Unified User-Centric Context: Who, Where, When, What, How and Why

Seiie Jang, Eun-Jung Ko, and Woontack Woo

Abstract—**To deploy context-aware applications, there has been a steadily increasing interest in context model to efficiently represent various contexts in daily life. However, most ways of modeling context are specific to purpose of each service or give undue value to particular information, e.g. location. To loosen the coupling between contexts and services, we propose unified context that represent user-centric contextual information in terms of 5W1H (Who, What, Where, When, How, and Why), to be shared among services. The proposed context can simply model a user's context in environments by assorting complicated information into six categories. Also, the unified context can enable sensor, user, and service to differently generate or exploit a defined 5W1H-semantic structure. Furthermore, unified context is structured with elements of 5W1H and attributes of each element so that any service can easily exploit the context for improving service. As a result, the proposed unified context enables context-aware services to more quickly provide personalized services by exploiting unified user-centric context.**

Index Terms—**Context model, unified context, user-centric context, and 5W1H**

I. Introduction

UbiComp-enabling technologies make our daily environments smarter to sense and to respond changes appropriately. In such environments, there has been a steadily increasing interest in context-aware applications which react to context of users or environments near them. To deploy context-aware applications, context model that simply represents complex contextual information plays an important role in creating, interpreting, and exploiting context.

A great deal of effort has gone into the understanding and modeling context over the past few years in the world of ubiquitous and pervasive computing. For example, Schilit and Theimer (1994) refer to context as location, identities of nearby people and objects and changes to those objects [1]. Schmidt, et al. (2000) define context as knowledge about state of the user and IT device, including surroundings situation, and, to a lesser extent, location [2]. Dey and Abowd (2000) define context as any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves [3]. However, most ways of modeling context are specific to each service or give undue value to particular information, e.g. location.

Additionally, various researches on context-aware computing have evaluated context modeling. For example, Held (2002) highlighted the requirement of context model for gathering, transferring, storing, and interpreting contextual information [4]. Strang, et al. (2004) introduced evaluation factors for modeling context based on relevance to ubiquitous computing [5]. Unfortunately, as these researches are still in the early stage, they have not been able to efficiently evaluate context model.

In this paper, we propose unified context that represents user-centric context in terms of 5W1H (Who, Where, When, What, How, and Why) and evaluate it with seven evaluation factors, i.e. structure, composition/decomposition, interchange, unification, extensibility, and scalability. The unified context is classified into 'preliminary context', 'integrated context', 'final context, and 'conditional context' according to the subject of generating and exploiting context, i.e. sensor, service, or user. From sensor's view, the preliminary context represents factual information about users in a service environment. From user's view, the conditional context depicts contextual condition that users specify in services corresponding to their desire. From service's view, the integrated context provides accurate information by means of fusing several preliminary contexts, and the final context triggers a service only if a correspondence occurs between integrated contexts and conditional contexts.

For implementing context-aware applications, unified context has the following advantages: First, unified context can simply model a part of user's situations in environments by assorting complicated information into six categories. From an experiment investigating the usage frequency of each element of 5W1H, unified context provides a basic-element set of user-centric context. Secondly, the unified context can enable sensor, user, and service to differently generate or exploit a defined 5W1H-semantic structure. Unified context is differently generated or exploited by dynamically building 5W1H's elements according to context hierarchy, i.e. preliminary, conditional, and integrated/final context. Finally, unified context is structured with elements and attributes so that any service can easily exploit the context for improving

This work was supported in part by the Seondo project of Ministry of Information and Communication in Korea.

Seiie Jang is with GIST U-VR Lab., Gwangju 500-712, S. Korea (e-mail: jangsei@gist.ac.kr).

Dong-oh Kang is with ETRI, Deajoen 305-350, S.Korea (email: dongho@etri.re.kr).

Woontack Woo is with GIST U-VR Lab., Gwangju 500-712, S. Korea. (corresponding author to provide phone: +82-62-970-2226; fax: +82-62-970-2249; e-mail: wwoo@gist.ac.kr).

service. From an experiment comparing response time, attribute-exploited services react faster than non attribute-exploited services.

This paper is organized as follows: In section 2, we briefly introduce the user-centric context and the context-model requirement. In section 3, we propose unified context and explain how to represent and to exploit unified user-centric context. Experimental results and analysis on context model are described in section 4. Finally, the conclusion and future works are discussed in section 5.

II. Background

A. User Centric Context

The goal of context in computing environments is to improve interaction between users and applications. This can be achieved by exploiting context, which works like implicit commands, enables applications to react to users or surroundings without user's explicit commands [6][7]. In ubiquitous computing domain, many definitions for context have been conceived. These definitions usually fall into two categories: first one is enumeration of context examples, or categorization [8][9][10]. The second one takes a more operational approach with a generic definition of context [3][11]. In case of the former, however, it is difficult to exploit context information out of definition's range. In case of the latter, it is not easy to share context among heterogeneous applications because of definition's generality.

To solve these problems, we define user-centric context in context-aware computing as "user-centric information among a variety of contexts in service environments that is interpreted, in terms of 5W1H, by applications". The user-centric context focuses on more user than the physical or the computational environments. This is because user-centric context can play an important role in providing fundamental clues about user's implicit commands to trigger services with an assumption that users always shed interactive information with services to environments. In addition, the definition gives a way to simply model a user's context by assorting complicated information into six categories. This context can be practically applied to context-aware applications.

B. Seven Requirements for Modeling Context

In ubiquitous computing environments, context is gathered, stored, and interpreted at different parts of the context-aware system. We have surveyed previous researches [4][5] and summarized seven evaluation factors in terms of:

•***Structure***: Context should be structured to represent huge contextual information of a user. A structured modeling provides a way to filter relevant information and to reduce context ambiguity by manifest labels. In addition, a structured model is necessary for context hierarchy since a context is generated, interpreted, and exploited differently according to sensors, users, and services [4].

•***Composition/Decomposition***: Many ubiquitous computing systems are derivative from a distributed computing system. Therefore, context model requires context composition in order to generate or to exploit context accurately by merging several contexts from distributed sensors or services. Besides, context decomposition is also essential to provide only the contexts required by each service among distributed services in an environment [4].

