
Adaptive Context-Aware Design Using Context State Information for the Internet of Things Paradigm

Derrick Ntalasha^{1,*}, Renfa Li², Yongheng Wang²

¹*School of Information and Communications Technology, Copperbelt University, Jambo Drive, Riverside, Kitwe, Zambia*

²*College of Information Science and Engineering, Hunan University, Lushan Road Changsha, 410082, China*

E-mail: dbntalasha@gmail.com

**Corresponding Author*

Received 13 March 2020; Accepted 05 May 2020;

Publication 19 June 2020

Abstract

In the Internet of Things (IoT) paradigm, context state information plays a critical role in advancing the development of adaptive pervasive applications. Pervasive services and context-aware computing are emerging as the next computing paradigms in which infrastructure and services are seamlessly available anywhere, anytime, and in any format. The IoT paradigm raises new opportunities and demands on the underlying systems, in particular, the need to have systems that are adaptive and context-aware using context state information. In this paper, we introduce a new adaptive context state design technique to model context-aware applications that are sensitive to context state information changes. Each context change event is captured, interpreted and reacted to so that applications and users use only the functionality and adaptability needs that are solutions to their needs. The solution is modeled using Finite State Machine (FSM) and semantic localization so that context state information within the IoT paradigm is aligned to events. The semantic localization process precisely estimates the proximity location of the user along with the quality of context (QoC) attributes using the Bluetooth cell-based approach. This semantic information is useful in determining and

Journal of Mobile Multimedia, Vol. 15_4, 289–320.

doi: 10.13052/jmm1550-4646.1542

© 2020 River Publishers

inferring the user activities in a location. The QoC attributes are used to determine the confidence of the user location and range of the Bluetooth beacons within the IoT domain. This will, in turn, be used to determine whether the user is in the location or not. The alignment technique in our model represents the proper and new solution concerning functionality and adaptability needs expressed by other user applications in the IoT environment. The experimental scenario results indicate that a user can continue to enjoy their daily activities while the IoT application adapts continuously to their changing needs and notifying service providers of the changes according to the events of the user.

Keywords: Context adaptation, context event alignment, context state information, finite state machine, semantic localization.

1 Introduction

Context-aware adaptation [1, 2] in the Internet of things (IoT) refers to systems being able to adjust their functionality and structures to the ever-changing environment without the users' intervention. According to our preliminary findings, most researchers develop context-aware platforms that include an adaptive functionality that reacts to context information changes. Adaptive context-aware systems are difficult to create as they must be able to understand context state information and dynamically change their behavior at runtime according to the context change events. Context information can include the user location, preferences, activities, environmental conditions, and availability of computing and communication resources.

The adaptability of the context-aware systems [3] can generate inconsistencies as well as integrity problems, and the combinatorial explosion of possible variants of these systems with a high degree of variability can introduce great complexity. The pointed out issues bring about technical challenges and increased cost of building and implementing adaptable context-aware systems. This has led to developers to seek ways of improving these systems by defining improved and advanced methods of specifying and modeling context-aware systems to ensure improved quality and reduced developmental time, efforts and costs.

The other reason for adaptable context-aware systems within the IoT domain is fast emerging is because of the heterogeneity and dynamic nature of the domain. The inherent and ubiquitous nature of the IoT environment has led to applications to deal with more information that changes frequently.

For this information to be easily used by an application, its definition and management are key to its adaptability.

Context information [4] and its life cycle management [5] has the ingredient for creating a whole new range of smart entertainment and business applications that are more supportive to the user in a more consistent way so that the user is not affected by the frequency at which this information changes.

When the change of context information occurs within the application operating environment, an adaptive context-aware application modifies its behavior to suit the requirements of both the user and the application. The trigger to the adaptation is usually the detection of a specific context resource under consideration.

The approach followed by many researchers to deal with contextual information in adaptive context-awareness without looking at the states of context raises questions about the possible events that can trigger actions that can change the state of context that a system can adapt to. In particular, it is not clear how a system should behave when the user and application context states change. Furthermore, it is also not clear how a system would adapt when detected user actions and locations based on the prior characterization of the space and sensed data are captured by Bluetooth enabled devices.

The Bluetooth Cell, beacons, [6–8] based approach of determining the proximity range for the location was adopted. One of the simplest ways to achieve meter-level accuracy in specific zones is by a proximity measurement. The proximity derives the location of a target object concerning its vicinity to the location of the beacon. The target object receives the Received Signal Strength Indicator (RSSI) [9] to measure the power present in a received radio signal from a given beacon, so the location of the beacon or the symbolic cell identification defines the location of a target. It is therefore easy to detect proximity to a Bluetooth Low Energy device [10, 11], and hence confidently trigger a location-based event [12]. Each beacon determines the location of the user using only the visibility of that beacon within the room without taking into account the distance or angle of measurement. The localization of the user's semantic location [13] is based on the knowledge of the limited range for each beacon. The single mobile user in our application is detected by the Bluetooth beacons with a Bluetooth enabled mobile phone. Every Bluetooth unit has its own unique identity, and it is easy to register it in the system. Bluetooth beacons were used to broadcast signals to the user thereby detecting his/her position in the house. The adaptive context-aware application was able to deliver contextually relevant content and information

to users depending on the specific location of the user. This unique user experience through location aware applications has never been easier.

To this effect, we propose a new adaptive context-aware design using context for the IoT paradigm that uses context state [14] information to align context information to an event using the finite state machine (FSM) [15] and semantic localization. The application uses a well-defined user model that captures user location semantics [16] and maps it to context state information. User semantic localization provided more meaning and extension to the location instead of the physical location. This is so by estimating precisely the geographic location information of the user along with the user's quality of context (QoC) attributes, such as Up-to-Dateness, Completeness, and Certainty. Besides, the confidence for the quality of the received signals from the beacon is calculated by taking into consideration all aspects that affect the range estimates of context information obtained from each beacon. The user position does not only include the physical coordinates but a detailed description of the geographic location. Since this semantic information has more attributes, it provides a systematic mapping of the user's geographic location and activities to the context state information, needed by FSM within the proposed system. The FSM manages the change in state context events and state transitions. The effect is those allowable events are constrained according to the current application state. The result of a typical IoT scenario simulation of the adaptive context-aware application changes action event, current state, the scenario that continuously adapts to the user needs according to the adaptation rules governing the context state transitioning for the context-aware application.

The organization of this paper is as follows; in Section 2, we present the pertinent state of artworks related to this study. In Section 3, we present the IoT Scenario and the context state transition diagram. In Section 4 we present the adaptive context-aware application design, quality of context, adaptation rules and the execution flow for the FSM. In Section 5, we present experimented results and in Section 6, we make our conclusions and future works.

2 Related Work

Over the past decade, context-aware adaptation research works have been done and many provide valuable insights on context state understanding, context events and indeed the realization of the Internet of Things (IoT) domain-specific applications. This section discusses the state of art-related works that highlight the important contributions to context adaptation.

