

A Study to Optimize Heterogeneous Resources for Open IoT

Yoji Yamato, Naoto Hoshikawa, Hirofumi Noguchi, Tatsuya Demizu and Misao Kataoka

NTT Network Service Systems Laboratories

NTT Corporation

Musashino-shi, Tokyo 180-8585, Japan

Email: yamato.yoji@lab.ntt.co.jp

Abstract—Recently, IoT technologies have been progressed, and many sensors and actuators are connected to networks. Previously, IoT services were developed by vertical integration style. But now Open IoT concept has attracted attentions which achieves various IoT services by integrating horizontal separated devices and services. For Open IoT era, we have proposed the Tacit Computing technology to discover the devices with necessary data for users on demand and use them dynamically. We also implemented elemental technologies of Tacit Computing. In this paper, we propose three layers optimizations to reduce operation cost and improve performance of Tacit computing service, in order to make as a continuous service of discovered devices by Tacit Computing. In optimization process, appropriate function allocation or offloading specific functions are calculated on device, network and cloud layer before full-scale operation.

Index Terms—IoT, Tacit Computing, COpen IoT, CMulti Layer-Optimization, User Context

I. INTRODUCTION

Recently, IoT (Internet of Things) technologies have been progressed (e.g., [1]–[7]), and many IoT devices are connected to networks. IoT application areas are wide such as manufacturing, supply chain, maintenance which Industrie4.0 targets and also health care, agriculture and energy.

However, current IoT applications tend to be one off solutions with vertical integration for specified targets. System integration businesses which carry out sensor selecting, analyzing, visualizing and taking actions together are popular for IoT applications. Therefore, each IoT application needs high development cost.

To reduce costs of various IoT application development and operation, shared use of devices such as sensors and actuators and integrating horizontal separated devices and services are needed. This approach is called Open IoT and attracts attentions for future IoT applications acceleration.

We have proposed Tacit Computing technology [9] to utilize shared use of devices from various services for Open IoT era. Tacit Computing can discover devices with necessary data for users on demand based on live data of each device, and can use them. We have also implemented elemental technologies of Tacit Computing.

However, we think discovering and using the device based on the situation at that time only answers the user's needs only one time. In order to make it a service that keeps answering to the needs of user, it is needed to provide the

continuous use of devices with necessary data for the user at a reasonable price. Therefore, in this paper, we propose three layers optimizations to reduce operation cost and improve performance so as to make a continuous service with devices which Tacit Computing discovers and uses. In optimization processing, appropriate function allocations and offloading specific processing are conducted on device, network and cloud layers before full-scale operation. This paper explains approaches of three layers optimizations and we report detail evaluation of each optimization technology in other papers.

II. OUTLINE OF TACIT COMPUTING

Tacit Computing answers users' requests by discovering and coordinating appropriate resources for users from cloud layer, network layer and device layer (Figure 1). Tacit Computing has three layers structure but its concept is to process as much as possible in the device layer which is close to user site to cope with the situation changing. By processing within lower layers, Tacit Computing can reduce network traffic amount and prevents leak of high privacy data. Here, we review live data discovering technology and device virtualization technology which are key technologies for Open IoT era.

Live data discovering technology discovers devices which provide necessary data for users. For example, there is an existing IoT application in which multiple sensors are attached to facilities such as bridges and degraded states are monitored continuously. In this case, it can not be thought that the facility degradation is going soon. Therefore, multiple points sensor data are collected to the cloud with a period of several hours, and the cloud analyzes the change of degraded state by statistical software or machine learning techniques. On the other hand, in an example where information guidance or warning alert are performed on a person appearing on a fixed point camera, only a few seconds the person appears in the camera, and only the image of the camera is meaningful to the person. In this way, data occurs in device layer and changes from moment to moment is called live data. To discover live data which is necessary for users, live data discovering technology distributes analyzing functions to lower layers instead of waiting for data to come up to cloud layer.

For example, we suppose a situation that a user's friend participates a marathon contest and a user wishes to see movies of cameras that the friend appears. In this case, bib number of

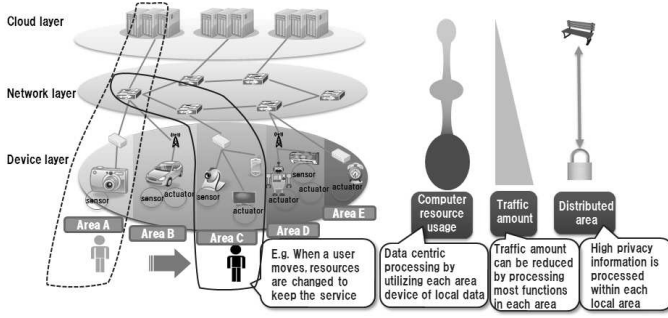


Fig. 1. Outline of Tacit Computing

the friend is requested for searching key to Tacit Computing. Tacit Computing distributes image analyzing functions such as OpenCV to gateways or network edges which accommodates cameras, and the functions analyze movies, extracts bib number of the friend by image analysis, then Tacit Computing can discover the camera which the friend appears at that time.