•***Interchange***: Context should be exchangeable among different components of the system, i.e., between sensors and services, and among services. Interchanging context enables a service to directly exploit contexts generated by sensors. Furthermore, context model needs to ensure that a service shares context in order to harmoniously cooperate with other services without conflicts. Therefore, context model requires interchanging context among context-aware entities [5].

•***Unification***: It is highly desirable for each participating party in context-aware computing to share the same interpretation of context. Unfortunately, complexity of context model or interpretation is increasing since a variety of contexts are used according to specific purpose of services. As a first step to unify context, essential items of context should be determined to harmonize services and resolve conflicts.

•***Extensibility***: The set of elements and attributes for representing context that will satisfy all future applications cannot be identified today. To ensure extensibility of context representation, the context must support methods of adding, modifying and deleting a set of elements and attributes for future extensions [4][5].

•***Uncertainty***: The set of context describing situations of users in service environment is usually incomplete or ambiguous. Joint processing of contexts with varying uncertainty results in vague interpretation of context. To keep the uncertainty as low as possible, context model provides a way to indicate incompleteness of the contextual information [5].

•***Scalability***: In ubiComp environments, heterogeneous sensors and services run simultaneously, and service environments around a user change dynamically. Context model requires context scalability so that context works reliably with the increasing number of sensors and services. In addition, context model allows context to be reused in more diverse application areas than specific application area.

III. Modeling Unified User-centric Context

A. Unified User-Centric Context

"User-centric context" refers to information that decides which service and which action of the service will be automatically triggered. According to the subject of exploiting user-centric context, it can be folded into three kinds of context; sensor, user, and service. A context of sensor's view, what we call preliminary context, describes the physical state of users in service environments. A context of user's view, what we call conditional context, is a set of conditions to appropriately trigger services that a user requires. A context of service's view, what we call final context, represents a reason that a service is automatically provided by observation of physical states satisfying user's requirements. Context-aware applications transform such contextual information into context by means of the context flow from sensor to service and from user to service. Then, context-aware applications exploit context to trigger a service appropriately.

To easily generate user-centric context, it is necessary to uniformly represent contextual information about users in service environments without depending on specific sensors or services. 5W1H is a popular way to uniformly describe a fact with "Who, What, Where, When, How and Why". 5W1H that is applied to user-centric context can depict "a certain user (Who) is", "in a certain location (Where)", "in a certain time (When)", "paying attention to a certain object/service (What)", "making a certain expression with physical signs (How)", or "because of a certain intention or emotion (Why)". This unified context enables sensor, user, and service to differently generate or exploit a uniformed 5W1H-semantic structure. A semantic structure of unified context from sensor's view is "someone$_{\text{Who}}$ is paying attention to something$_{\text{What}}$ or representing a certain expressions with some signs$_{\text{How}}$ in some location$_{\text{Where}}$ on some time$_{\text{When}}$". A semantic structure of unified context from user's view is "I want to get a certain service$_{\text{What}}$ if I$_{\text{Who}}$ am in a certain location$_{\text{Where}}$, on a certain time$_{\text{When}}$, with some expression$_{\text{How}}$ or in a certain mood$_{\text{Why}}$". A semantic structure of unified context from service's view is "a certain service$_{\text{What}}$ is automatically triggered if a user$_{\text{Who}}$ is in a certain location$_{\text{Where}}$, on a certain time$_{\text{When}}$, with a certain signs$_{\text{How}}$ or in a certain mood$_{\text{Why}}$". Because all unified contexts share basic elements (5W1H) representing user-centric context, they can be easily converted to each other. Therefore, unified context, in terms of 5W1H, enables applications to recognize and exploit context conveniently.

B. Modeling Unified User-Centric Context

Unified context is categorized into preliminary, integrated, final, and conditional context. Each context consists of elements representing 5W1H and attributes describing the features of each element. Elements depict user's contexts in service environments. Attributes provide the meta-data related to an element or relationship with other elements. Examples of the attribute include 'Generator', 'Confidence' and 'Uncertainty'. 'Generator' provides the identity of a sensor or service that makes elements of 5W1H. 'Confidence' means the physical or logical quality of being certain of abilities or capacities (i.e. 0~100%) of the generator. 'Uncertainty' describes the range (i.e. 0~100%) within which correct values of the element have a specified probability of being found. It is necessary to refer to attributes during the context flow in order to generate or exploit user-centric context more accurately. This is because even the same element has different values of confidence and uncertainty according to the context-aware subject. In addition, all contexts have 'Working Area' as an attribute that refers to the available range (e.g. geographical region or ownership between user and sensor/service, etc) of the context in a service environment. By exploiting 'Working Area', sensors deliver preliminary contexts to all services in the same working area and services share final contexts with each other. Especially, all sub-elements of final context have a special attribute, 'HitRatio', that is each element's ratio of correspondence between integrated and conditional context. This is used to choose elements with high-ratio value among complicated sub-elements of 5W1H for simplifying the context comparisons.

- **Who**

'Who' element of 5W1H provides identification of a user in service environments and is a basis for processing a set of 5W1H. As shown in Figure 1, 'Who' consists of 'Name' and 'Profile' as sub-elements. 'Name' has user name as a value and 'UID' and 'PWD' as attributes. 'UID' and 'PWD' refer to meta data for accessing services in home or office environments, e.g. social security number. 'Profile' is personal information that is open to environments by users. Personal information (e.g. favorite service lists, social relationships, etc) plays an essential role in triggering services harmoniously. However, it is difficult for applications to recognize personal information and to exploit it without user's permission. Therefore, 'Profile' enables users to make such personal information open to service environments. 'Profile' consists of 'FavoriteService' and 'Relationship'. 'FavoriteService' has service list as an element and 'At' and 'Type' as attributes. 'At' refers to service environments such as home and office. 'Type' alludes to a sort of service, e.g. movie, music, light, etc. 'Relationship' has social relationship with persons as a value and 'Person' and 'Priority' as sub-elements. 'Person' refers to identification of other person and 'Priority' shows the priority over the person. In case of 'Priority', it has a value range from -100 to 100. The positive means a user has higher priority than another and the negative means a user has lower priority than another.

