] Tran, T.T.L., Peng, L., Diao, Y., McGregor, A. and Liu, A.: CLARO: Modeling and processing uncertain data streams, *The VLDB Journal*, Vol.21, pp.651–676 (2012).
[48] Amazon: Amazon Machine Learning, available from ⟨https://aws.amazon.com/machine-learning/⟩ (accessed 2015-11-13).
[49] Microsoft: Azure Machine Learning, available from ⟨https://azure.microsoft.com/en-us/services/machine-learning/⟩ (accessed 2015-11-13).
[50] Jubatus: Distributed Online Machine Learning Framework, available from ⟨http://jubat.us/en/⟩ (accessed 2015-11-15).
[51] Guinard, D., Trifa, V. and Wilde, E.: A resource oriented architecture for the web of things, *Proc. Internet of Things* (*IOT*), pp.1–8 (2010).
[52] Blackstock, M. and Lea, R.: IoT mashups with the WoTKit, *Proc. Internet of Things* (*IOT*), pp.159–166 (2012).
[53] IBM: Node-RED, available from ⟨http://nodered.org/⟩ (accessed 2015-11-10).
[54] Blackstock, M. and Lea, R.: Toward a Distributed Data Flow Platform for the Web of Things (Distributed Node-RED), *Proc. 5th Intl. Workshop on Web of Things*, pp.34–39 (2014).
[55] Hong, K., Lillethun, D., Ramachandran, U., Ottenwälder, B. and Koldehofe, B.: Mobile fog: A programming model for large-scale applications on the internet of things, *Proc. 2nd ACM SIGCOMM Workshop on Mobile Cloud Computing*, pp.15–20 (2013).
[56] Fujisawa, K., Hirabe, Y., Suwa, H., Arakawa, Y. and Yasumoto, K.: Automatic Content Curation System for Multiple Live Sport Video Streams, The *11th IEEE Int'l. Workshop on Multimedia Information Processing and Retrieval* (*MIPR 2015*) (2015).
[57] IoTivity, available from ⟨https://www.iotivity.org/⟩ (accessed 2015-11-13).
[58] AllJoyn Framework, available from ⟨https://allseenalliance.org/framework⟩ (accessed 2015-11-13).
[59] oneM2M, available from ⟨http://www.onem2m.org/⟩ (accessed 2015-11-13).
[60] Industrial Internet Consortium, available from ⟨http://www.iiconsortium.org/⟩ (accessed 2015-11-13).
[61] IPSO Alliance, available from ⟨http://www.ipso-alliance.org/⟩ (accessed 2015-11-13).
[62] FESTIVAL: FEderated interoperable SmarT ICT services deVelopment And testing pLatforms, available from ⟨http://www.festival-project.eu/en/⟩ (accessed 2015-11-13).
[63] Vital, The future of Smart Cities, available from ⟨http://vital-iot.eu/⟩.
[64] Ning, H. and Liu, H.: Cyber-physical-social based security architecture for future internet of things, *Advances in Internet of Things*, Vol.2, No.1, p.1 (2012).

Keiichi Yasumoto received his B.E., M.E., and Ph.D. degrees in information and computer sciences from Osaka University, Osaka, Japan, in 1991, 1993 and 1996, respectively. He is currently a professor of the Graduate School of Information Science at Nara Institute of Science and Technology. His research interests include distributed systems, mobile computing, and ubiquitous computing. He is a member of ACM, IEEE, and IEICE.

Hirozumi Yamaguchi received his B.E., M.E., and Ph.D. degrees in information and computer sciences from Osaka University, Japan, in 1994, 1996, and 1998, respectively. He is currently an associate professor at Osaka University. His current research interests include design, development, modeling, and simulation of mobile and wireless networks and applications. He is a member of IEEE.

Hiroshi Shigeno received his B.S., M.E. and Ph.D. degrees in instrumentation engineering from Keio University, Japan in 1990, 1992 and 1997. Since 1998, he has been with the Department of Information and Computer Science at Keio University, where he is currently a professor. His current research interests include mobile and ubiquitous computing, network architecture, and Intelligent Transport Systems. He is a member of IEEE, ACM, and IEICE.