Next, Tacit Computing needs to use the device after it discovers the device with necessary user data. Because IoT devices are developed by many companies and each device has different protocol, interface, address and so on. Therefore, device virtualization technology conceals differences of each device interface and so on. In the above example, the method of use varies for each camera. Tacit Computing converts each request from common requests such as camera image acquisition to specific requests for individual camera by each device adapter on the gateways or network edges. By this method, users can use devices without considering each device difference. When we use devices, we can also use Semantic Web Services or other methods (e.g., [10]-[26]).

III. PROBLEMS TO BE SOLVED

Tacit Computing enables ad-hoc devices using for users by live data discovering technology and device virtualization technology. However, we think discovering and using the device based on the situation at that time by previous Tacit Computing only answers the user's needs only one time.

For example of Tacit Computing, let us consider tracking cameras. Tracking cameras are usage that movies of small children in schools or roads are taken by security cameras near the children and parents can see the movies by their mobile terminals. Tracking cameras can satisfy parents' needs to confirm children's safety at that time, but we think it is difficult to accept if it is charged as a certain price as once movie checking. We think it can be acceptable that it is charged as a monthly fee of continuous monitoring service in which parents can see movies of their children when they want to see on mobile terminals, but children are watched by machine learning or so on even when parents do not see and alerts are sent to parents in case of anomaly such as when a suspicious person approaches to children on roads.

As a processing of tracking cameras, Tacit Computing discovers the camera in which the child appears, and delivers

movies of the camera to the parents' mobile terminals when the parents request movies. And for watching, image analyzing functions such as OpenCV library are arranged on gateways or network edge SSE (Subscriber Service Edge) which accommodate cameras and image analyzing functions analyze images from the camera movie in which the child appears. The result of image analysis is summarized to the feature vector and it is aggregated to the cloud, and using cloud technologies (e.g., [27]-[46]), anomaly such as suspicious person approaching is analyzed by machine learning techniques (e.g., Local Outlier Factor). When anomaly is detected, alerts are sent to parents.

Tracking cameras can be achieved by not only Tacit Computing but also solution services using security cameras and machine learning techniques. Therefore, we set a problem to be solved as "providing continuous services that use devices discovered by Tacit Computing at reasonable prices".

In Tacit Computing, when we discover and use a device ad-hoc, we need to be able to use devices first and costs and performances are ignored. However, in order to provide it reasonably as a continuous service, it is necessary to reduce operation cost and improve performance.

Therefore, in this paper, we propose optimizations of appropriate function allocation and offloading specific processing in device, network and cloud layers. Optimizations are supposed to be performed during the first ad-hoc device usage period.

IV. PROPOSAL OF THREE LAYERS OPTIMIZATIONS

In this section, we propose optimization approaches to reduce operation cost and improve performance for continuous service with device using of Tacit Computing.

In device layer, it is necessary to switch a device that satisfies the needs of user at first. In the case of tracking cameras, it means to select the camera in which the image of the child appears based on location of the child. At this time, if we analyze images with all cameras connected to the network and use only the camera with the child appears, the cost will be very high. Thus, it is necessary to narrow down cameras near the child by identifying children with some identifiers and using cameras' metadata of location. Since many IoT devices are non-IP devices, we are considering to analyze communication patterns of IoT devices including non-IP devices and to assign metadata automatically or semi-automatically from the communication pattern.

Furthermore, in device layer, where to allocate analyze function on a gateway from gateways which accommodate devices affects operation costs much. In the case of tracking cameras, an image analysis function such as OpenCV is used. We are considering that image analysis functions are arranged in the gateways of the areas where children often pass, and when children move beyond that areas, image analysis functions are distributed at that times.

In network layer, as well as function placement for the gateways, we are considering optimizing function placement on network edges such as SSE. Because the caching and arrangement of information on the network is studied by information centric network, we consider to apply those ideas.

Depending on the service characteristic, bandwidth reservations, priority controls or other setting may be necessary in the network. For example, in the case of NGN (Next Generation Network), there is a function for reserving bandwidth called RACF (Resource and Admission Control Function), and such network side setting preparation also needs to be performed during the optimization processing period.