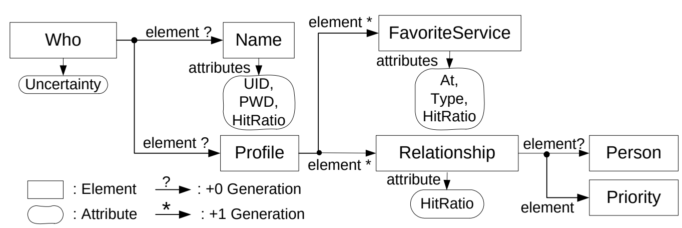

Figure 1: 'Who' Element Modeling

- **Where**

'Where' element of 5W1H gives a user's location in service environments. The location plays an import role in accessing available services near a user because all applications operate in a working range. As shown in Figure 2, 'Where' contains 'Location'. Among various ways to represent location, coordinates-based and symbolic methods are popular in context-aware applications. 'Location' consists of 'Coordinates' and 'Symbol'. In addition, it has 'Type' as an attribute which refers to whether a location is for indoor or outdoor places. 'Coordinates' has 'X', 'Y', and 'Z' sub-elements representing a 2D or 3D position. This also has 'Granularity' and 'Origin' as attributes. 'Granularity' specifies a unit of coordinates, e.g. centimeter (cm) or meter (m). ‘Origin’ refers to the origin of coordinates such as the door. ‘Symbol’ has abstract-level location corresponding to coordinates as a value and ‘Reference’ as an attribute. ‘Reference’ specifies a translator which converts a coordinates into symbol.

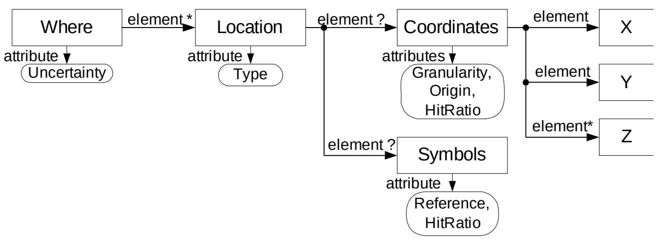

Figure 2: ‘Where’ Element Modeling

• When

‘When’ element of 5W1H represents time when a context is available, i.e. a given context is generated or valid at a given time or interval. As shown in Figure 3, ‘When’ consists of ‘TimePoint’ and ‘Interval’ as sub-elements. ‘TimePoint’ has time point as a value. ‘Interval’ consists of ‘From’ and ‘To’ that represents time duration. All contexts can be indexed by time information and then they can be exploited such as indexing contexts with time for context pattern of triggering services. There are various ways to represent time like location. Among them, absolute and symbolic representations are popular. Both ‘TimePoint’ and ‘Interval’ have ‘Type’ and ‘Reference’ as attributes. ‘Type’ refers to whether time is absolute or symbolic. If time is symbolic, ‘Reference’ indicates a translator that converts absolute into symbolic time.

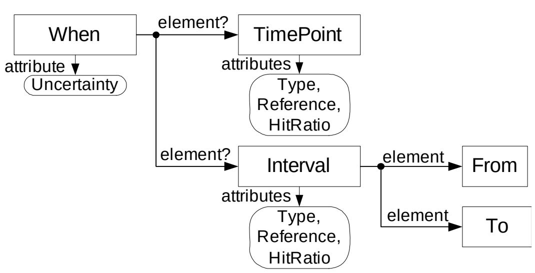

Figure 3: ‘When’ Element Modeling

• What

‘What’ element of 5W1H is information of an object which a user is paying attention to. As shown in Figure 4, ‘What’ consists of ‘Destination’ and ‘Manipulation’ as sub-elements. ‘Destination’ is comprised of ‘Identity’. ‘Identity’ has object name as a value and ‘Type’ as an attribute. ‘Type’ refers to a sort of service which the object provides. ‘Identity’ contains ‘Conflict’ as sub-element. ‘Conflict’ has identity of other object that the attended object collides with if they are triggered simultaneously, e.g. TV and Audio. In addition, it contains ‘Priority’ as sub-element that refers to the priority over the conflicted object. ‘Manipulation’ provides operational information of the object which a user is interested in. So it consists of ‘Function’ and ‘Parameter’. ‘Function’ has operations of the service as a value and ‘Parameter’ contains parameters that a function uses as input.

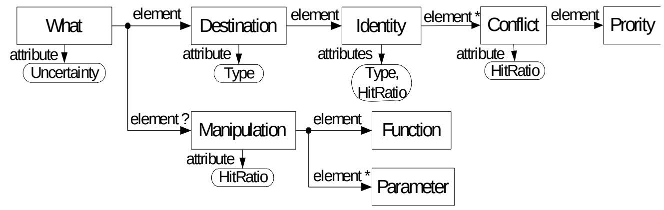

Figure 4: ‘What’ Element Modeling

• How

‘How’ element of 5W1H depicts a user’s expression with signs such as behaviors or bio-signals. As shown in Figure 5, ‘What’ consists of ‘Behavior’ and ‘BioCondition’ as sub-elements. ‘Behavior’ is comprised of ‘Gesture’, ‘Action’ and ‘Activity’. ‘Gesture’ has movements of the hands, legs, and body as values. ‘Action’ contains high-level information that is translated from successive gestures. ‘Activity’ shows abstract information that is interpreted from some actions. The reason why user’s behaviors are categorized into gesture, action, and activity is that they represent a hierarchy of user’s behaviors. For example, “a user gets a right hand up and down” is a gesture. However, if the gesture successively occurs, it is an action that means “a user swings right hand”. Also, if the action repeats for a period, an activity occurs with a meaning, “a user exercises”. ‘BioCondition’ provides indirect information of user’s expression through the changes of bio-signals such as pulses, temperature, and galvanic skin response. So ‘BioCondition’ consists of ‘PPG (photoplethysmogram)’, ‘GSR(galvanic skin response)’ and ‘SKT(skin temperature)’. All sub-elements of ‘BioCondition’ have ‘Type’ as an attribute because each bio-signal is differently represented according to measurements.

