Process-Aware Internet of Things: A Conceptual Extension of the Internet of Things Framework and Architecture

Meesun Kim[1], Hyun Ahn[1], and Kwanghoon Pio Kim[1*]
[1] Collaboration Technology Research Laboratory
Department of Computer Science
KYONGGI UNIVERSITY
154-42 Kwangkyosan-ro Youngtong-gu Suwon-si Gyeonggi-do, 16227, South Korea
[e-mail: mskim, hahn, kwang@kgu.ac.kr]
*Corresponding author: Kwanghoon Pio Kim

Received April 12, 2016; revised July 5, 2016; revised August 26, 2016; accepted August 31, 2016; published August 31, 2016

Abstract

This paper tries to extend the conventional conceptual framework of the Internet of Things (IoT) so as to reify an advanced pervasive IoT-community collaboration concept, which is called the *process-aware Internet of Things*. The extended conceptual framework is embodied as a referential architecture that can be a standardized reference model supporting the conceptual integration of the Internet of Things and the process awareness. The extended referential architecture covers the full range of the architectural details from abstracting the process-aware behavioral semantics to reifying the IoT-process enactments. These extended framework and architecture ought to be the theoretical basis for implementing a process-aware IoT-community computing system supporting process-aware collaborations of Things in pervasive computing environments. In particular, we do point up that the proposed framework of the process-aware Internet of Things is revised from the Internet of Things framework announced in ITU-T SG13[3] Y.2060 [26] by integrating the novel concept of process awareness. We strongly believe that the extended conceptual framework and its referential architecture are able to deliver the novel and meaningful insight as a standardized platform for describing and achieving the goals of IoT-communities and societies.

Keywords: pervasive process model, the Internet of Things, the Web of Things, process-aware smart-object collaboration, ITU-T IoT framework and architecture, process-aware Internet of Things

A preliminary version [25] of this paper was presented at ICONI 2015, and was selected as an outstanding paper. This work was supported by the KGU Research Grant (Grant No. 2015-083) funded by Kyonggi University, Republic of Korea.

http://dx.doi.org/10.3837/tiis.2016.08.032 **ISSN : 1976-7277**

1. Introduction

Currently, we are in the middle of an era of pervasive computing and mobile environments. The widespread use of mobile communications is evident. These gadgets have become an integral and intimate part of everyday life for many millions of people, even more than the internet. This phenomenon is igniting a significant development of future platforms and networks to radically transform our corporate, community, and personal spheres. Embedding short-range mobile transceivers into a wide array of additional gadgets and everyday items has created a new form of pervasive computing platform through mobile communications between people and Things and between Things themselves. This development is able to support a new dimension of connectivity: anyone at any time in any place can have connectivity with the world of information and communications technologies. As a consequence, a new form of pervasive computing platform with three dimensions of connectivity has created an entirely new and dynamic network of networks—the Internet of Things (IoT) [3][17][20][22]—as a pervasive computing platform.

The collaborative work of IoT-related study groups and focus groups of ITU-T such as SG13, SG16, JCA-IoT, and IoT-GSI [5][6] has recently created the conceptual foundation and architectural reference models [3][11][13][18][20][23][24] needed for the Internet of Things. The Internet of Things reference model is an architectural framework to build a pervasive and mobile community computing environment that provides a computerized situation or space formed by a group of smart objects (so-called Things) such as devices, sensors, actuators, and even people, each of which may have various computing capabilities and/or mobile networking capabilities. However, no feasible modeling methodologies to describe, control, and achieve the goals to be accomplished by a group of collaborative smart objects in the computerized situation or space has been developed.

Therefore, we propose an advanced and novel pervasive community computing platform, which is called ***process-aware Internet of Things***, by extending the conventional conceptual architecture of the Internet of Things. The proposed concept is originated from the concept of process-driven Internet of Things firstly announced by the authors' research group through [11][18][25], and it is a methodology to describe, control, and achieve community goals controlled and supervised under the IoT-based infrastructure and environments. Imagine that a group of community members (smart objects or Things) in an IoT-based pervasive community performs their own roles to accomplish the community's goals in process-aware collaboration. For the sake of this situation, we revise the ITU's standardized IoT framework [3][5][6] in a different way. In other words, we need to revise the ITU's standardized IoT framework and integrate the concept of process-aware collaborations [1][18] into the standardized IoT conceptual architecture [5][24]. These two tasks are non-trivial. This paper gives feasible solutions for each of these tasks by proposing a novel community computing model and its conceptual architecture (*i.e.*, the pervasive process model and architecture, respectively), which are applicable to implementing this process-aware IoT-community computing platform and system.

We begin the paper by introducing related studies from the literature. We also give a conceptual description of the process-aware IoT-community framework extended from the standardized IoT framework announced by the ITU-T. In the following two sections, we present detailed functional descriptions of the proposed platform, process-aware IoT, by formally and graphically defining the pervasive process model and its conceptual architecture. Finally, we summarize the implications of the proposed platform with an application example.

2. Related Work and the Scope

Thus far, there exist several pervasive (or ubiquitous) community computing models [6][8] such as the context-aware community collaboration model [6], the society collaboration model [5][8], and the member collaboration model [1][8] for building an IoT-based pervasive community computing environment [11][13][22]. The context-aware community collaboration model [6] supports a community goal through a series of situations from the initiating situation to the terminating situation in the corresponding community. Each of the member-objects fulfills its assigned role in a specific situation in the community. It is assumed that the defined community goal is accomplished if all of the member-objects achieve their assigned goals. Additionally, context-aware community collaboration models can be classified into the three levels of intellectualized community, i.e., the simple collaboration model, the dynamic collaboration model, and the autonomous collaboration model, according to the degree of intellectualization, the extent of which can be determined by considering both the context-aware capability and the role-aware capability of the member-objects organized in the corresponding IoT-based pervasive community.