Authors of [5] proposed a context cycle management scheme that used a lifetime of contexts to perform a multistage reasoning process based on Web Ontology Language (OWL). The model developed a scheme using the refined context event classification. Using this scheme, the context was classified into states; Ready, Running, Suspended, Resumed, Expired and Terminated. The scheme identified the specific cycle of each context in question as it pointed to the context of the user's activity. In their context management system, an event was divided into two parts; instant event and interval event. Instant events are events that are instantaneous while an interval event represents the exact interval of occurrence of an event, it starts at the moment the event commences and finishes when the event concludes. In other words, both the event and the interval, start at the same instant of time and extend for the same duration. While the research looked at temporal event classification into the context management in ubiquitous computing environments, our study uses the FSM and semantic localization using defined adaptation rules in demonstrating how user events and intentions can be modeled in IoT.

Context-aware adaptive applications that are intensely context-aware and continually adaptive to changes in context were addressed by [15]. The authors defined a platform that received continuous streams of context values that required that the application dealing with such would affect the changes using its internal configuration. The definition of an adaptive finite state machine (A-FSM) addressed the configurations by defining and applying adaptation logic and asynchronous updating of context information, with the latter leading to inconsistencies between the external physical context and its internal representation within an application. To better the adaptation process, the researchers identified adaptation fault patterns which would describe algorithms that could detect faults automatically through the A-FSM. To evaluate this model, a context-aware adaptive application was developed that used adaptation rules which automated the significant part of the application. The development and execution used two aspects; an event-driven context manager and an adaptation manager. The event-driven context manager was used to collect and manage context information and the adaptation manager maintained, evaluated and applied a set of rules to ensure that only adaptive actions could be executed. This model applied context information to achieve adaptation in the context-aware application without taking into account the state of context which is vital in achieving adaptation.

In [17] a mobile device content adaptation model was defined, a context-aware model that looked at the context of the user and the environment of the mobile device. This device-independent model was used for content

adaptation depending on the semantic capabilities of the device requiring the next content. The model had components to manage contexts, query functions and storage facilities for devices. The querying of devices by the system was done through the XQuery language [17] for device profiles and the results of the query would be a Simple Object Access Protocol (SOAP) message [18]. This application only queried and exchanged context fragments with a direct relation with content adaptation, unlike our model that uses FSM to communicate context state information to the applications that need adaptation.

A context-aware adaptive application that modeled a system behavior using information derived from the context information was discussed in [19]. The goal-oriented approach used nonfunctional goals to analyze and select possible adaptation alternatives, in consideration of the particular contextual event. The context was a composition of many aspects like where you are, who you are with and what resources. The safety application called HOPE (Helping Old People Easily), illustrated how the application of context information, rules, and adaptation could deal with this safety-critical application. This mobile adaptive context-aware application [20] gave the gist of how modeling a solution in an environment can also take a rational approach in which the adaptive measure is best suited for the current event. The consideration of context state was concerning the situation of the source of context information. Context information has states that are similar to the process states in an operating system. These context states align context to an event. Our research exploits the context states to design an adaptive context aware application taking into account the user location and activity.

In [21] a situation-aware application that applies to mobile computing environments was discussed. The authors used the extended FSM [22] to model a situation-aware scenario [23]. They used the state of context to represent a condition that a system can present. The application used contextual information from historical and internal action records. The dynamic system changes as a result of context state transitions were modeled using FSM. The developed application was able to respond to the situation, adjusting the behavior as the situation changed. The application was used for validation of situation-aware requirements and generating situational changes in systems as the mobile environment dictated. The state transition was used as the context input to the model unlike our model that uses event action to depict a state change in context that is used for adaptation.

In [24] authors defined an android framework to develop context-aware applications by decoupling the functionality, mapping, and monitoring of

context to achieve adaptation. The decoupling, mapping, and monitoring were all done at runtime. Context information was obtained through various sensors of the phone. The sensor collected context parameters such as energy, network connectivity, location, and user activity. The framework used the sensor parameters and application configuration at runtime to determine the current context of the phone for adaptation to be achieved. Adaptive context awareness is better achieved when the state of context is taken into account.

Authors of [25] introduced a new concept based on context aware authentication and authorization that adaptively provided access rights and privileges to authentic users using dynamic context information. The main purpose of the paper was to dynamically and efficiently assign access rights and privileges to users depending on a current situation. The aim was to ensure that smart things are context aware and can be controlled remotely securely. This scheme called Context Aware Attribute Based Access Control (CAABAC) integrated contextual information and its attributes into the access policy. Furthermore, authors evaluated the worthiness of the Ciphertext-Policy Attribute-based Encryption (CP-ABE)[26] scheme in granting fine grained authorization. A contextual token was used to define contextual information parameters.

The token corresponded to the secret key that was defined from the contextual information in the access policy under a predefined context by a context manager. The context manager is deployed to verify the validity of data consumer's requests and generate access tokens for them. To decrypt a ciphertext, the user needs not only to possess the proper attribute set but also to have a valid access token. The contextual tokens can be set arbitrarily in the access policy along with attributes, and one ciphertext can be linked to many contextual tokens as the number of the considered contextual parameters (location, time, emergency or normal situation, data type. Our research exploits the context states and its attributes to design an adaptive context aware application without taking into account access rights and privileges for the specific smart thing context attributes.

3 IoT Scenario

Jerry has returned home after a long day at work. Upon reaching home, he usually makes a cup of tea, goes to the living room to drink his tea while watching his TV favorite sports channel. So he goes to the kitchen to make his tea. He goes to the living room after making his tea to watch the sports channel program which the application detects and the TV set is switched

on as he entered the room. The IoT application initializes the Watching TV action to the ready state and to running as soon as he sits down. While watching TV and drinking his tea, the living room temperature gets higher than he wants it, he shifts to the bedroom where he continues to watch his sports channel using the TV set in the bedroom and where there is an air conditioner. As he is leaving the living room, the watching TV state is transitioned by the application from running state to suspended state and notifies the state transition event to corresponding service providers for the TV in the bedroom to READY. As he enters the bedroom, the IoT application notifies the state of the TV in the living room from which the context state of the TV in the bedroom adapts to transition to the suspended 'Watching TV' context state to resumed state. Jerry continues to watch his game with the application transitioning from resumed to running state. The soccer game comes to an end and the application transitions from running to expired state. If Jerry decided to stop watching TV after the soccer game the IoT application transitions from running to the terminated state. This event eventually makes Jerry stop watching TV with the IoT application detecting this event by turning off the TV. In all this, Jerry continues to enjoy his routine with the IoT application continuously adapting to his changing needs and notifying service providers of the changes.

3.1 Semantic Localization

Localization of the user context in a particular room or hallway in our study was achieved through semantic localization. We defined a new way of dealing with the localization of the user by taking into account information that affects localization by adding quality of context information. Semantic localization is defined as the determination of positional relations on a defined object, for the inclusion in the principled integration of heterogeneous localization context information such as those derived from ubiquitous sensors in the Internet of Things [27]. As such, the problem of the mapping of geographical location onto semantic descriptions, location-activity relationships was solved. This context information described the location of the user at any particular instance and is defined as follows;

Location = (User, LocationName, UserActivity, QoC_Confidence, Time)

User defines the object that is being located within the defined environment, LocationName is the name of the location being considered i.e. Room, and User Activity is the event associated with the user in the location. Each

location where the user is, in the environment, has a defined activity associated with that specific location, QoC Confidence is the location estimation of the measuring device of the user along with the quality of the context attributes and Time, is the Timestamp.