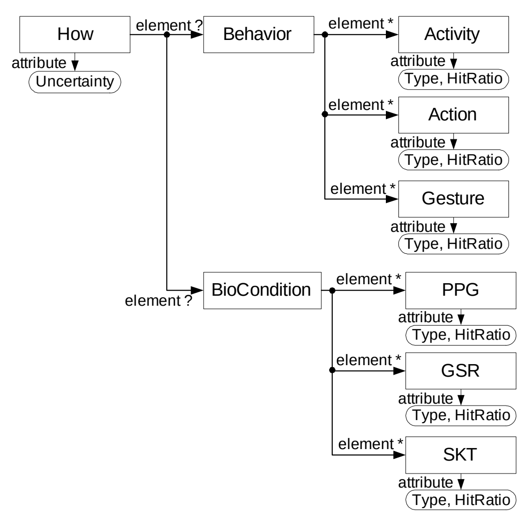

Figure 5: ‘How’ Element Modeling

• Why

‘Why’ element of 5W1H represents a mental state of the user such as intention or emotion. Because intention or emotion cannot be detected by sensors, it is difficult to represent a state of user's mind correctly. The goal of using 'Why' is not to be aware of full mentality but to provide a clue to trigger a service or to get user's feedback on given services. As shown in Figure 6, 'Why' consists of 'Intention' and 'Emotion' as sub-elements. 'Intention' represents user's mental states to manipulate service, e.g. turn on, tune off, etc. 'Emotion' represents user's response to given services such as good, bad, happy, and unhappy. Both 'Intention' and 'Emotion' contain 'Type' as an attribute because they are represented with various values and in categories.

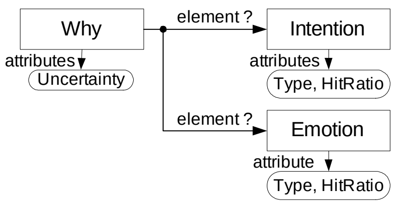

Figure 6: 'Why' Element Modeling

C. *Modeling Unified User-Centric Context*

Unified context is identified as 'preliminary', 'integrated', 'final', and 'conditional' context according to the subject of generating or exploiting context [2][12][13]. Especially, the proposed context model enables users to specify individual conditions to trigger services as shown in Figure 7, so that each user is provided with different actions of a service. All contexts are comprised of 5W1H elements with attributes. Preliminary context is a user-centric context of sensor's view and has a part of 5W1H. Conditional context is a user-centric context of user's view and depicts conditions that users specify in services corresponding to their desire. It may contain several 5W1Hs because a user sets many conditions in order to automatically trigger several operations of services. Integrated context is a user-centric context of service' view and generates a 5W1H by merging several preliminary contexts according to each element. Final context is also a user-centric context of service' view and decides which service is automatically triggered by correspondences between integrated and conditional context.

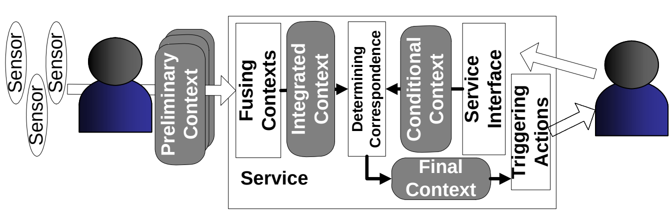

Figure 7: Unified Context: Preliminary, Integrated, Final, and Conditional Context

Preliminary context, generated by a sensor, represents contextual information about a user in a service environment. Figure 8 shows an example of preliminary context. From attributes of 'Context' element, this context is generated by 'ubiTrack', a sensor with confidence 80%, in working area 'B'. 'Who' element represents user's identification and certainty that a user is 'Seiie Jang' is 90% (or uncertainty: 10%). 'Where' element depicts that someone is located at (3,4) in coordinates whose origin is 'Door' and a unit of X and Y is 80cm. This location information is correct in certainty 60%. 'When' element indicates that this preliminary context is generated at 21:30 ~ 31, April 12, 2005 in certainty 80%. 'What' element provides information that the user gives attention on TV by detecting user's head position or eye's gazing in certainty 70%. This preliminary context can be interpreted with sensor's semantic structure as "Seiie Jang$_{\text{Who.Name}}$ is paying attention to TV$_{\text{What.Destination.Identity}}$ at (3,12) location$_{\text{Where.Location.Coordinates}}$ in 21:30~32$_{\text{When.Interval}}$". This unified context is delivered to all services in working area B.

```
<?xml version="1.0" ?>
<!DOCTYPE Context (View Source for full doctype...)>
<Context WorkingArea="area B" Generator="ubiTrack" Confidence="80">
<Preliminary>
<Who>
  <Name UID="731219-xxxxx" Uncertainty="10">Seiie Jang</Name>
</Who>
<Where>
   <Location Type="Indoor">
     <Coordinates Granularity="80cm" Origin="Door" Uncertainty ="60">
         <X>3</X><Y>12</Y></Coordinates>
      <Symbols Reference="ubiHome">TV</Symbols>
   </Location>
</Where>
<When>
  <Interval Type="Absolute" Uncertainty="20">
    <From>200504122130</From>
    <To>200504122131</To>
  </Interval>
</When>
<What>
  <Destination Type="Object">
    <Identity Type="MultiMedia" Uncertainty="30">TV</Identity>
  </Destination>
</What>
</Preliminary>
</Context>
```

Figure 8: Example of Preliminary Context

Conditional context is a set of conditions in order to trigger personalized service. This is specified by users and updated by services according to user's feedback on a given service. Figure 9 shows an example of conditional context. The context contains two conditions on TV and light services for a user, 'Seiie Jang'. Conditional context on TV is in certainty 100% since the user specifies it by means of 'WPS', a personal device. In case of light service, meanwhile, certainty is 50% because it is modified by light service. The user sets 'Who.Profile.Relationship' in order to provide TV with conflict-resolved information if the user and another user, 'Woontack Woo', access TV service simultaneously. With this context, TV provides higher priority to 'Woontack Woo' than 'Seiie Jang'. Also, the user specifies 'Where.Location.Symbols' and 'What.Destination.Identity' in order to represent that TV automatically takes an action if he is nearby TV and pays attention to TV. In addition, the user sets 'What.Destination.Identity.Conflict' to provide TV with conflict-resolved information if TV and Audio turn on simultaneously at the same working area. With this context, TV has higher priority to provide the user with a service than Audio. After specifying such conditions, the user sets 'What.Manipulation' for a kind of action of TV corresponding to the condition. This conditional context can be interpreted with a semantic structure of user's view as "I want to get