The context-aware community computing model uses the society collaboration model [8] to define a group of activities, which is executable in a pervasive (or ubiquitous) intellectual mobile-space. The society collaboration model defines a specific society consisting of a group of smart objects (or Things), each of which is called a society member, including humans, and software and hardware components (such as applications, services, sensors, and actuators). In a specific society formed on a pervasive intellectual mobile-space, a variety of communities can be organized; each community becomes a virtual team to collaboratively accomplish the target goal (or perform the chosen service) in the corresponding community. Managing the member-objects in the society and its communities is described by the member collaboration model, and is performed via a memberfication procedure of the member collaboration framework. However, the literature still needs more sophisticated methodologies for supporting the description and specifications of a collaborative goal to be achieved by a group of members in a specific IoT-based pervasive community or society. Therefore, as the scope of the paper, we focus on proposing a semantic and procedural methodology, which is called the *pervasive process model*, not only for formally and graphically describing a collaborative process of roles but also to allow the described pervasive processes to be performed by a group of members in the corresponding process-aware IoT-based community.

3. An Extended Conceptual Architecture

In this section, we propose an advanced conceptual architecture for organizing IoT-based pervasive communities and societies, namely, process-aware Internet of Things, and the pervasive process model for formally describing the collaborative behavior of a process-aware pervasive community on the process-aware Internet of Things environment. The conceptual architecture is devised to integrate the concept of process-aware collaborative communities into the standardized framework [3] of the Internet of Things. Before examining the details of the proposed conceptual architecture, we start from the basic concept of a process-aware pervasive community computing environment, which is the foundation of the process-aware Internet of Things. The process-aware pervasive community computing environment is a computerized situation or society in which a partial group of the member-objects is able to organize a process-aware collaborative community (statically, dynamically, or autonomously) to accomplish its goal, and disorganize the community into the corresponding society after completing the goal. To organize and manage a process-aware pervasive community computing environment, we must revise the ITU-T's standardized architectural framework [26] for the Internet of Things and propose the concept of pervasive process model as a means for describing and the concept of achieving process-aware community goals.

3.1 The ITU-T Conceptual Architecture

In terms of the architectural framework of the IoT, the ITU-T's study groups and focus groups such as SG13, SG16, JCA-IoT, and IoT-GSI [5][6][26] provided the concept and the architectural reference model [3] of the IoT. As shown in **Fig. 1** [26], the conceptual architecture of the IoT is made up of the physical world, the information world, and the map of connections between these two worlds. The physical world builds a pervasive and mobile community computing environment and provides a computerized situation or space formed by a group of physical Things (such as devices, sensors, and actuators), each of which may have various computing capabilities and/or mobile networking capabilities. The information world consists of a group of virtual Things (such as smart objects, web objects, and data archives) mapped to (or gathered from) the physical Things. Organizing pervasive communities over the physical world and the information world can be managed by the context-aware community collaboration model [6][8], the society collaboration model [8], or the member collaboration model [8]. The infrastructure of IoT can be a platform for the Web of Things (WoT) [5] or the Web of Objects (WoO) [5], thus forming pervasive communities, which have been recently described in the RFID and USN (ubiquitous sensor network) literature. The pervasive community and society computing concept [8] implies a computerized situation or space formed from the physical world and the information world.

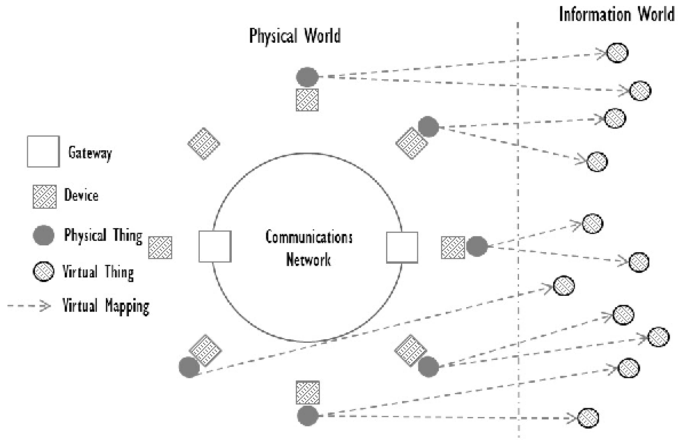

Fig. 1. The ITU's Conceptual Architecture of the Internet of Things

We propose a new, advanced pervasive community computing concept, which is called process-aware Internet of Things, to be deployed on IoT-based infrastructure and platforms. Imagine that a group of community members (smart objects or Things) in an IoT-based pervasive community performs their respective roles to accomplish the community's goal using process-aware collaborations. To realize this concept, we need to address the following issues:

– We must extend the ITU's standardized IoT framework.
– We must embed the concept of process-aware collaborations into the standardized IoT conceptual architecture.

This paper provides feasible solutions to these issues by proposing a pervasive process model and an extended conceptual architecture for the process-aware IoT platform that can be implemented as a process-aware IoT-community computing system and environment.