The location context is modeled within the adaptive context-aware application to capture the user's location and activity so that appropriate adaptive service is provided to the user. The adaptive application uses the location context information states to dynamically provide services to the user by considering the user's current location and events occurring in the house. Context information states are aligned by the finite state machine to events of the user in the location. The context information in the IoT paradigm has six states namely: ready, running, suspended, resumed, expired, and terminated. These states are also similar to the process states in an operating system. These context states align context to an event. The initial state of every context is assumed to be in a ready state. This state indicates that context is ready to be executed upon being validated with the transition from the ready state to a running state. An invalidated context that is assumed to be still blocked but ready, remains in a suspended state. When a context that had remained in a suspended state is validated, it is transitioned to a resumed state. If the timeout event occurs, the context that was in a suspended state is transitioned to an expired state. A context that is not valid anymore is transitioned to a terminated state.

3.2 Semantic State Flow Diagram

Using the semantic location information, a state flow diagram was constructed that defined the state transition protocol. The diagram models each of the location semantic attributes as a process in achieving the state change. The change of the state for the adaptive context-aware application takes into account all the attribute instance values. Every process state transitions into another state when there is a change in the attribute state. Using the state process flow, the change in the state occurs if and only if the user's location, activity, QoC Confidence and time change as shown in Figure 1.

The input data (User, Location, User Activity, QoC_Confidence, Time) in the context flow diagram are the semantics attributes of the location context while the state represents the initial state of the application. When data is captured by the Bluetooth beacons using received signal strength indicator (RSSI) as a measurement of the power present in a received radio signal[28], instance values for User, location, User activity, QoC Confidence and Time,

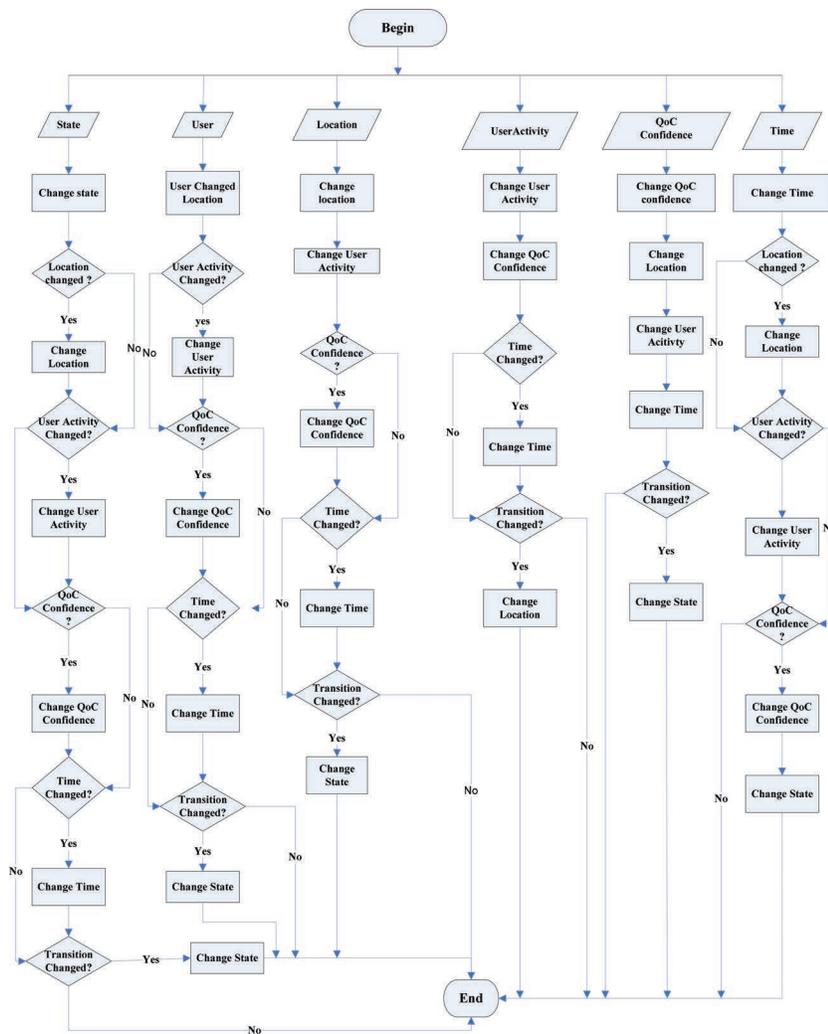


Figure 1 Context state flow diagram.

are processed by their respective task handlers. Each of them is evaluated to determine whether its value will affect the overall state of the system. If there was a change in all attributes to their individual tasks, the state transitions to another state which the adaptive application uses to provide appropriate services to the user at an appropriate place at an appropriate time, considering the user’s activity.

The scenario in Section 3 above is a typical IoT application that we modeled with a predefined number of application states which change or transition depending on actions by Jerry as he does his daily routines. His actions trigger events in our IoT application which lead to the application going through multiple states changes during a runtime session. The state transitions are initiated by runtime events. The adaptation property that we are modeling is achieved through actions when an event is received and the resulting change of state. Each event causes a transition to a new state that invokes a change in the immediate behavior of the whole application.

The IoT application uses context objects to receive context change events that are allowed according to the current state of the application. When a context object detects an event, the application changes its transition state. The application uses the new state to check if a context state change can be adapted according to the collection of rules that define the application events and states that are allowable. These rules define which states are reachable from any other state. In this way, only services that correspond to the state of the application are adapted after the state change notifications are sent to the listening context objects of the application.

4 Adaptive Context-aware Application

In Figure 2, the adaptive context-aware application is illustrated a use case diagram, detailing all the structures involved in achieving the intended services to the user. The application input is the location context as defined in section 3.1. Instance values of the semantic location variables are key to the adaptive nature of the application. The state adaptation for the application is achieved through the FSM that manages state transitions when the condition is satisfied. The context-aware adaptable application communicates with the FSM application that provides the adaptation rules. To be specific, it provides a set of rules of event action change and transition that specify a possible adaptation. The FSM has full control over the set of rules that may change at any time depending on the event that can trigger a transition, regardless of the state of the running applications. The state of the running application only changes when listening objects receive notice.

In our architecture user activity was defined by the user Entry, InRoom, and Exit to the locations. Three locations were simulated, House, Living room and Bedroom. Each of the locations was assigned a Bluetooth low energy beacon with a specific range to measure the proximity of the user. Bluetooth Low Energy (BLE) signals from battery-driven beacons are at the core of our

