

ISSUES IN DEVELOPING CONTEXT AWARE INFORMATION SYSTEMS FOR HEALTH-CARE

Someswar Kesh, University of Central Missouri, kesh@ucmo.edu

ABSTRACT

Context-aware systems have spawned applications that did not exist few years back. In spite of that, many challenges remain to bring these systems to the mainstream of computing applications. This research critically examines applications of existing context aware systems, especially in health care, and discusses the issues involved in the development of such systems. It is expected that awareness of these issues will provide a roadmap for the developers.

Keywords: Context-aware systems; health-care applications; context-aware architecture, health care information systems

INTRODUCTION

Context-aware computing refers to applications and systems that can change its behavior based on the context the system is in (Schilit, Adams, Want, 1994). The definition of context goes well beyond location; it also includes such parameters as the presence of other people, accessible devices and objects, as well as time (Dey, Salbar, and Abowd, 2001). Obviously, the success of a context aware system depends on two parameters; identification of the correct context as well as responding with behavior appropriate to the context. For example, if a nurse and a doctor are in the same location, the context aware system may display different pieces of information on the nurse's device than on the doctor's device.

A major advantage of context-ware systems, particularly with reference to health-care applications has been the ability of such systems to understand context implicitly. Implicit context are those that humans being cannot provide explicitly (Abowd 1997). For example, a patient may not be able to seek help, but a system may detect the need for help and ask for it.

This paper discusses various uses that context aware information systems have in health care, examines the architecture behind context aware applications, and discusses the issues involved in the development of such systems.

Uses of Context-aware information systems in health-care

There are two major drivers for context-aware systems in healthcare. These are cost and the quality of service to the patient. In addition to the fact that patients may not be in a position sometimes to explicitly seek help, context aware systems are also useful in organizations because both patients and assets in health care organizations are highly mobile. For example, a patient might be in surgery and then moved to the intensive care unit, and equipment like wheel chairs may be scattered all over the hospital and significant time and money may be lost trying to locate them.

This research revealed two primary uses of context aware systems in healthcare. These are:

- Tracking
- Improving and smoothing workflow

Tracking: In health care, it is critical to track both assets and patients. Quick asset tracking not only reduces cost, but being able to track the right asset at the right time may indeed save lives in an emergency. Patient tracking systems are extremely useful for patients requiring special attention, for example, children, adults with dementia etc. (Cisco, 2012). Asset tracking can also reduce asset loss. For example, when a patient is leaving the hospital, an employee can be dispatched to retrieve a wheel chair.

Improve and Smoothing Workflow: Smooth workflow not only reduces cost but also reduces treatment errors. The difficulty of smoothing workflow with traditional computer based systems lies in the fact that healthcare clinical work is characterized as being hectic, nomadic, collaborative, ad hoc and interrupted (Center for pervasive healthcare). Traditional computer based systems can cope up only with some of these demands. The use of context based systems for workflow improvement has been very well illustrated by Kuwahara et. al (2009). Their study showed that the top five categories disturbing nurse's awareness are; interruptions by patients, illegible orders from doctors, wrong IV drip speed, wrong patients, and transcription errors.

To use context aware systems for error prevention, the authors studied the context of the nurse, the place where the activity typically takes place, the behavior of the nurse, and the items that interact with the nurse. For example, to mix IVs, the IV drip mixing area is used; the nurse mixes meds by using injection syringes, and uses pen, meds, IV bottles, and tray. The location of the nurse is detected by WLAN devices, and digital pens are used to strike out meds that the nurses are mixing. This combination can determine the context for the nurses, which is mixing IV drugs, and automatically turns off equipment that causes distractions, for example cell phones.

Architectural Issues

In this section we critically examine the various architectural issues with context aware information system. The architectural framework has been developed by analyzing multiple frameworks in the literature and is shown in Figure 1. For example, Computers in the Human Interaction Loop (CHIL) has agents (sensors), ontologies, and inference/reasoning techniques (Viterbo 2009). Most of the others have similar components. Based on that, the generic architectural components for context aware systems may be classified as:

- Hardware
- Data Storage
- Ontology and Knowledge Representation
- Reasoning and Uncertainty Handling
- Security

Hardware: The hardware portion of context aware systems are made of two critical components; the sensors themselves that collect data and a network infrastructure (Dey, Salbar, and Abowd 2001). Various types of sensors may be used. For example, Radio Frequency Identification (RFID) tags are the simplest sensors available. These are useful in relatively simple applications, like device tracking. Active badges are capable of collecting more complex information than RFID tags. The type of sensors used and the information the sensors collect will depend on the domain. Loke (2007) has developed a sensor architecture in which four issues related to sensors have been identified. These are:

- *What information will the sensors collect?*
- *What kind of sensor technology is available?*
- *What will be the location of sensors?*
- *How can multi sensor data be fused?*

Moreover, the sensors should be capable of communicating with each other and collaborate to form a network as well as seek context information from each other. Sensors should also be capable of monitoring their own behavior and change their behavior based on context.