TV(Play: channel 9, volume 20)$_{\text{What.Manipulation}}$ if Seiie Jang$_{\text{Who.Name}}$ is paying attention to TV$_{\text{What.Destination.Identity}}$ around the TV$_{\text{Where.Location.Symbols}}$ and is standing$_{\text{How.Activity.Action}}$".


```
<?xml version="1.0" ?>
<!DOCTYPE Context (View Source for full doctype...)>
<Context Generator="WPS" Confidence="100">
<Conditional Service="TV" Uncertainty="0">
<Who>
  <Name UID="731219-xxxxx">Seiie Jang</Name>
  <Profile>
    <Relationship> Disciple
      <Person>Woontack Woo</Person>
      <Priority> -40</Priority>
    </Relationship>
  </Profile>
</Who>
<Where>
  <Location Type="InDoor">
    <Symbols Reference="ubiHome">TV</Symbols>
  </Location>
</Where>
<What>
  <Destination Type="Object">
    <Identity Type="MultiMedia"> TV
      <Conflict>Audio</Conflict>
      <Priority>30</Priority>
    </Identity>
  </Destination>
  <Manipulation>TV
    <Function>TurnOn</Function>
    <Parameter>Channel 9</Parameter>
    <Parameter>Volume 20</Parameter>
  </Manipulation>
</What>
<How>
  <Activity>
    <Action>Standing</Action>
  </Activity>
</How>
<Why>
  <Intention>Turn On</Intention>
</Why>
</Conditional>
<Conditional Service="Light" Uncertainty="50">
  <Who> <Name UID="731219-xxxxx">Seiie Jang</Name> </Who>
  <Where><Location Type="InDoor"> <Symbols Reference= "ubiHome">
  TV</Symbols> </Location> </Where>
  <What> <Manipulation> Set <Parameter>Brightness 2</Parameter>
    </Manipulation></What>
    </Conditional>
</Context>
```


Figure 9: Example of Conditional Context


```
<?xml version="1.0" ?>
<!DOCTYPE Context (View Source for full doctype...)>
<Context WorkingArea="area B">
<Integrated>
<Who Uncertainty="20">
  <Name UID="731219-xxxxx">Seiie Jang</Name>
</Who>
<Where Uncertainty="30">
  <Location Type="Indoor">
    <Coordinates Granularity="80cm" Origin="Door">
    <X>3</X> <Y>12</Y> </Coordinates>
    <Symbols Reference="ubiHome">TV</Symbols>
  </Location>
 </Where>
<When Uncertainty="10">
    <TimePoint Type="Symbolic">Evening</TimePoint>
    <Interval Type="Absolute">
      <From>200504122130</From>
      <To>200504122132</To>
    </Interval>
</When>
<What Uncertainty="30">
  <Destination Type="Object">
     <Identity Type="MultiMedia">TV</Identity>  </Destination>
</What>
<How Uncertainty="20">
    <Activity><Action>Standing</Action> </Activity>
</How>
<Why Uncertainty="40">
  <Intention>Turn On</Intention>
</Why>
</Integrated>
</Context>
```


Figure 10: Example of Integrated Context

Integrated context generated by a service is a result of fusing each element of 5W1H of preliminary contexts from all sensors in the same working area. Figure 10 shows an example of integrated context. Each 5W1H element of the context has different uncertainty. This is the reason why context-fusion method is different according to each element of 5W1H. In addition, sensors in working area B have their own confidence and all elements of a preliminary context have different uncertainty. As a result of selecting elements with high certainty (low uncertainty) among fused context in order to provide accurate user-centric context accurately, a set of integrated context is generated. This integrated context can be interpreted with a semantic structure of service's view as "Seiie Jang$_{\text{Who.Name}}$ is paying attention to TV$_{\text{What.Destination}}$ at (3,12) location$_{\text{Where.Location.Coordinates}}$ in 21:30~32$_{\text{When.Interval}}$ and is standing$_{\text{How.Activity.Action}}$ for turning on TV$_{\text{Why.Intention}}$".


```
<?xml version="1.0" ?>
<!DOCTYPE Context (View Source for full doctype...)>
<Context Generator="ubiTV" WorkingArea="area B">
<Final>
<Who>
  <Name UID="731219-xxxxx" HitRatio="95">Seiie Jang</Name>
  <Profile>
    <Relationship> Disciple
      <Person>Woontack Woo</Person>
      <Priority>40</Priority>
    </Relationship>
  </Profile>
</Who>
<Where>
  <Location Type="Indoor">
    <Coordinates Granularity="80cm" Origin="Door" HitRatio="40">
      <X>3</X> <Y>12</Y>  </Coordinates>
    <Symbols Reference="ubiHome" HitRatio="80">TV</Symbols>
</Location>
</Where>
<When>
  <TimePoint Type="Symbolic" HitRatio="60">Evening</TimePoint>
  <Interval Type="Absolute" HitRatio="30">
    <From>200504122130</From> <To>200504122132</To> </Interval>
</When>
<What>
  <Destination Type="Object" HitRatio="60">
    <Identity Type="MultiMedia"> TV
      <Conflict>Audio</Conflict>
      <Priority>30</Priority>
    </Identity>
  </Destination>
  <Manipulation> TV
    <Function>TurnOn</Function>
    <Parameter>Channel 9</Parameter>
    <Parameter>Volume 20</Parameter>
  </Manipulation>
</What>
<How>
  <Activity><Action HitRatio="80">Standing</Action> </Activity>
</How>
<Why>
   <Intention HitRatio="20">Turn On</Intention>
</Why>
</Final>
</Context>
```


Figure 11: Example of Final Context

Final context is generated by a service that searches conditional contexts in correspondences with a set of integrated context and then combines conditional and integrated context if accord occurs. Figure 11 shows an example of final context. To simplify comparison between integrated and conditional context, elements with higher hit ratio such as 'Who.Name', 'Where.Location.Coordinates', and 'How.Activity.Action' are determined with first priority. This final context is interpreted with a semantic structure of service's view as "TV (Play: channel 9, volume 20)$_{\text{What.Manipulation}}$ is triggered if Seiie Jang$_{\text{Who.Name}}$ is paying attention to TV$_{\text{What.Destination.Identity}}$ in the TV$_{\text{Where.Location.Symbol}}$ and is standing$_{\text{How.Activity.Action}}$ (for turning on TV)$_{\text{Why.Intention}}$". However, the service checks possibility of this service causing a certain conflict before triggering the service. For this, all services in the same working area share the final context with each other.