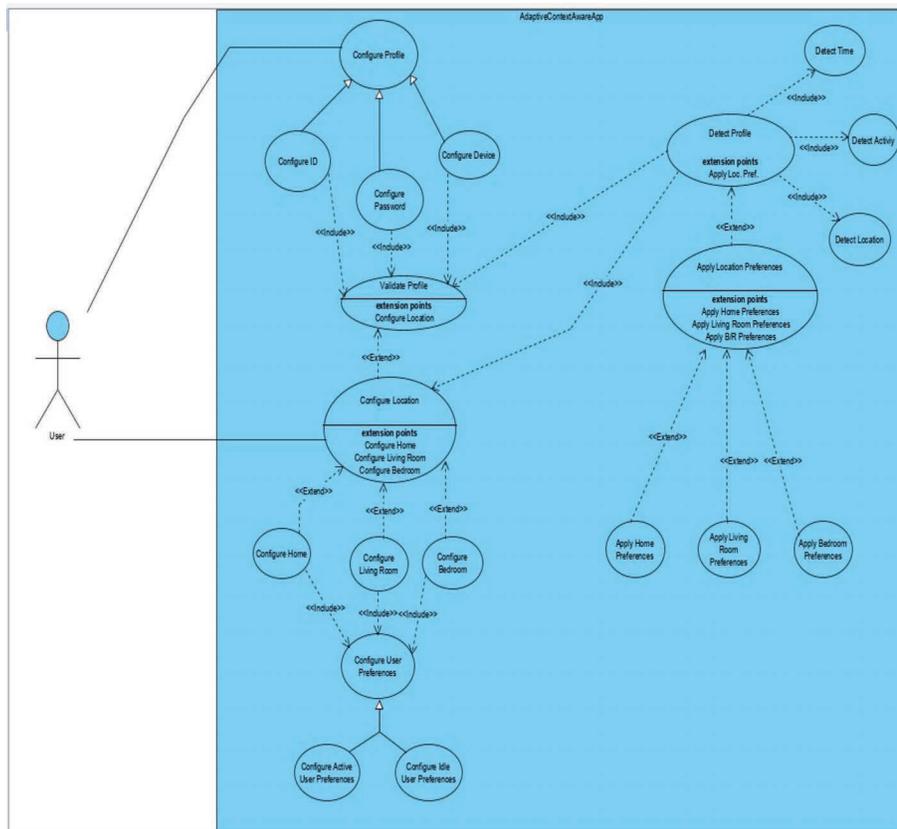


Figure 2 Adaptive context-aware architecture.

research. It's one of the latest technologies that has emerged and become an industry-standard available on most devices today. It uses BLE beacons that are inexpensive, small, has long battery life and does not require an external energy source. The user mobile phone detects the signal from the beacon and the application identifies the position of the beacon and hence estimates the location of the user.

The use of Beacons was setup that each location (House, Living Room, and Bedroom) had a beacon placed. The purpose of the beacon was to transmit a one-way signal within proximity to the user Bluetooth enabled device. When the user enters the beacon's zone, the application determines the user's proximity. To ensure the user's proximity is well guaranteed, the application calculates the user's proximity confidence by using QoC attributes. The three

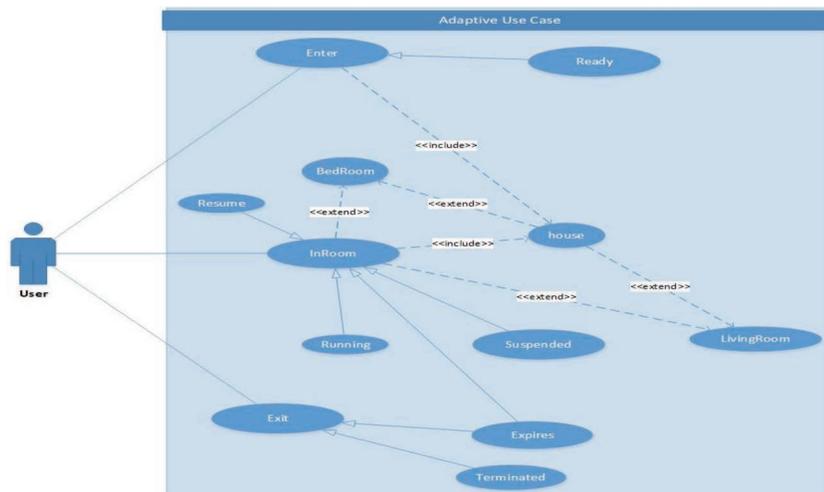


Figure 3 FSM use case transitions.

objective attributes are up-to-dateness, certainty, and completeness as will be explained in the next section. If the value of the sensed beacon value has high proximity confidence and the user activity is evaluated to be true, then the state adaptation is triggered. The FSM uses the adaptation rules to set the required state for the application.

Each rule is such that it includes a description of the activity to be adapted, an applicability condition specifying when the rule is applicable, the next state of the activity and the set of variables required by the activity.

As the application runs, each rule is matched against the user's activity through the FSM to find out whether adaptation should occur as shown in Figure 3. To define the rules for our system, the FSM is used to align context information to an event. In every state there is defined behavior which will only be executed when the object is in that state. Every FSM has an initial state, this means which state it starts in when it is created and has to be defined when constructed or instantiated. Every state either constantly evaluates if it should transition to another state or will transition to another state based on a triggered event. When a FSM is created, the following rules are defined and are unique to the computation;

- Each of its transitions is uniquely determined by its source state and input symbol,
- Reading an input symbol is required for each state transition.

In our application the following adaptation rule set was applied:

- A context-aware application defines rules to be used for adaptation;
- The application context states, variables for the next state for events listening objects must be evaluated for the adaptation process to occur. And this process only occurs when the condition is met.
- When all conditions are true, adaptation occurs. The state of the application as determined by the adaptation application is updated to the new state taking into account the adaptation rule.

4.1 Quality of Context

Bluetooth beacons were used to determine how far or near the user is from the beacon. Beacon range estimates have significant degrees of uncertainty due to many reasons such as reflections of the signal, obstructions, and orientations of the user mobile devices and the beacon itself. When a beacon sends a signal, the packet includes a power field that measures how strong a signal is at a known distance. This known distance (range) is a calibrated value to measure the signal strength for a specific room. The application determines the user location by comparing the signal level received and the set or calibrated value. Since the beacon sends both the measured and calibrated value, confidence can be calculated. Confidence calculation is obtained by taking into consideration all aspects that affect the range estimates of context information obtained from each beacon and is defined as “*the measure of confidence in the measured context information (signal) as provided by the context object (Beacon).*”

To standardize the collection and measurement of context information, every argument that affects context information was taken into account and used in calculating the confidence of the context. The measured and collected context was classified into two aspects; context weighting and QoC value.

Confidence of context calculation used up-to-dateness, certainty, and completeness QoC parameters for each signal received from a beacon. The QoC parameters used are specific to the application needs. The weighting of the contexts adds an important dimension to the quality parameter by attaching a weight to the signal about the expected value. This gives a true state of the quality of the context object in question.

4.1.1 RSSI Weighting

The evaluation of the weight for the proximity information using the received RSSI values takes into account the closeness of the measured RSSI value is to the calibrated value. Table 2 has three (3) values representing the expected

RSSI values at each distance. This rating is between 0 and 1. As beacon signals are obtained, they are rated proportionally to the variances for the expected and expected variable values using equation 4.1;

$$\omega_i = \frac{1}{\delta_i^\gamma} [29] \quad (1)$$

ω is the weight, δ is the mean and γ the variance. Small values for the weight is given to the RSSI values with less precise measurement compared to the calibrated and more weight to measurements that are closer to the calibrated. When the received signal strength is closer to the calibrated, its weight is closer to 1. When a zero (0) rating is obtained then the value is invalid and cannot be used to calculate beacon value confidence.

4.1.2 Quality of RSSI Value

This research takes into consideration three (3) objective parameters of the RSSI values to measure whether the RSSI values are closer to the calibrated values as measured by the beacon. Let the received signal values received from a beacon be the context object *CO*. *Up-to-Dateness*, *Completeness* and *Certainty* parameters were used to calculate the Quality of RSSI value.