3.2 An Extended Conceptual Architecture

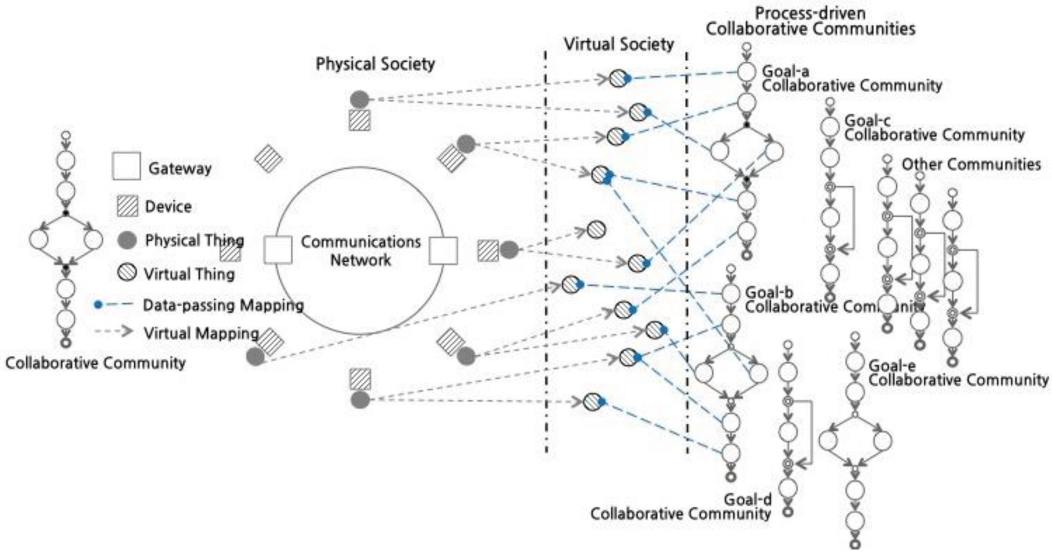

Fig. 2. A Conceptual Framework of the Process-Aware IoT-Community Computing Environment

For the basic concept, **Fig. 2** depicts the proposed conceptual framework that forms a process-aware pervasive IoT-community computing environment based on the ITU-T's standardized IoT framework. As shown in the figure, the conceptual platform consists of the physical society, the virtual society, and process-aware collaborative communities. The physical society is arranged into a networked group of physical Things including devices, sensors, actuators, and even people. The virtual society is formed by a group of virtual or logical Things that is mapped with physical Things via the Internet. A pervasive IoT-community computing environment is physically configured by a set of equipment (such as devices, sensors, and actuators) on a pervasive and mobile communication networking environment. Each of the configured devices can have embedded computing abilities with the characteristics of smart objects, which are physical Things. Additionally, each of the physical Things is mapping to (and communicating with) one or more virtual Things in the virtual society to match the roles to the member-objects (the physical Things) to accomplish the goal of the process-aware collaborative community. To successfully reach the IoT-community's goal, a reasonable means to describe and specify the procedural activities (and their order of execution) that should be performed by the physical Things and the virtual Things of the corresponding community is needed.

In this paper, therefore, we propose a novel modeling methodology, which is called the pervasive process model, to describe a process-aware goal that is comprised of a set of procedural activities and their execution orders of the physical Things and the virtual Things in the process-aware IoT-community computing environment. We devise an architectural reference model to reify the process-aware Internet of Things platform as a process-aware pervasive IoT-community computing system.

3.3 The Pervasive Process Model

We describe the pervasive process model by constructing a generic meta-model, which is used to define a process-aware IoT-community goal that can be accomplished by a group of member-Things belonging to the corresponding community.

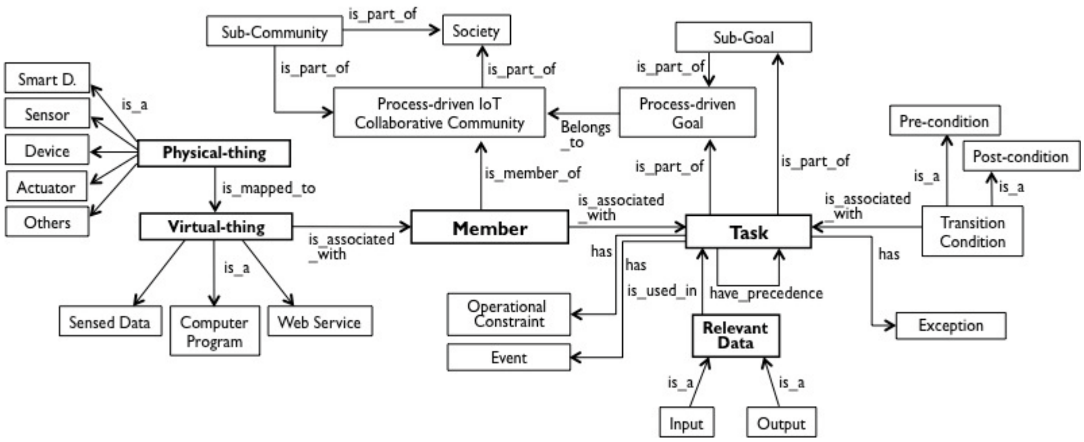

Fig. 3. Meta-model of the Pervasive Process Model

Fig. 3 shows the generic meta-model in the form of an entity-relationship diagram, and defines the major entity-types and their relationships, which constitute a pervasive smart space, such as a pervasive society and a community. In the figure, rectangles and arrows indicate entity types and relationships, respectively, and all of the major entity-types used to form a process-aware IoT-community goal are listed as process-aware goals, tasks (or roles), smart members, relevant data, and Things (physical and virtual). The following section provides detailed explanations of the major entity types:

- The process-aware goal is defined by a predefined or intended set of tasks or roles, called activities, and their temporal ordering of executions. A pervasive IoT-community computing system helps to organize, control, and execute process-aware IoT-community goals defined by the pervasive process model proposed in this paper. A process-aware goal can be described by a set of temporal orders of the associated activities through combinations of sequential logic, conjunctive logic (after task A, do tasks B and C), disjunctive logic (after task A, do task B or C), and loop logic. The completion of the process-aware goal implies the accomplishment of the targeted goal of the IoT-community such that all of the tasks (or activities) are successfully performed in compliance with the corresponding pervasive process model.
- The task is a semantic entity of the basic unit of work (activity or role). All of the tasks in a process-aware goal are arranged as a set of precedence relationships to specify their execution sequences. Additionally, a task can be precisely specified by one of the task

entity types: atomic task, compound task, trigger task, repetitive task [12], loop-block task [1], multiple threads task [12], or gateway task (such as OR-split, OR-join, AND-split, AND-join, XOR-split, and XOR-join). A compound task is a task containing another process-aware goal, which is called a sub-goal. The atomic task is the task that is performed by a physical Thing through a syntactic mapping with smart objects (virtual Things). Specifically, a pair of split-join gateway tasks controls the execution flows of the associated atomic and/or compound tasks.

- The member is a conceptual entity type representing the virtual Things, each of which performs a certain role in completing a corresponding process-aware goal. A member is associated with one or more virtual Things (smart objects) that are embodied as physical Things. In the end, the member will be bound with an actual Thing, such as person, program, sensor, actuator, device, or service, which can fulfill the designated role or be responsible for completing the corresponding process-aware goal.
- Finally, the condition is an entity-type that is required to perform the tasks of the process-aware goal. There are two different types of conditions: relevant-data conditions and transition conditions. The relevant-data conditions of each task are matched to input and output data that are required as pre-conditions and post-conditions, respectively, for the execution of the task, whereas the transition conditions on the alternative gateway edges are needed to select control-paths on the disjunctive gateway task or the loop gateway task. The relevant-data condition provides a communication channel between the engine of the process-aware IoT-community computing system and the smart-object bound to the corresponding task. The transition condition itself is specified using the relevant input/output data.

The Formal Description. Based on the meta-model, a pervasive process model can be defined by defining the goals, tasks and their control precedence, members, and relevant data/transition conditions, as described in the previous section. In this subsection, we give a formal description of the pervasive process model that describes process-aware collaborative relationships among the community members using a series of semantic perspectives such as the control flow perspective, the data flow perspective, the resource assignment perspective, and the exception perspective. The following definition, Definition 1, is a formal definition of the pervasive process model and its functional components for expressing each of the semantic perspectives.

Definition 1: The Pervasive Process Model (PPM) is used to formally define the process-aware pervasive IoT-community goals. The basic PPM is defined by an 8-tuple formula: $\boldsymbol{\Gamma} = (\delta, \rho, \gamma, \lambda, \pi, \kappa, \mathbf{I}, \mathbf{O})$, over a set of **A** tasks (including a set of sub-goals), a set of $\mathbf{E} \subseteq (\mathbf{A} \times \mathbf{A})$ edges (pairs of tasks), a set **T** of transition conditions, a set **R** of relevant-data variables, a set **V** of virtual Things, a set **G** of physical Things, a set **P** of members, and a set **C** of exceptions, where $\mathsf{P}(\mathbf{A})$ represents a power set of **A**:

- **I** is a finite set of initial input relevant-data variables, assumed to be loaded with information by external process-aware goals before execution of the PPM;
- **O** is a finite set of final output relevant-data variables, perhaps containing information used by some external process-aware goals after execution of the PPM;
- $\delta = \delta_i \cup \delta_o$ /* Control Flow Perspective */

  where $\delta_o : \mathbf{A} \rightarrow \mathsf{P}(\mathbf{A})$ is a multi-valued mapping function from a task to its sets of (immediate) successors, and δ_i : **A** → P(**A**) is a multi-valued mapping function from a task to its sets of (immediate) predecessors;

– $\rho = \rho_i \cup \rho_o$ /* Data Flow Perspective */

where ρ_o : **A** → P(**R**) is a single-valued mapping function from a task to its set of output relevant-data variables, and ρ_i : **A** → P(**R**) is a single-valued mapping function from a task to its set of input relevant-data variables;

– $\gamma = \gamma_i \cup \gamma_o$ /* Data Flow Perspective */

where γ_o : **R** → P(**A**) is a single-valued mapping function from a relevant-data variable to its set of out-degree tasks, and γ_i : **R** → P(**A**) is a single-valued mapping function from a relevant-data variable to its set of in-degree tasks;

– $\lambda = \lambda_a \cup \lambda_p$ /* Resource Assignment Perspective */

where λ_p : **A** → P(**P**) is a single-valued mapping function from a task to its members, and λ_a : **P** → P(**A**) is a single-valued mapping function from a member to its set of associated tasks;

– $\pi = \pi_v \cup \pi_p$ /* Resource Assignment Perspective */

where π_p : **V** → P(**P**) is a single-valued mapping function from a virtual Thing to its set of associated members, and π_v : **P** → P(**V**) is a single-valued mapping function from a member to its set of associated virtual Things;

– $\iota = \iota_g \cup \iota_v$ /* Resource Assignment Perspective */

where ι_v : **G** → P(**V**) is a single-valued mapping function from a physical Thing to its set of associated virtual Things, and ι_g : **V** → P(**G**) is a single-valued mapping function from a virtual Thing to its set of associated physical Things;

– $\kappa = \kappa_i \cup \kappa_o$ /* Data Flow Perspective */

where κ_i : **E** → P(**T**) is a single-valued mapping function from an edge to a set of transition conditions, and κ_o : **T** → P(**E**) is a single-valued mapping function from a transition condition to a set of edges.