IV. Experimental Results & Analysis

A. Experiments

We have implemented 'ubiHome', a test bed for ubiComp-enabling home environment[13], where sensors generate preliminary context and services provide personalized service by exploiting user-centric context, as shown in Figure 12. A sensor detects the user's situation in home and then generates preliminary contexts. Examples include ubiKey[14], ubiFloor [15][16], SpaceSensor[17], ubiTrack[18], RFID Sensor [19][20], etc. A service takes appropriate actions according to a final context based on user-specified conditional contexts. Examples include c-MP (context-based Media Player)[14], c-Mail checker (context-based eMail cheker)[20], TMCS (Tangible Media Control System)[12], cPost-it (Context-based Post-it) [21][22], ubiTV[23], etc.

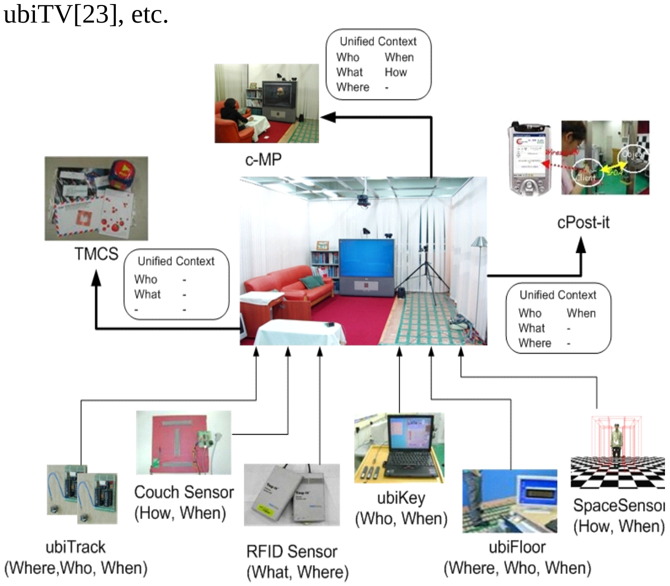

Figure 12: Example of Sensors and Services based on Unified User-centric Context

To investigate usefulness of unified user-centric context, we gathered integrated, final, and conditional context from services (i.e. cPost-It, c-MP, ubiTV, and TMCS) used by 5 volunteers (3 males and 2 females) for 30 days. All volunteers are in 20~30 years old range and are familiar with using the services. Each volunteer can specify a set of conditional contexts with his/her PDA. The PDA provides graphic interface which enables volunteers to select sub-elements among 5W1H for conditional contexts and to combine them with services. The conditional contexts are delivered from the PDA to all services in a working area where a user is located. After the user is out of the working area, the services remove the user's conditional contexts. A goal of this experiment is to determine basic sub-elements of 5W1H for modeling unified user-centric context. We measured the usage frequency of each element of conditional context for all services and analyzed it according to service and user. The reason why we investigate the usage frequency of elements of conditional context is that conditional context directly influences triggering of a service.

As shown in Figure 13, some sub-elements in 5W1H are referred more frequently than others by users to trigger services. In case of "Who" element, "Name" is the frequently used sub-element since users generally let a service to identify themselves for personalized service. In case of "Where", "Symbols" is mostly used as sub-elements since users prefer representing location symbolic with daily objects, e.g. home appliance, furniture, etc. In case of "When", "TimePoint" is mostly used because many services are triggered in synchronization with some events such as TV or radio program that start on specific time. In case of "What", "Identity" is mostly used to represent user's interested object and "Manipulation" is often referred to specify actions of service with proper parameters. In case of "How", "Gesture" is mostly used since users are familiar with manipulating simple services with hand or body gestures. In case of "Why" that is least frequently used among others of 5W1H, "Intention" is used to represent command for terminating or modifying services by force as user's feedback. Indeed, it does not mean that only these highly-frequently used sub-elements of 5W1H contexts can be shared by all kind of services. However, "Name" in "Who", "Symbols" in "Where", "TimePoint" in "When", "Identity" in "What", "Gesture" in "How" and "Intention" in "Why" are the basis of user-centric context since they are mostly referred by users regardless of service's purpose. Therefore, to simply represent user-centric context for services, it is necessary to investigate basic elements to trigger or manage service effectively while resolving service conflicts or linking services based on user's task.

To evaluate context representation for efficient comparison between integrated and conditional context, we measured average response time of two types of service; one type is based on a search method that finds correspondences between integrated and conditional context by exploiting an attribute, 'HitRatio', of sub-elements in final context. The other is based on a search method without exploiting the attribute.

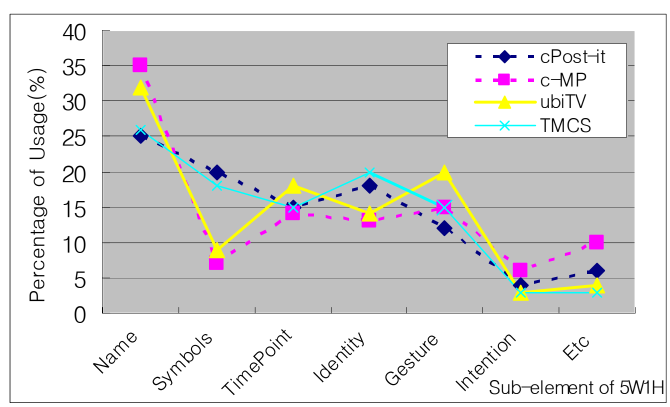

Figure 13: Number of usage of each elements of conditional context per service.