The Up-To-Datedness parameter is used to determine the validity of RSSI information as given by the beacon at a given time.

$$Age(CO) = t_{curr} - t_{meas}(CO) [29] \quad (2)$$

Age (CO) is how long the RSSI value remains valid and it denoted by lifetime, t_{curr} , is the current time and t_{meas} the measurement time of RSSI. Therefore, uptodateness is given as

$$U(CO) = \begin{cases} 1 - \frac{Age(CO)}{Lifetime(CO)} & [30] \\ \text{if } Age(CO) < lifetime(CO) & \text{otherwise}(CO) \end{cases} \quad (3)$$

The completeness attribute of the RSSI value indicates much information provided by the beacon question. Table 1 provides basic information for the beacon. Therefore, completeness is the ratio of the number of attributes available to the total number of attributes and is given as;

$$C(CO) = \frac{\sum_{j=0}^m \omega_j(CO)}{\sum_{i=0}^n \omega_i(CO)} [29] \quad (4)$$

m is the number of the parameters of beacon that have been assigned a value and $w_j (CO)$ represents the weight of the jth parameter of the beacon that

has been assigned a value. Similarly, n is the total number of the parameters of the beacon and w_i (CO) represents the weight of the i th parameter of the beacon.

The certainty parameter measures the reliability of a beacon of providing the RSSI values and is determined by the broadcast advertised message and received message and is given by;

$$Ce(CO) = \begin{cases} C(CO) * \frac{N_j + 1}{N_i + 1} & \text{if } F(CO) \neq 0 \text{ and } CO \neq 0 \text{ then} \\ \frac{N_j}{N_i + 1} & [29] \end{cases} \quad (5)$$

N_j+1 is the number of the broadcast advertised messages, N_i+1 is the number of received messages, $C(CO)$ is the Completeness and $F(CO)$ is the freshness or Age (CO). This measures the advertising which is the time between the beacon sending an RSSI value and when the application receives the message.

4.1.3 Confidence for Quality of RSSI

RSSI confidence for the above QoC parameters is derived from the three objective QoC parameters and the weight. It measures the confidence in the QoC parameters provided in the beacon specification. RSSI confidence is determined by applying the RSSI weight and the actual RSSI values measured by the beacon. RSSI confidence is given by;

$$Cf(CO)_{conf} = \sum_{i=0}^n (Cf_{act}) * C(CO)_{QoC} \quad (6)$$

$Cf(CO)_{conf}$ is the cumulative confidence of the RSSI values using quality parameters and weight, $C(CO)_{QoC}$ is the calculated quality value for the QoC parameter of the measured RSSI values as given in the Section 4.1.2. Cf_{act} is the weight for the RSSI.

4.2 Execution Flow Chart for the Adaptive Context-aware Application Using FSM

The flowchart executes as follows;

1. On starting up, the application calls the event provider to initialize the fsm, context adaptation rule set.

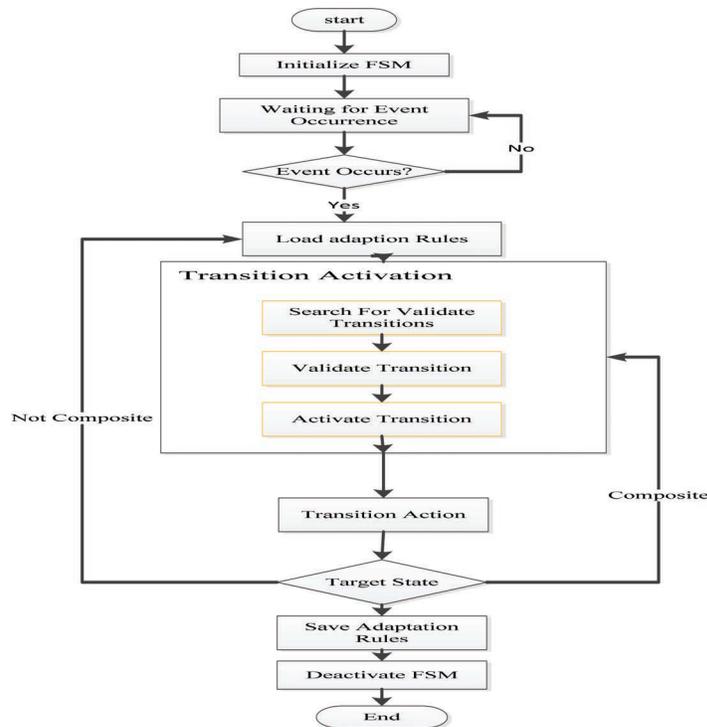


Figure 4 FSM FlowChart for the adaptive context-aware system.

2. On detecting an event, the application calls the fsm event manager to do the following:
 - Loads current fsm adaptation rules using the associated context adaptation rule set. If the loaded ruleset is null, then the initial state of the application is in active State;
 - Gets the next Transitions from active State that has Events equals to detected Event; looks through all selected Transitions and calculates Guard Boolean expressions; stops when calculated Boolean expression is true; Transition that owns such Boolean expression is triggered;
 - Executes output Actions on Transition that is being triggered, executes on-enter Actions in target State;
 - If target State is composite, continues processing from step 2.2;
 - If the target State is not composite, executes includes fsm starting from step 2.1.

- *Saves new fsm adaptation rules using associated context adaptation rule set;*
3. *If root StateMachine active State has type State Type of end, calls the event provider; disposes all event providers associated with root fsm and stops accepting Events.*

5 Experiment Setup and Results

To validate our proposed technique, we developed an application using the IoT scenario of Section 3 with the Bluetooth cell-based approach. It is important to note that our experiment is based on a single mobile user so that context state transitions are fully recorded. The AltBeacon specification [31, 32] was adopted to define the format of the message that the Bluetooth Low Energy proximity beacons broadcast. These beacons do not communicate with each other. Reason being, beacons are servers which provide certain information. Mobile phones connect to them as a client and fetch information. Since the beacons are servers they cannot connect to each other. To connect to each other either of them should be client. Which is not there in case of beacons in this experiment setup. A python program was used to read beacon advertisements using a Bluetooth Dongle on Android on the IntelliJ IDEA platform a commercial Java integrated development environment IDE [33]. IntelliJ IDEA provided a dedicated Android Support plugin. This plugin is bundled with the IDE and is enabled by default. The experiments were done by deploying on to an Android phone in a house. The proximity of the mobile user for each room tested according to Table 4. The AltBeacon specification provided for accurate proximity for messages exchanged between the beacon and the mobile user's device (Android Phone). This approach used two phases to achieve the location state. The beacons used were placed in each section of the house as shown in Figure 5.3 and no beacon covered more than one section. In the first phase of the experiment, all beacons were assigned a room with an associated range. Each beacon within the rooms observed by the mobile user device contained the identifying information (Beacon Name, UUID, major, minor) along with proximity, TXpower, and an RSSI as shown in Table 1.