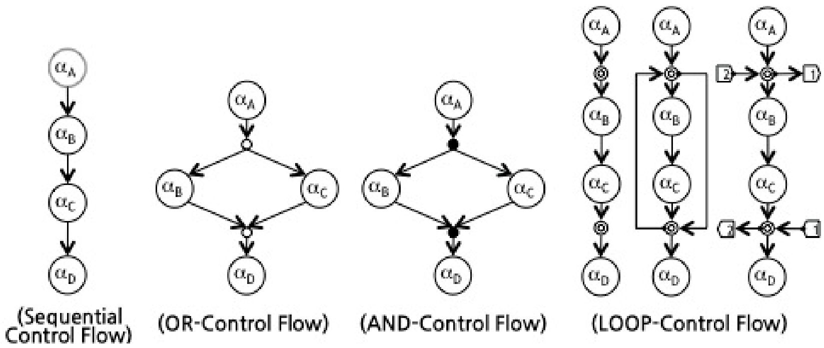

Fig. 4. Control Flow Primitives of the Pervasive Process Model

Control Flow Semantics. Given the formal definition, the temporal ordering of tasks (activities) in a pervasive process model can be interpreted as follows: For any task α, in general,

$$\delta(\alpha) = \{ \{\beta_{11}, \beta_{12}, \ldots, \beta_{1m(1)}\}, \{\beta_{21}, \beta_{22}, \ldots, \beta_{2m(2)}\}, \ldots, \{\beta_{n1}, \beta_{n2}, \ldots, \beta_{nm(n)}\} \}$$

which means that upon completion of task α, a control transition that simultaneously initiates all of the tasks β_{i1} through $\beta_{im(i)}$ occurs, which is called a parallel control-flow; otherwise, only one value of i ($1 \le i \le n$) is selected as the result of a decision made within the task α, which is called a decision control transition. Note that if $n = 1 \wedge m = 1$, then neither a decision nor parallel processing is needed after completion of task α, which means that the transition is a sequential control transition. Additionally, if $m(i) = 1$ for all i, then no parallel processing is initiated on completion of α.

Based on this interpretation, we graphically present these primitive control transition types as shown in **Fig. 4**. A task with a conjunctive (or parallel) control transition is represented by a solid dot (●), and a task with a disjunctive (or decision) control transition is represented by hollow dot (○). These special types of tasks are called gateway tasks. To be syntactically safe, it is very important for these gateway tasks to retain structured properties such as proper nesting and matched pair properties. Not only do each of the gateway tasks maintain matched pairs with split and join types of gateway task in a pervasive process model, but also multiple sets of gateway tasks remain in a properly nested pattern. In summary, the following statements formally describe the basic control-transition types modeled by the exclusive-OR and AND gateway tasks depicted in **Fig. 4**.

(1) Sequential control-transition between tasks

incoming → $\delta_i(\alpha_B) = \{\{\alpha_A\}\}$; outgoing → $\delta_o(\alpha_B) = \{\{\alpha_c\}\}$;

(2) Exclusive OR control-transition through XOR-gateway

XOR-split → $\delta_o(\alpha_A) = \{\{\alpha_B\}, \{\alpha_C\}\}$; XOR-join → $\delta_i(\alpha_D) = \{\{\alpha_B\}, \{\alpha_C\}\}$;

(3) AND control-transition through AND-gateway

AND-split → $\delta_o(\alpha_A) = \{\{\alpha_B, \alpha_C\}\}$; AND-join → $\delta_i(\alpha_D) = \{\{\alpha_B, \alpha_C\}\}$;

Loop-block Task. We must carefully define the iterative (loop) control transition, which is essential and is a common construct in modeling of the temporal ordering of tasks. We must graphically define the iterative control-transition gateway task as a pair of double-hollow dots of gateway tasks, as shown in the right-hand side of **Fig. 4**. At a glance, it can be interpreted as a special type of disjunctive control-transition task; however, if we replace this transition task with a disjunctive control-transition task, it is very difficult to maintain the structured properties (i.e., matched pairs and proper nesting) in the pervasive process model. Therefore, we introduce the concept of the loop-block task to retain the structured properties in the modeling of the pervasive process model. The LOOP-block task contains two gateways (i.e., LOOP-split and LOOP-join tasks), and temporally orders the tasks inside of the LOOP-split and LOOP-join gateway tasks. Regarding the loop-gateway tasks, we must specify the loop's exit conditions in the modeling time. Accordingly, the formal definition of a LOOP-block task's gateways, shown in **Fig. 4**, is as follows:

– LOOP-Split Gateway → δ_i ($\alpha_{loop-split}$) = {{α_A}}; δ_o ($\alpha_{loop-split}$) = {{α_B}};
– LOOP-Join Gateway → δ_i ($\alpha_{loop-join}$) = {{α_C}}; δ_o ($\alpha_{loop-join}$) = {{α_D}};

Data Flow Semantics. The data flow perspectives represent the effects of input- and output-relevant data needed by each of the tasks in a pervasive process model. From the specified data flow perspectives, the data transitions of each task and the data dependencies between tasks are defined by analyzing the definitions (outputs or writes) and uses (inputs or reads) of each relevant-data variable in the pervasive process model. $\rho = \rho_i \cup \rho_o$ describes the in-data transitions (uses, ρ_i) and the out-data transitions (definitions, ρ_o) of each task, whereas $\gamma = \gamma_i \cup \gamma_o$ expresses the in-data dependencies (γ_i) and the out-data dependencies (γ_o) between tasks via a specific relevant-data variable.