In cloud layer, where to process in the cloud greatly affects cost and performance. Firstly, we deploy the processing function to the cloud of the DC (Data Center) that has a small delay from the network edge that accommodates devices frequently used. Furthermore, since the size of the cloud resource also affects the operation cost, its resource size is also subject to optimization. For example, in the case of tracking cameras, the cloud analyzes summarized feature vectors by machine learning techniques and alerts suspicious person and so on. To detect anomaly in real time, stream processing functions such as Storm or Spark are used, so the cloud keeps appropriate resource sizes for these functions.

In addition, in recent days, servers with heterogeneous hardware such as GPU and FPGA are increasing in the clouds. By utilizing them, it is possible to achieve high performance by offloading matrix calculation and so on to GPU and offloading specific processing such as FFT (Fast Fourier Transformation) calculation to FPGA.

Furthermore, although we have described each of device layer, network layer and cloud layer so far, depending on the usage style of services, appropriate function placement is changed. Therefore, by modeling the cost (server cost for reserving resources and network cost based on generated traffic) and performance (throughput and delay) when each processing is processed in each layer, we are considering to calculate appropriate function placement. It is a combination optimization problem in order to optimize the performance with the price/cost for users as the constraint conditions, thus we adopt a method with a certain time calculation to find a certain degree of solution during limited optimization period.