This experiment has benefitted from BLE characteristics, in particular its low power consumption, which allows it to be embedded in small devices with low-charge and small batteries, which can also last a few years. For all these reasons, BLE is a good candidate to be a revolutionary technology in the

Table 1 Broadcast message for beacon

Variable	Description	Purpose
<i>Name</i>	Beacon Name	Identifies the device by name
<i>UUID</i>	A beacon ID number	Identifies the Beacon
<i>Major</i>	A label	Identifies the group it belongs to
<i>Minor</i>	Another label	Identifies the subgroup it belongs to
<i>TX PWR</i>	Transmission Power	Allows calculation of proximity when compared with power received (RSSI)

Table 2 BLE performance metrics

No	Performance Attribute	Value
1	Throughput	100–230 kbps
2	Maximum number of connectable sensors	7
3	Power consumption	Less than 15 mA
4	Latency	3 ms
5	Maximum reachable range	77 m

Table 3 Comparison table

Protocol Attribute	BLE	Zigbee	Symphony
TX Antenna Height (m)	6	6	6
TX Power (dBm)	4	18	18
TX Antenna Gain (dB)	0	0	0
Frequency (MHz)	2400	2400	915
RX Antenna Height (m)	1	1	1
RX Antenna Gain (dB)	−6	−6	−6
Structure Loss (dB)	11	11	11
Sensitivity (dBm)	−70	−102	−140
Margin (dB)	20	20	20
Range (m)	77	291	2594

present market of wireless communications. Additionally, we have included Additional performance metrics in Table 2 with the aim to identify what are the current limits of BLE technology and in Table 3 performance metrics being compared with existing routing protocols.

The RSSI is signal strength (in decibels) received from the beacon. For experimental purposes, calibration constants were set as shown in Table 5. For each proximity setting, the purpose of the beacon is to estimate the

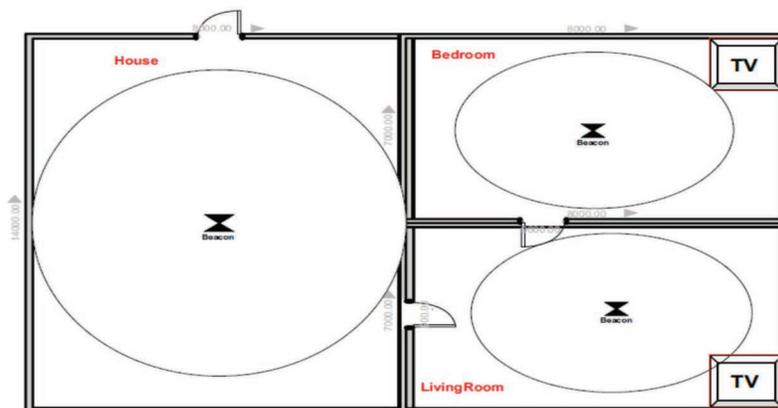


Figure 5 Beacon layout in the house.

proximity of the user using RSSI measurements. The calibrated value indicated what the RSSI should be if the user is in the location within a specific distance. Naturally, the RSSI signal is strong when the device is closer to the beacon and weaker when it is further away. Proximity provides a relative distance from the device to the beacon. This range was calibrated and stored for inference. The second phase involved the discovery of the beacons by the mobile user and the subsequent comparison of the discovered beacon value and the calibrated value.

The house and rooms as shown in Figure 5 were assigned a beacon with a proximity range to provide context about the user's location proximity. The application was setup to detect beacons that were nearby. Depending on the value of the proximity signal (RSSI) reported/detected, the state of the application would change the TV service. The location and the user activities determined by the cell-based application used as a triggers for events. User activities are in three (3) categories depending on the user location. Each beacon in a specified location was triggering Entry, InRoom, and Exit Events. This allowed us to determine when a user is near a location and how much time they spent. In Figure 6, the detected Bluetooth RSSI signal strength for the mobile user's presence is shown.

As shown in Figure 6, as the user device gets further away from the beacon, the RSSI signal values start to progressively reduce. According to the experiment, the RSSI values were stronger when the user device is closest to the beacon in the room. This indicated the presence of the user. In Figure 7, the user is detected to have moved away and within the scanning period, the

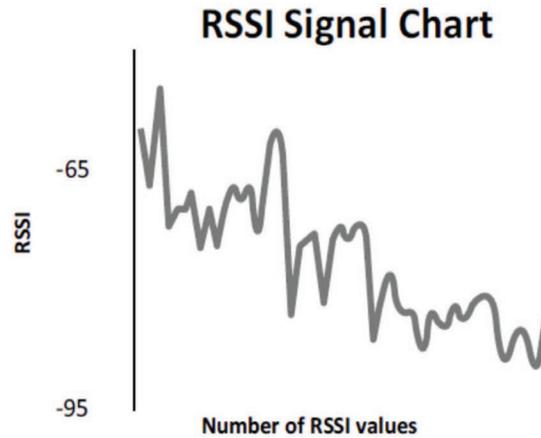


Figure 6 RSSI signal for User moving away.

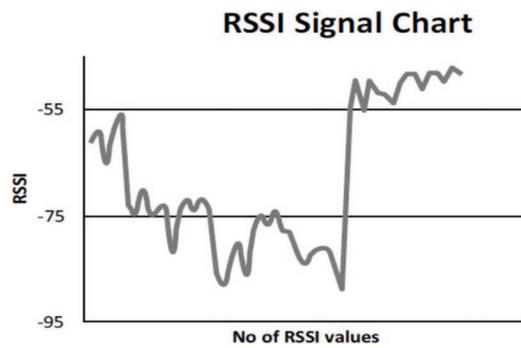


Figure 7 RSSI signal for User moving back to the room.

user is detected to have moved back in the room. The RSSI values are stronger and gradually reduce until the user starts to move closer to the beacon.

Beacon technology as used in our application was used to estimate how far the mobile user is from the beacon as well as determining the location of the user in the house. The proximity and the activity (Entry, InRoom, and Exit) triggered the change of state of the TV application. By its nature, the beacon technology estimate of the proximity has a degree of uncertainty. For experimental purposes, proximity was defined as shown in Table 4. All the considered proximities have a calibrated RSSI value and distance. Table 4 also shows the testing scenarios that were set up to indicate how the rooms of the house would have a specific beacon and its associated activity. For

Table 4 Proximity table

No	Proximity	RSSI	Distance
1	Immediate	60	1 m
2	Near	78	3 m
3	Far	85	7 m

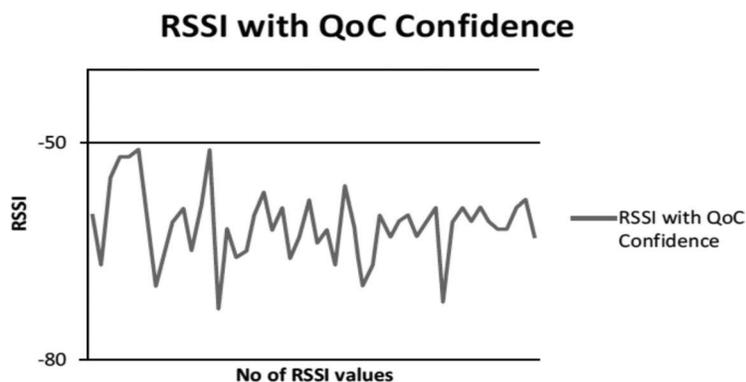


Figure 8 RSSI with QoC confidence.

experimental purposes, beacons placed in each room recorded an activity indicating with the mobile has entered that room, stayed in the room, or exited.