The data-flow paths in the pervasive process model can be constructed as follows: Let α and β be tasks in a pervasive process model. There is a data flow path from α to β with respect to a relevant-data variable η *iff* (1) there exists a control-flow path from α to β and (2) the model satisfies one of the following conditions:

– $\eta \in \rho_o(\alpha) \wedge \eta \in \rho_i(\beta)$: α contains a definition of η, and β contains a use of η;
– $\eta \in \rho_i(\alpha) \wedge \eta \in \rho_o(\beta)$: α contains a use of η, and β contains a definition of η;
– $\eta \in \rho_o(\alpha) \wedge \eta \in \rho_o(\beta)$: α contains a definition of η, and β contains a definition of η.

Resource Assignment Semantics. The resource assignment perspectives are represented by $\lambda = \lambda_a \cup \lambda_p$, $\pi = \pi_v \cup \pi_p$, and $\iota = \iota_g \cup \iota_v$, which refer to the task-member assignments, member-virtual-Things assignments, and the virtual-physical-Things assignments, respectively. As defined in the previous section, the pervasive IoT-community computing environment might be supported by a variety of special tasks such as multiple-thread tasks, event tasks, trigger tasks, repetitive tasks, and gateway tasks. Every special task should be supported by suitable resource assignment policies and functions. Specifically, multiple-thread tasks are characterized as either homogenous multiple-thread tasks or heterogeneous multiple-thread tasks; each thread that is spawned from a heterogeneous multiple-thread task must be assigned to a different type of virtual Things.

4. A Standardization Reference Model

This section describes a standardization reference model that can be possibly revised from the ITU-T's functional architecture [26] for supporting the proposed type of IoT-community computing architecture to be implemented as a process-aware Internet of Things platform. Imagine that a group of community members (smart objects or Things) in an IoT community performs their own roles to accomplish the community's goal supported through process-aware collaborations of the community-members. We must describe, specify, and enact the process-aware collaborations for the community-members to achieve the community's goals. We have already presented a novel concept of the process-aware Internet of Things to solve this issue by proposing an extended conceptual platform and its referential architecture to implement a process-aware IoT-based pervasive community computing system and environment.

The process-aware IoT-based pervasive community computing environment is a computerized situation and society in which a group of Things, including smart objects such as devices, sensors, actuators, and even people, is connected via the Internet, and a partial group of the members is able to organize a collaborative community (statically, dynamically, or autonomously) to accomplish its goals, and disorganize the community into the corresponding society after completing the goals. To organize and manage this IoT-based pervasive community-computing environment, ITU-T announced the standardized architectural reference model of the Internet of Things [3][26]; we try to advance the architectural framework by applying the concept of process automation [1][7][18][23][25] as a means for describing, specifying, and achieving community goals.

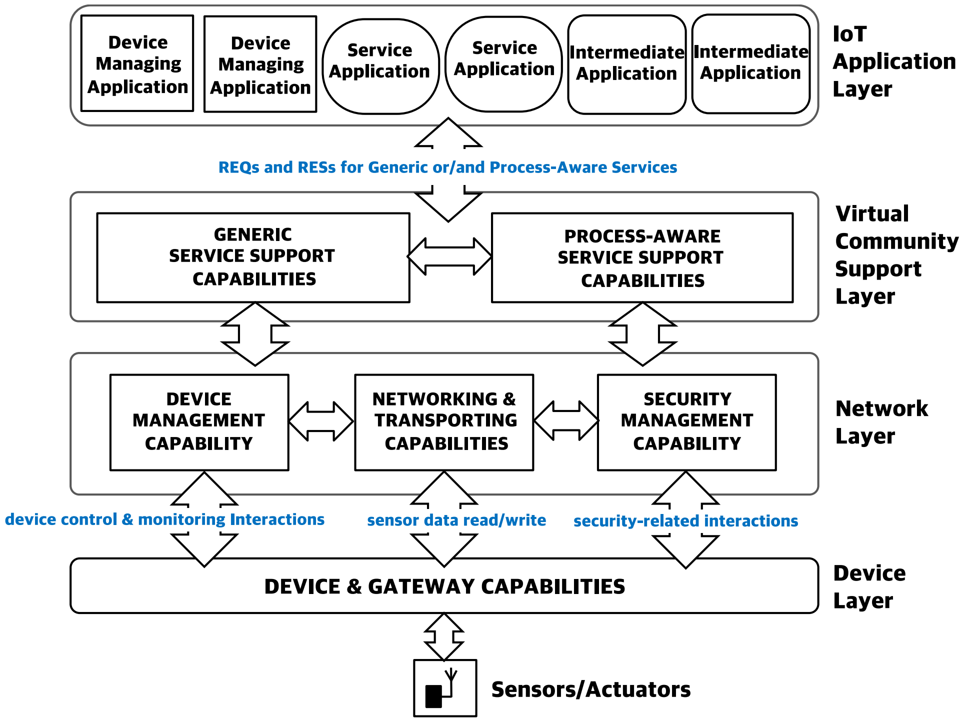

Fig. 5. A Standardization Reference Model for the Process-Aware Internet of Things Architecture

Fig. 5 illustrates the proposed functional reference model that forms the process-aware IoT-community computing environment based on the ITU-T's standardized IoT framework [5][6][26]. As shown in the figure, the process-aware IoT reference model comprises four layers such as application, virtual community service/application support, network, and device layers, and these layers are supported by management capability as well as security capability. The conceptual descriptions of the layers are as follows:

– The IoT application layer contains three groups of IoT applications such as device managing applications, intermediate applications, and generic/specific service applications including process-aware service applications.
– The virtual community service/application layer consists of two capability groupings: generic service support capabilities and process-aware service support capabilities. The generic service support capabilities are common capabilities which can be commonly requested by all the IoT application groups, such as data processing, data storing, or data gaining, and these capabilities may be also invoked by the process-aware service support capabilities. The process-aware service support capabilities are such specific capabilities which provide for the requirements of process-aware IoT community-computing environments.
– The network layer basically comprises two types of functional capabilities such as networking

capabilities and transporting capabilities, and two types of managerial capabilities such as device management capabilities and security management capabilities. The networking capabilities [26] maintain relevant network connectivity, such as access and transport resource control functions, mobility management, and authentication/authorization/accounting (AAA). The transport capabilities [26] provide connectivity for transporting not only the IoT service and application information but also the IoT-related control and management information. Additionally, the management capabilities consist of the device management [26] and the security management [26] capabilities supporting the requests from the upper layer and the lower layer as well as from the networking and transporting management services such as local network topology management and traffic and congestion management. Especially, the security management capabilities [26] support authorization, authentication, data confidentiality and integrity protection, privacy protection, security audit, anti-virus, access control, device integrity validation, and others, which are requested from the application layer, virtual community support layer, network layer, and the device layer.

– The device layer is logically classified into two types of capabilities such as device capabilities and gateway capabilities. The device capabilities are able to support direct interaction and indirect interaction with the communication network, *ad hoc* networking, and sleeping and wake-up mechanisms for energy-savings. The gateway capabilities include the multiple interfaces through different kinds of wired or wireless technologies (controller area network bus, ZigBee, Bluthooth or Wi-Fi, public switched telephone network, 2G/3G networks, long-term evolution (LTE) networks, Ethernet, or digital subscriber lines), and the protocol conversion between the device layer and network layer.

After all, the IoT-community computing environment is physically configured using a group of devices such as sensors and actuators in a pervasive community networking environment, and each of the configured devices may have embedded computing abilities with the characteristics of smart objects, which are physical Things. Additionally, each of the physical Things is mapped to (and communicating with) one or more virtual Things in the information community to virtually match the roles to the member objects to accomplish the goal of the community. However, to accomplish the IoT-community's goal, a reasonable means to describe and specify the procedural activities and the order of their execution that should be performed by the physical Things and the virtual Things of the corresponding community is needed. Therefore, we propose a conceptual platform adopting the concept of pervasive processes, which is depicted in the right-hand side of **Fig. 2**. Assume that the circled nodes of the pervasive process are matched to the nodes of the information community, and that the pervasive process is used to specify the order of virtual-Things' executions using a modeling methodology such as the process-aware IoT-community computing model.

5. Conclusion

In the paper, we present a method of achieving a process-aware goal using a pervasive collaboration community organized on the infrastructure of the process-aware Internet of Things. That is, the paper proposes a new methodology for formally describing the process-aware goal, which is dubbed the process-aware IoT-community computing model. Additionally, we revised the ITU-T's standardized conceptual architecture of the Internet of Things by including the concept of process automation, and propose a standardization reference model to implement process-aware IoT-community computing systems. In conclusion, the main contribution of the paper ought to be on the novel framework and architecture for accomplishing process-aware IoT-community computing environments where we enable to formally and graphically describe a process-aware goal of the Internet of Things' community as a service. In future work, we must add entity types such as events, exceptions, and operational constraints to the pervasive process model proposed in the paper.

Acknowledgement

This work was supported by the KGU research grant (Grant No. 2015-083) funded by Kyonggi University, Republic of Korea. The authors would transfer our thanks to the colleagues and the institutions for sponsoring this collaborative research. Particularly, appreciation is extended to Dr. Minjae Park, Dr. Haksung Kim, and the graduate students in the collaboration technology research laboratory of Kyonggi University.

References

[1] K. Kim, C. A. Ellis, "Section II / Chapter VII. An ICN-based Workflow Model and Its Advances," *Handbook of Research on BP Modeling*, IGI Global, ISR, pp. 142-172, 2009. Article (CrossRef Link).

[2] N. Clarles Russell BSc MinfTech, "Foundations of Process-aware Information Systems," Ph.D. Dissertation, Queensland University of Technology, 2007.

[3] "ITU Internet Reports 2005: The Internet of Things," Executive Summary, International Telecommunication Union, 2005.

[4] M. Skerlavaj, V. Dimovski, K. C. Desouza, "Patterns and Structures of Intra-organizational Learning Networks within a Knowledge-intensive Organization," *Journal of Information Technology*, Vol. 25, No. 2, pp. 189-204, 2010. Article (CrossRef Link).

[5] M. In, *et al.*, "Web of Things (WoT) Standardization Status," *TTA Journal*, Vol. 138, pp. 79-84, 2011.

[6] S. You, *et al.*, "Smart Mobile Services: M2M Technology and Its Standardization Status," *Telecommunications Trends Analysis*, Vol. 26, No. 2, pp. 50-60, 2011.

[7] C. A. Ellis, *et al.*, "Investigations on Stochastic Information Control Nets," *Information Sciences*, Vol. 194, pp. 120-137, 2012. Article (CrossRef Link).

[8] W. Cho, *et al.*, "u-SERVICE DESIGN: Ubiquitous Computing Experts Series," Jinhan M&B Publishing Co., 2009.

[9] K. Kim, "A Model-driven Workflow Fragmentation Framework for Collaborative Workflow Architectures and Systems," *Journal of Network and Computer Applications*, Vol. 35, Issue 1, pp. 97-110, 2012. Article(CrossRefLink).