REFERENCES

- [1] Y. Yamato, et al., "Predictive Maintenance Platform with Sound Stream Analysis in Edges," *Journal of Information Processing*, Vol.25, 2017.
- [2] Y. Yamato, "Proposal of Vital Data Analysis Platform using Wearable Sensor," *ICIAE* 2017, 2017.
- [3] Y. Yamato, et al., "Proposal of Lambda Architecture Adoption for Real Time Predictive Maintenance," *CANDAR* 2016, 2016.
- [4] Y. Yamato, et al., "Proposal of Real Time Predictive Maintenance Platform with 3D Printer for Business Vehicles," *International Journal of Information and Electronics Engineering*, Vol.6, No.5, 2016.
- [5] Y. Yamato, et al., "Security Camera Movie and ERP Data Matching System to Prevent Theft," *IEEE CCNC* 2017, 2017.
- [6] Y. Yamato, "Experiments of posture estimation on vehicles using wearable acceleration sensors," *BigDataSecurity* 2017, 2017.
- [7] Y. Yamato, et al., "Analyzing machine noise for real time maintenance," *ICGIP* 2016, 2016.
- [8] Y. Yamato, et al., "Proposal of Shoplifting Prevention Service Using Image Analysis and ERP Check," *IEEJ Transactions on Electrical and Electronic Engineering*, Vol.12, 2017.
- [9] J. Sasaki, et al., "Tacit Computing," *Business Communication*, Vol.54, No.5, 2017.
- [10] Y. Yamato, et al., "Context-Aware Service Composition and Component Change-over using Semantic Web Techniques," *IEEE ICWS* 2007, 2007.
- [11] Y. Yamato, et al., "Context-aware Ubiquitous Service Composition Technology," *CONFENIS* 2006, 2006.
- [12] Y. Nakano, et al., "Effective Web-Service Creation Mechanism for Ubiquitous Service Oriented Architecture," *IEEE CEC/EEE* 2006, 2006.
- [13] Y. Yamato, et al., "Study of Service Processing Agent for Context-Aware Service Coordination," *IEEE SCC* 2008, 2008.
- [14] Y. Nakano, et al., "Method of creating web services from web applications," *IEEE SOCA* 2007, 2007.
- [15] Y. Yamato, "Method of Service Template Generation on a Service Coordination Framework," *UCS* 2004, 2004.
- [16] Y. Yokohata, et al., "Service Composition Architecture for Programmability and Flexibility in Ubiquitous Communication Networks," *IEEE SAINTW'06*, 2006.
- [17] M. Takemoto, et al., "Service Elements and Service Templates for Adaptive Service Composition in a Ubiquitous Computing Environment," *APCC* 2003, Vol.1, 2003.
- [18] Y. Miyagi, et al., "Support System for Developing Web Service Composition Scenarios without Need for Web Application Skills," *IEEE SAINT'09*, 2009.
- [19] Y. Nakano, et al., "Web-Service-Based Avatar Service Modeling in the Next Generation Network," *APSITT* 2008, 2008.
- [20] Y. Yokohata, et al., "Context-Aware Content-Provision Service for Shopping Malls Based on Ubiquitous Service-Oriented Network Framework and Authentication and Access Control Agent Framework," *IEEE CCNC* 2006, 2006.
- [21] Y. Yamato, et al., "Study and Evaluation of Context-Aware Service Composition and Change-Over Using BPEL Engine and Semantic Web Techniques," *IEEE CCNC* 2008, 2008.
- [22] M. Takemoto, et al., "Service-composition Method and Its Implementation in Service-provision Architecture for Ubiquitous Computing Environments," *IPSJ Journal*, Vol.46, No.2, 2005.
- [23] H. Sunaga, et al., "Ubiquitous Life Creation through Service Composition Technologies," *World Telecommunications Congress 2006 (WTC 2006)*, 2006.
- [24] Y. Yamato, et al., "Development of Service Control Server for Web-Telecom Coordination Service," *IEEE ICWS* 2008, 2008.
- [25] H. Sunaga, et al., "Service Delivery Platform Architecture for the Next-Generation Network," *ICIN2008*, Session 9-A, Oct. 2008.
- [26] Y. Yamato, et al., "Context-aware Service Composition and Change-over Using BPEL Engine and Semantic Web," *INTECH Open Access Publisher*, 2010.
- [27] Y. Yamato, et al., "Fast and Reliable Restoration Method of Virtual Resources on OpenStack," *IEEE Transactions on Cloud Computing*, 2015.
- [28] Y. Yamato, "Server Selection, Configuration and Reconfiguration Technology for IaaS Cloud with Multiple Server Types," *Journal of Network and Systems Management*, Springer, 2017.
- [29] Y. Yamato, "OpenStack Hypervisor, Container and Baremetal Servers Performance Comparison," *IEICE Communication Express*, Vol.4, No.7, 2015.
- [30] Y. Yamato, "Cloud Storage Application Area of HDD-SSD Hybrid Storage, Distributed Storage and HDD Storage," *IEEJ Transactions on Electrical and Electronic Engineering*, Vol.11, No.5, 2016.
- [31] Y. Yamato, "Performance-Aware Server Architecture Recommendation and Automatic Performance Verification Technology on IaaS Cloud," *Service Oriented Computing and Applications*, Springer, 2016.
- [32] Y. Yamato, "Server Structure Proposal and Automatic Verification Technology on IaaS Cloud of Plural Type Servers," *NETS* 2015, 2015.
- [33] Y. Yamato, et al., "Survey of Public IaaS Cloud Computing API," *IEEJ Transactions on Electronics, Information and Systems*, Vol.132, 2012.
- [34] Y. Yamato, "Implementation Evaluation of Amazon SQS Compatible Functions," *IEEJ Transactions on Electronics, Information and Systems*, Vol.135, 2015.
- [35] Y. Yamato, et al., "Fast Restoration Method of Virtual Resources on OpenStack," *IEEE CCNC* 2015, 2015.
- [36] Y. Yamato, "Automatic verification technology of virtual machine software patch on IaaS cloud," *WASET, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, Vol.9, No.1, 2015.
- [37] Y. Yamato, "Automatic verification for plural virtual machines patches," *ICUFN* 2015, 2015.
- [38] Y. Yamato, "Use case study of HDD-SSD hybrid storage, distributed storage and HDD storage on OpenStack," *IDEAS* 2015, 2015.
- [39] Y. Yamato, et al., "Development of Resource Management Server for Production IaaS Services Based on OpenStack," *Journal of Information Processing*, Vol.23, No.1, 2015.
- [40] Y. Yamato, "Automatic verification technology of software patches for user virtual environments on IaaS cloud," *Journal of Cloud Computing*, Springer, 4:4, 2015.
- [41] Y. Yamato, "Automatic system test technology of user virtual machine software patch on IaaS cloud," *IEEJ Transactions on Electrical and Electronic Engineering*, Vol.10, Issue.S1, 2015.
- [42] Y. Yamato, et al., "Evaluation of Agile Software Development Method for Carrier Cloud Service Platform Development," *IEICE Transactions on Information & Systems*, Vol.E97-D, No.11, 2014.
- [43] Y. Yamato, et al., "Software Maintenance Evaluation of Agile Software Development Method Based on OpenStack," *IEICE Transactions on Information & Systems*, Vol.E98-D, No.7, 2015.
- [44] Y. Yamato, "Proposal of Optimum Application Deployment Technology for Heterogeneous IaaS Cloud," *WCSE* 2016, 2016.
- [45] Y. Yamato, "Optimum Application Deployment Technology for Heterogeneous IaaS Cloud," *Journal of Information Processing*, Vol.25, No.1, 2017.
- [46] Y. Yamato, et al., "Development of Template Management Technology for Easy Deployment of Virtual Resources on OpenStack," *Journal of Cloud Computing*, Springer, 3:7, 2014.