Using this information in Table 4, the QoC Confidence was used to reduce uncertainty in the proximity estimations. To calculate the QoC confidence of the location in each the user is detected, Uptodateness, completeness, and certainty were used. For the proximity considered in the experiment, the RSSI received from the beacon and calibrated values were taken into account. Figure 8 shows the RSSI values with QoC Confidence at immediate proximity according to the testing scenario. It can be seen that the RSSI values stable within the range.

Generally, each user activity has a timestamp for evaluating events. When a user appears in the location, the Entry Event is recorded and the application state is updated until a new InRoom event with its timestamp is received. This process continues until the application no longer receives beacon signals within a certain amount of time, it is at this point the Exit Event is triggered. In our simulation, when a user is in a particular location, three signal values are evaluated, Entry Signal, Exit Signal, and Exit Interval Signal. Entry signal values affect the Entry event, Exit Signal values affect Exit Events and Exit

Interval determines how long a beacon has not been receiving signals and affects the Exit Event as well. Here pseudo-code rules that the application uses to determine how user location and activities translate to adaptable events for the application and to effect the transition from one state to another.

```
WHILE QoC Confidence (Beacon signals) >= Beacon Range
  Do
    IF Location = Locations AND UserActivity >=
      Entry/Exit Signals THEN
      Entry/Exit Event
    ELSE
      Error
    IF Location = Locations AND User Activity >
      Exit Interval THEN
      Exit Event
    ELSE
      Entry Event
```

The adaptive context-aware application is in a house setup where the user moves from one room to another as he watches his favorite sports channel. The house has several television sets from which he can watch his game. For this implementation, only the TV sets from the living room and bedroom are used. The model uses events to refine the effected deduced contexts within the application. Whenever a user moves from one room to the other, the event is propagated to the FSM which then will affect an action depending on the change action event. The application proceeds as shown in the proceeding figures. Figure 9 shows the interface of the application and how it is initialized. As Jerry enters the house, the application detects his presence and the TV is ready for him to watch his football game.

As he enters the Living room the application moves from the ready state to running state and executes his favorite sports channel. The running state will run until another event occurs so that the state changes depending on Jerry's actions. As Figure 10 shows that, the TV on the left is playing as the user watches.

In Figure 11, Jerry decides to leave the living room for a much cooler room where an air conditioner is installed. The application detects this event of him leaving the room, pauses the TV set in the room while waiting for the beacon in the other room to record an event. The context information state transitions from running state to suspended, as the application waits for the next event.

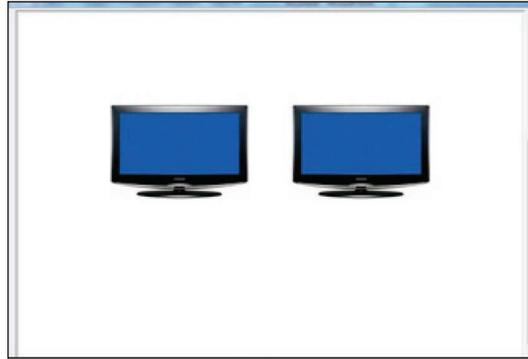


Figure 9 Application initialization.

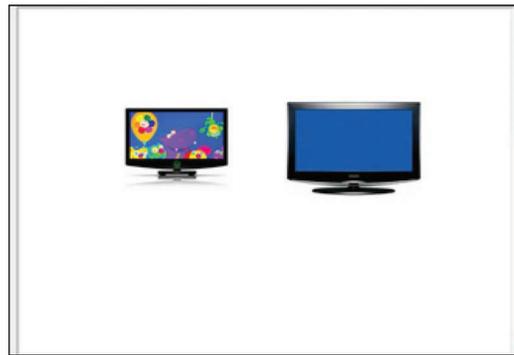


Figure 10 The application running Living Room TV.

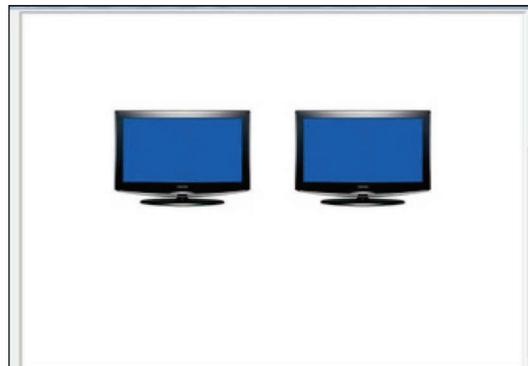


Figure 11 TV Application Paused.



Figure 12 Application Running the Bedroom TV.

In Figure 12, Jerry decided to watch his sports game from the bedroom and the TV set in the bedroom shows his game. As the figure shows, the right TV set is now showing the game. The TV set will continue to show the game until either the sports channel closes or he leaves the room. At that stage, if the channel closes the TV set will not show his game. The state will be expired. If he leaves the room or decides to switch off the TV set, the state will be terminated until he is ready to watch again.

6 Conclusion and Future Works

The adaptive context-aware application using FSM and location semantics for IoT has been presented. Our system used the cell-based approach to achieve proximity detection and location awareness by using location semantics. The location semantics provided more information for the application to use the location context information states to dynamically provide services to the user by considering the user's current location and events occurring in the house. The approach was not about providing the exact position but detecting the presence of the user in the location. Using the location semantics and the Bluetooth technology, the application was able to fully detect which room or location of the house the user was located.

Since the IoT environment supports multi-devices, users, applications, systems that are sensitive and respond differently to peoples' actions and behaviors, the functionality of this application delivers services and experiences according to the user's expectations. The use of context in IoT varies in many aspects but context states define a generic factor that context generally possesses. FSM exemplified the unique way of dealing with context states by managing the change state context events and state transitions. Taking into

account the user's needs, the adaptive context-aware application offered a good model the balances generality, concreteness and openness in offering users' exact needs.

The simulated scenario provides a solution through context state information alignment to typical events and behavior of the user, the model offered the solution to context-aware adaptation in a typical IoT environment as it continuously adapted to changing user need. The future will involve measuring how context-aware adaptation will evolve as IoT devices advance in functionality and services. Furthermore, this application can be enhanced to deal with context awareness where a mobile user's activity can be known. This would allow the application to determine the specific usage of a room in the house.

Acknowledgments

This paper and the research behind it would not have been possible without the exceptional support of my supervisors, Prof Li Renfa and Prof Wang. Their enthusiasm, knowledge and exacting attention to detail have been an inspiration and kept my work on track from my first encounter with the embedded and Networking lab.

Funding

This research work is done under the projects funded by the Study of Proactive Complex Event Processing for Large-scale Internet of Things a project of National Natural Science Foundation of China (61371116) and National High Technology Research and Development Program of China (Program 863) No. 2012AA01A301-01.

Conflict Of Interest

The authors certify that there is no conflict of interest in the materials discussed in this manuscript.

Availability Of Data and Materials

The authors confirm that the data supporting the findings of this study are available within the article.