[10] K. Kim, "A Layered Workflow Knowledge Grid/P2P Architecture and its Models for Future Generation Workflow Systems," *Future Generation Computer Systems*, Vol. 23, No. 3, pp. 304-316, 2007. Article(CrossRefLink).

[11] H. Ahn, *et al.*, "A Process-driven IoT-Community Computing Model," *ICIC Express Letters*, Vol. 8, No. 2, pp. 483-488, 2014.

[12] N. Clarles Russell B.Sc. MInfTech, "Foundations of Process-Aware Information Systems," Ph.D. Dissertation, Queensland University of Technology, 2007.

[13] H. Ahn, and K. Kim, "An Advanced IoT Conceptual Platform for Ubiquitous Community Computing Systems," in *Proc. of the 8th Asia Pacific International Conference on Information Science and Technology*, pp. 275-278, 2013.

[14] "Internet of Things in 2020: A Roadmap for the Future," INFSO D.4 Networked Enterprise & RFID, INFSO G.2 Micro & Nanosystems in co-operation with the RFID Working Group of the European Technology Platform on Smart Systems Integration (EPOSS), 2008.

[15] "RFID and the Inclusive Model for the Internet of Things," Final Report, CASAGRAS: An European Union Framework 7 Project (EU Project Number 216803), 2010.

[16] B. Jamoussi, "IoT Prospects of Worldwide Development and Current Global Circumstances," Presentation Slides in the 2010 China International IOT Exposition, 2010.

[17] Luigi Atzori, Antonio Iera, and Giacomo Morabito, "The Internet of Things: A Survey," *Computer Networks*, Vol. 54, No. 15, pp. 2787-2805, 2010. Article(CrossRefLink).
[18] Sonja Meyer, Andreas Ruppen, and Carsten Magerkurth, "Internet of Things-Aware Process Modeling: Integrating IoT Devices as Business Process Resources," *Advanced Information Systems Engineering*, Springer Berlin Heidelberg, 2013. Article(CrossRefLink).
[19] Benedikt Schmidt, and Markus Schief, "Towards Agile Business Processes Based on the Internet of Things," *Advanced Manufacturing and Sustainable Logistics*, Springer Berlin Heidelberg, pp. 257-262, 2010. Article(CrossRefLink).
[20] Miao Wu, *et al*., "Research on the Architecture of Internet of Things," IEEE *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering,* Vol. 5, 2010. Article(CrossRefLink).
[21] Zhuming Bi, Li Da Xu, and Chengen Wang, "Internet of Things for Enterprise Systems of Modern Manufacturing," *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 2, pp. 1537-1546, 2014. Article(CrossRefLink).
[22] Da Xu, Wu He Li, and Shancang Li, "Internet of Things in Industries: a Survey," *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 4, pp. 2233-2243, 2014. Article(CrossRefLink).
[23] Hsiao-Hsien Chiu, and Ming-Shi Wang, "A Study of IoT-Aware Business Process Modeling," *International Journal of Modeling and Optimization*, Vol. 3, No. 3, pp. 238, 2013. Article(CrossRefLink).
[24] Martin Bauer, *et al*., "IoT Reference Model: Enabling Things to Talk," *Springer Berlin Heidelberg*, pp. 113-162, 2013. Article(CrossRefLink).
[25] M. Kim, *et al*., "A Conceptual Extension of the Internet of Things Architecture and Its Standardization Reference Model," in *Proc. of the 7th International Conference on Internet* (ICONI), pp. 169 ~ 172, Kuala Lumpur, Malaysia, 2015.
[26] "Next Generation Networks – Frameworks and functional architecture models: Overview of the Internet of Things," ITU-T Y.2060, *Series Y: Global Information Infrastructure, Internet Protocol Aspects and Next-Generation Networks*, 2016.

Meesun Kim is a full-time M.S. candidate of computer science department and a graduate member of the collaboration technology research laboratory at Kyonggi University, South Korea. She received her B.S. degree in computer science from Kyonggi University in 2015. Her research interests include workflow systems, BPM, CSCW, workflow supported social networks discovery and analysis, and process-aware IoT.

Hyun Ahn is a full-time Ph. D. student of computer science department and a graduate member of the collaboration technology research laboratory at Kyonggi University, South Korea. He received B.S. and M.S. degrees in computer science from Kyonggi University in 2011 and 2013, respectively. His research interests include workflow systems, BPM, scientific workflow systems, workflow-supported social and affiliation networks discovery, analysis, and visualization.

Kwanghoon Pio Kim is a full professor of computer science department and the founder and supervisor of the collaboration technology research laboratory at Kyonggi University, South Korea.

He received B.S. degree in computer science from Kyonggi University in 1984. And he received M.S. degree in computer science from Chungang University in 1986. He also received his M.S. and Ph.D. degrees from the computer science department at University of Colorado Boulder, in 1994 and 1998, respectively. He had worked as researcher and developer at Aztek Engineering, American Educational Products Inc., and IBM in USA, as well as at Electronics and Telecommunications Research Institute (ETRI) in South Korea. In present, he is a vice-chair of the BPM Korea Forum. He has been in charge of a country-chair (Korea) and ERC vice-chair of the Workflow Management Coalition. He has also been on the editorial board of the journal of KSII, and the committee member of the several conferences and workshops. His research interests include groupware, workflow systems, BPM, CSCW, collaboration theory, Grid/P2P distributed systems, process warehousing and mining, workflow-supported social networks discovery and analysis, process aware information systems, data intensive workflows, and process-aware IoT.