As shown in Figure 14, average response time of services is different for each service. All ‘HitRatio’ search-based services react faster to user’s context than non-‘HitRatio’ search-based services. This is because ‘HitRatio’ search-based services reduce search time, which determine a correspondence between integrated context and conditional context, by comparing sub-elements according to the order of ‘HitRatio’ priority as shown in Table 1. If a sub-element ‘Name’ of ‘Who’ element exists in an integrated context, in case of cPost-it, a set of conditional context having the equal value of ‘Name’ of ‘Who’ element of the integrated context is extracted. Then, a ‘Symbols’ of ‘Where’ is searched in the extracted set of conditional contexts since the ‘Symbol’ has the second highest ‘HitRatio’ among the other sub-elements. As a result, such a search process reduces more comparison time than that of non- ‘HitRatio’ search-based service. Especially, the difference between average response times is larger if the complexity of conditional context is increased, e.g. number of conditional context in a service, various sub-elements usage, etc. If the number of a set of conditional context in a service is *N*, complexity of search exploiting ‘HitRatio’ can be $O(N\log N)$. Otherwise, search complexity without exploiting any attribute such as ‘HitRatio’ is $O(N^2)$. As the number of a set of conditional context increases, the difference of average response time between ‘HitRatio’ search-based and non-’HitRatio’ search-based service become larger. Therefore, it is necessary to represent meta data (i.e. hit ratio, uncertainty, confidence, etc) as well as user-centric information (i.e. 5W1H) in context to provide faster context-aware services

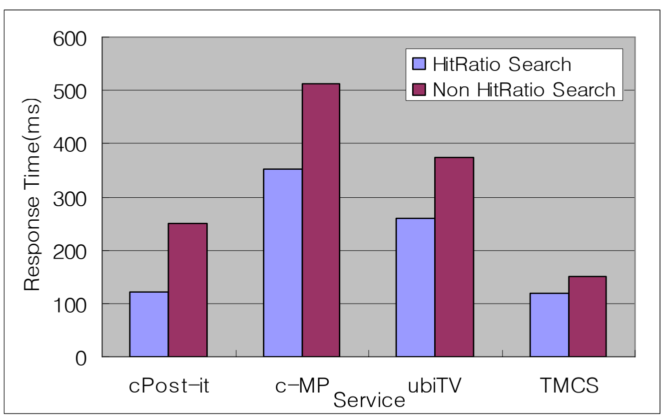

Figure 14: Comparison of average response time between ‘HitRatio’ search-based and non-’HitRatio’ search-based service.

Table 1: ‘HitRatio’ of basic sub-element per service

Basic Context Element	cPost-it	c-MP	ubiTV	TMCS
Name (Who)	26%	28.8%	22.3%	18.9%
Symbols(Where)	18.9%	18%	13.3%	15.5%
TimePoint(When)	16.5%	16.2%	17.8%	15.5%
Identity(What)	13.2%	10.8%	14.2%	18.9%
Gesture(How)	11.8%	9.3%	17.8%	16.3%
Intention(Why)	2.3%	3.6%	2.6%	1.7%
Etc	11.3%	13.3%	12.0%	13.2%

B. Evaluation of Unified Context

We evaluate the 5W1H-based context model with seven factors. Table 2 shows the results of the evaluation.

Structure of Unified Context: Unified context consists of elements and attributes. An element represents user’s situations and an attribute describes features of the element. Unified context enables an element to include sub-elements representing user-centric situations in details. In addition, all elements and attributes are labeled to reduce the ambiguity that may occur during interpreting context.

Composition/Decomposition of Unified Context: Unified context is classified into preliminary, integrated, final, and conditional context. This is adaptable to create, interpret, and exploit context in distributed computing environment. Pervasive sensors in daily life generate contextual information as preliminary context and deliver it to all services in the same working area. Each service composes preliminary contexts and interprets integrated context to trigger actions. To support such a process, unified context guarantees the composition of context. Also, user-specified conditional contexts are distributed to services that reside in same active area with the user. This requires unified context to support decomposition of context representation.

Interchange of Unified Context: Unified context guarantees the serialization of context representation because it is implemented by XML. Unified context enables any service to use context from any sensor in the same working area. In addition, it guarantees harmonized services that share the state of operation with others by exchanging final context.

Unification of Unified Context: Unified context represents user-centric context in terms of 5W1H and is exploited according to the view point of sensor, user, and service. 5W1H-based unified context has ability to simply represent user’s complicated situations. However, it requires formalizing relationships among sub-elements of 5W1H enabling users (or services) to easily specify(exploit) conditional(integrated/ final) context.

Extensibility of Unified Context: Unified context guarantees extensibility of context by means of structural representation that enables an element to contain sub-elements. However, there is a restriction that all elements must fall into six categories. In addition, unified context must be extended to represent user-centric information to both physical and computing environments.

Table 2: Unified Context Evaluation

Factor	Rating	Comment
Structure	High	Element/attribute pairs, Unambiguous element/attribute naming
Composition /Decomposition	High	Preliminary Context, Integrated Context, Final Context, Conditional Context
Interchange	High	By XML Serialization
Unification	High	By 5W1H
Extensibility	Middle	Structural Restrictions (only six categories and not easy modifying DTD)
Uncertainty	Middle	Various Attributes for Ambiguity
Scalability	High	Applied to home, wearable, and virtual environments

Uncertainty of Unified Context: Unified context represents a way to quantify level of the ambiguity of user-centric context because each element has special attributes such as confidence and uncertainty. However, it lacks quantitative measurement of confidence and uncertainty. To solve the problem, we need to standardize the level of context-awareness and requirement per context level.

Scalability of Unified Context: Unified context has been applied to several sensors and services working in a test bed. Furthermore, unified context representing user-centric context has played an important role in seamlessly connecting applications in heterogeneous area such as home[13], wearable[24], and virtual[25] environments.

V. Conclusion

In this paper, we proposed unified context that represents user-centric context in terms of 5W1H and evaluated it with seven factors. Experiments showed that the proposed context can simply represent a user's context in environments by assorting complicated information into six categories. Also, the unified context can enable sensor, user, and service to differently generate or exploit a defined 5W1H-semantic structure. Finally, unified context is structured with elements of 5W1H and attributes of each element so that any service can easily exploit the context for improving service. As a result, the proposed unified context enables context-aware services to quickly provide personalized services by exploiting unified user-centric context.

However, there is still a need to improve unified context in future works. This includes categorizing usage patterns of 5W1H according to kind of service and investigating useful meta data for better service performance. Furthermore, we should complement unified context since it has produced inefficient rate on extensibility and uncertainty among seven evaluation factors.