References

- [1] V. Genaro Motti, “A computational framework for multi-dimensional context-aware adaptation,” in *Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems*, 2011, pp. 315–318.
- [2] J. E. May, “Systems and Methods for Context Aware Adaptation of Services and Resources in a Distributed Computing System,” ed: Google Patents, 2019.
- [3] A. K. Dey and G. D. Abowd, “CybreReminder: A context-aware system for supporting reminders,” in *Handheld and Ubiquitous Computing*, 2000, pp. 172–186.
- [4] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, “Towards a better understanding of context and context-awareness,” in *Handheld and ubiquitous computing*, 1999, pp. 304–307.
- [5] H. Chang, S. Shin, and C. Chung, “Context life cycle management scheme in ubiquitous computing environments,” in *Mobile Data Management, 2007 International Conference on*, 2007, pp. 315–319.
- [6] L. Pei, R. Chen, Y. Chen, H. Leppakoski, and A. Perttula, “Indoor/outdoor seamless positioning technologies integrated on smart phone,” in *Advances in Satellite and Space Communications, 2009. SPACOMM 2009. First International Conference on*, 2009, pp. 141–145.
- [7] Y. Wang, X. Yang, Y. Zhao, Y. Liu, and L. Cuthbert, “Bluetooth positioning using RSSI and triangulation methods,” in *Consumer Communications and Networking Conference (CCNC), 2013 IEEE*, 2013, pp. 837–842.
- [8] T. Nuradha, I. Gnanarathne, L. Perera, D. Denipitiyage, and D. Dias, “Beacon placement algorithm for hybrid indoor positioning with Wi-Fi and bluetooth low energy,” in *2019 Moratuwa Engineering Research Conference (MERCon)*, 2019, pp. 135–140.
- [9] M. Golestanian and C. Poellabauer, “Indoor localization using multi-range beaconing: poster,” in *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2016, pp. 397–398.
- [10] G. Vavoula, M.-A. Tseliou, and Z. Tsiviltidou, “Bluetooth Low Energy Beacon-based Positioning for Multimedia Guides in Heritage Buildings: a Case Study,” in *World Conference on Mobile and Contextual Learning*, 2019, pp. 102–109.

- [11] F. Martelli, “Bluetooth low energy,” *University of Bologna*, vol. 25, 2014.
- [12] T. Rattenbury, N. Good, and M. Naaman, “Towards automatic extraction of event and place semantics from flickr tags,” in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 103–110.
- [13] B. Lee, J. Oh, H. Yu, and J. Kim, “Protecting location privacy using location semantics,” in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 1289–1297.
- [14] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, “Context aware computing for the internet of things: A survey,” *Communications Surveys & Tutorials, IEEE*, vol. 16, pp. 414–454, 2014.
- [15] M. Sama, S. Elbaum, F. Raimondi, D. S. Rosenblum, and Z. Wang, “Context-aware adaptive applications: Fault patterns and their automated identification,” *Software Engineering, IEEE Transactions on*, vol. 36, pp. 644–661, 2010.
- [16] D. Zhao and X. Zhang, “Location Semantics in Positioning Services,” in *Proc. of 2 nd International Conference on Future Computer and Communication*, 2010.
- [17] T. Lemlouma and N. Layaida, “Context-aware adaptation for mobile devices,” in *Mobile Data Management, 2004. Proceedings. 2004 IEEE International Conference on*, 2004, pp. 106–111.
- [18] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana, “Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI,” *IEEE Internet computing*, vol. 6, p. 86, 2002.
- [19] H. Wang, R. Mehta, S. Supakkul, and L. Chung, “Rule-based context-aware adaptation using a goal-oriented ontology,” in *Proceedings of the 2011 international workshop on Situation activity & goal awareness*, 2011, pp. 67–76.
- [20] M. Raento, A. Oulasvirta, R. Petit, and H. Toivonen, “ContextPhone: A prototyping platform for context-aware mobile applications,” *IEEE Pervasive Computing*, vol. 4, pp. 51–59, 2005.
- [21] Y. Wang, “An FSM model for situation-aware mobile application software systems,” in *Proceedings of the 42nd annual Southeast regional conference*, 2004, pp. 52–57.
- [22] H. Rudin, “An informal overview of formal protocol specification,” *IEEE Communications Magazine*, vol. 23, pp. 46–52, 1985.

- [23] S. Wu, Q. Liu, P. Bai, L. Wang, and T. Tan, "SAPE: A System for Situation-Aware Public Security Evaluation," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
- [24] S. Elmalaki, L. Wanner, and M. Srivastava, "CAreDroid: Adaptation Framework for Android Context-Aware Applications," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 386–399.
- [25] A. Arfaoui, S. Cherkaoui, A. Kribeche, and S. M. Senouci, "Context-Aware Adaptive Remote Access For IoT Applications," *IEEE Internet of Things Journal*, 2019.
- [26] X. Yan, X. He, J. Yu, and Y. Tang, "White-Box Traceable Ciphertext-Policy Attribute-Based Encryption in Multi-Domain Environment," *IEEE Access*, vol. 7, pp. 128298–128312, 2019.
- [27] M. Weber and E. Lee, "A model for semantic localization," in *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*, 2015, pp. 350–351.
- [28] R. J. Ohira, T. Sullivan, A. J. Abotomey, and J. Jo, "A Comparative Study of Wi-Fi and Bluetooth for Signal Strength-Based Localisation," in *Robot Intelligence Technology and Applications 4*, ed: Springer, 2017, pp. 589–597.
- [29] A. Manzoor, H.-L. Truong, and S. Dustdar, "On the evaluation of quality of context," in *European Conference on Smart Sensing and Context*, 2008, pp. 140–153.
- [30] Z. Abid, S. Chabridon, and D. Conan, "A framework for quality of context management," in *International Workshop on Quality of Context*, 2009, pp. 120–131.
- [31] D. Helms, M. Wallace, D. Young, C. Sexton, and D. Martin, "Interleaving Multiple Bluetooth Low Energy Advertisements," ed: US Patent 20,160,105,788, 2016.
- [32] D. Deugo, "Using Beacons for Attendance Tracking," in *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS)*, 2016, p. 155.
- [33] D. Priefer, P. Kneisel, and G. Taentzer, "JooMDD: A model-driven development environment for web content management system extensions," in *Proceedings of the 38th International Conference on Software Engineering Companion*, 2016, pp. 633–636.

Biographies



Derrick Ntalasha was born in Zambia, in 1978. He received the BSc. Computer Science degree from the Copperbelt University (CBU) in Zambia, in 2002 and an advanced MSc. degree from the University of Manchester United Kingdom, in 2005, Information Systems Engineering. He has a Ph.D. degree in Computer Science and Technology from the College of Information Science and Engineering, Hunan University, China. His research interests are Context awareness computing and Internet of Things (IoT). He is currently the head of computer science Department in the School of Information and Communications Technology at the Copperbelt University.



Renfa Li is currently the President and Ph.D. supervisor of computer science within the School of Computer and Communication, Hunan University, with specialty in the fields of Computer architecture and computer application technology (CAT).

He has been a chief investigator on research grants totaling over RMB 1,000,000 from sources including Natural Science Foundation of China (NSFC) and others at state or provincial level. He has published over 60 scholarly papers in journals, book chapters and international conferences and acted as editor of a dozen books in recent years.

Professor Li is a councilor of China Computer Federation, a senior member of the Institute of Electronic and Electrical Engineers, a senior member of the Association for Computing Machinery.



Yongheng Wang was born in Hebei, China, in 1973. He received the Ph.D. degree in computer science from the National University of Defense Technology, Changsha, China, in 2006. Since December 2008, he has been working as a teacher at the College of Information Science and Engineering, Hunan University. His research interests include data mining, event processing and big data.