References

[1] B.N.Schilit, I.A.Norman and R.Want, "Context-aware computing applications," In the Proceedings of the 1st International Workshop on Mobile Computing Systems and Applications, pp. 85-90, Santa Cruz, CA, IEEE. (1994)

[2] A. Schmidt, KA Aidoo, A. Takaluoma, U. Tuomela, K. Van Laerhoven, and W. Van de Velde, "Advanced Interaction in Context," In the Proceedings of 1th International Symposium on Handheld. And Ubiquitous Computing (HUC99),. & Lecture notes. In computer science; vol. 1707, ISBN 3-540-. 66550-1; Springer. (1999)

[3] A.K.Dey, and G.D.Abowd, "Towards a better understanding of context and contextawareness," In the Workshop on the What, Who, Where, When and How of Context-Awareness, affiliated with the 2000 ACM Conference on Human Factors in Computer Systems. (2000)

[4] A. Held, S. Buchholz, and A. Schill, "Modeling of Context Information for Pervasive Computing Applications," In Procceding of the World Multiconference on Systemics, Cybernetics and Informatics. (2002)

[5] T. Strang, and C. Linnhoff–Popien, "A Context Modeling Survey," In Workshop on Advanced Context Modelling, Reasoning and Management associated with the Sixth International Conference on Ubiquitous Computing (UbiComp 2004), Nottingham/England, September. (2004)

[6] H. Leibernman, and T. Selker, "Out of Context: Systems That Adapt to, and Learn from, Context," IBM System Journal 39, 3&4, pp 617-632. (2000)

[7] A.Schmidt, "Implicit human-computer interaction through context," 2nd Workshop on Human Computer. Interaction with Mobile Devices. (1999)

[8] B.N. Schilit, and M.M. Theimer, "Disseminating active map information to mobile hosts," IEEE Network 8(5), pp.22-32, (1994)

[9] N.Ryan, J.Pascoe, and D.Morse, "Enhanced Reality Fieldwork: the Context-Aware Archaeological. Assistant," Computer Applications in Archaeology Gaffney,V., van Leusen, M., Exxon, Editors. Oxford. (1998)

[10] A.K.Dey, "Context-Aware Computing: The CyberDesk Project," In Proceeding of AAAI '98 Spring Symposium, Technical Report SS-98-02, pp.51-54. (1998)

[11] G.Chen, and D.Kotz, "A Survey of Context-Aware Mobile Computing Research," Technical. Report. TR2000-381. Department of Computer Science, Dartmouth College.(2000)

[12] S.Jang, and W.Woo, "ubi-UCAM: A Unified Context-Aware Application Model for 'ubiHome'", LNAI 2680, pp.178-189. (2003)

[13] S.Jang, C.Shin, Y.Oh, and W.Woo, "Introduction of "UbiHome" Testbed," The first Korea/Japan Joint Workshop on Ubiquitous Computing & Networking Systems. (2005)

[14] Y.Oh, and W.Woo, "A unified Application Service Model for 'ubiHome' by Exploiting Intelligent Context-Awareness", *2nd International Symposium on Ubiquitous Computing Systems*.(2004)

[15] S.Lee, and W.Woo, "THE INTERACTIVE DANCING FLOOR: THE SMART FLOOR PLAYS MUSIC," *WIAMIS03*, pp. 328-331. (2003)

[16] J.Yoon, S.Lee, W.Woo, and J. Ryu, "The User Identification System Using Walking Pattern over the ubiFloor," *ICCAS2003*, pp. 1046-1050. (2003)

[17] D.Hong, and W.Woo, "SpaceSensor:Real-Time Gesture Tracking for I-NEXT," *International Conference on Artificial Reality and Telexisitence(ICAT03)*, paper ISSN 1345-1278, pp. 95-99. (2003)

[18] S.Jung, and W. Woo, "UbiTrack: Infrared-based user Tracking System for indoor environment," *International Conferece on Artificial Reality and Telexisitence(ICAT04)*, paper ISSN 1345-1278, pp. 506-509. (2004)

[19] S.Jang, N.Zhang, and W.Woo, "Tangible Media Player with embedded RF tags," *International Conference on Human – Computer Interaction (HCII03)*, pp. 666-670. (2003)

[20] SJ.Oh, and W.Woo, "Manipulating multimedia contents with Tangible Media Control," *LNCS (ICEC)*, Vol.3166, pp. 57-67. (2004)

[21] S.Jang, S. Lee, and W. Woo, "Digital Message Sharing System in Public Places," *UbiComp03*, pp. 177-178. (2003)

[22] S.Jang, S.Lee, and W.Woo, "cPost-it: Context-based Information Sharing System," *LNCS(MIPS03)*, Vol.2899, pp. 352-363. (2003)

[23] Y.Oh, C.Shin, S.Jang, and W.Woo, "A Unified Context-aware Application Model for Ubiquitous Computing Environments", The first Korea/Japan Joint Workshop on ous Computing & Networking Systems. (2005)

[24] D.Hong, and W.Woo, "wear-UCAM: A Toolkit for Wearable Computing," *The first Korea/Japan Joint Workshop on Ubiquitous Computing & Networking Systems.* (2005)

[25] SJ.Oh, Y.Lee, and W.Woo, "vr-UCAM2.0 for Designing Reactive Cyber Space," The first Korea/Japan Joint Workshop on ous Computing & Networking Systems. (2005)

Seiie Jang received his B.S. degree in CS from Sogang University, Seoul, Korea, in 1997 and M.S. degree in CS from Sogang, Seoul, Korea, in 1999. He received his Ph. D. in Information an Communication from Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, in 2005. His research interests include context-aware, context-representation for ubiquitous computing, HCI, etc.

Eun-Jung Ko received her B.S. and M.S. degrees in computer science from Chungnam University, Korea, in 2002 and 2004 respectively. Since 2004, she has been working at Electronics and Telecommunications Research Institute, and she is a member of engineering staff of the Post-PC Platform Research Team. Her research interests include intelligent agent, ontology reasoning, digital library and semantic web.

Woontack Woo received his B.S. degree in EE from Kyungpook National University, Daegu, Korea, in 1989 and M.S. degree in EE from POSTECH, Pohang, Korea, in 1991. He received his Ph. D. in EE-Systems from University of Southern California, Los Angeles, USA. During 1999-2001, as an invited researcher, he worked for ATR, Kyoto, Japan. In 2001, as an Assistant Professor, he joined Gwangju Institute of Science and Technology (GIST), Gwangju, Korea and now at GIST he is leading U-VR Lab. His research interests include 3D computer vision and its applications including attentive AR and mediated reality, HCI, affective sensing and context-aware for ubiquitous computing, etc.