To function properly, context aware systems need to have the properties of autonomous systems. They should be; self-configuring, self-healing, self-optimizing, and self-protecting. Self-configuring systems adapt automatically and dynamically to changing environments. Self-healing systems diagnose and correct system anomalies on their own. This keeps downtime to a minimum. Self-optimizing systems decide on the best course of action based on the situation and self-protecting systems manage access and can take defensive measures if it detects an attack.

Various agent models have been developed to communicate with web servers. For example, the CALMA agent model has the BDI framework with the following properties: (Loke 2007)

Rationality: represents the reasoning capabilities of agents

Mobility: Localized communication needed for communication with agents in other contexts.

Context awareness: agents may run on devices with limited resources. For example, the agent may run on a device with very low battery power.

Social Behavior: Agents communicate with another agent or service

Explicit Web Services Interaction: Enables management of web services interactions

Data Storage: The context data has to be stored in a database, so that the necessary data can be accessed whenever needed and from whatever location with very quick response time. The requirement for speed of access necessitates the use of a distributed database.

Knowledge and Ontological Representation: Various context models are available for representing context aware system knowledge. In particular, six types of context models have been defined and used. These are, attribute-value pairs, schema based models, graphic models, logic based models, object-oriented models and ontology-based models (Strang and Linnhoff-Popien, 2004). These authors have concluded that the more expressive the model is, the better it is for context-aware computing. Amongst these, ontology based models are the most expressive and have a rich vocabulary, and are most commonly used with context aware systems.

Ontologies can be classified based on the structure and knowledge representation. For example, the Context Aware Middleware for URC System (CAMUS) uses a four layered ontological space (Viterbo 2009). The common ontology layer uses knowledge of high abstraction that can be used by multiple applications. The domain ontology layer has knowledge specific to a particular domain. The infrastructure domain ontology holds the schema of the context model while the specific domain ontology provides knowledge of specific services. (Viterbo 2009).

Some knowledge is represented as first order predicates, which includes the type of context and the arguments for the context. For example, in the case of nursing workflow, the type of context may be the location of the nurse. The argument may be mixing – which describes the work to be done.

Other ontologies like CoBra use agents, their actions, devices, meetings, time and space. A similar but slightly different ontology is represented by SOUPA, which uses person, agent, belief-desire-intention, action, policy, time, space and event. It also has a rich set of vocabulary. SAMOA takes a slightly different approach, in which the ontology has been expressed as profiles. For example, the place profile can describe location, while the user profile can identify the user as well as a series of user activities. The discovery profile provides the awareness for the location (Viterbo 2009).

The framework for the nursing workflow model is based on the OWL ontology (Kuwahara 2009), which uses six classes to describe the nurses control flow. For example, the ANY class allows Job classes in a bag to be executed in unspecified order but not concurrently. Like other models, an error detection algorithm is also included. The

error detection algorithm is based on two concepts, the hypothetical reasoning algorithm and the concept of a scenario. An error occurs when a scenario is violated.

Because different systems may have different ontologies, a common set of ontologies may have to be developed. This is known as ontology alignment (Viterbo 2009).

The context infrastructure deals with three key issues. As raw context data is obtained from various sources, they come in a wide variety of formats. Semantic Space allows an explicit representation of the data in a uniform format. Expressive context queries allow selective data to be accessed by applications and reasoning capabilities allow inference of higher level contexts (Viterbo, et. al, 2009).

Reasoning and Dealing with Uncertainty: A wide variety of inference/reasoning techniques have been used in context aware systems. For example, the Gaia system which uses lower level context data to deduce higher level context information uses two inference approaches; rule based synthesizers and machine learning techniques like Bayesian learning. Rule based synthesizers use rules to determine knowledge and context, while Bayesian learning uses past context to train and learn. In many cases, inference and reasoning will also include dealing with uncertainty. Also, many of these systems have inference engines that are integrated with the particular ontological representation. For example, CoBrA's context reasoning uses the Jena rule engine, and the Java Expert System Shell (JESS). The reasoning systems are also equipped to deal with uncertainty. For example, if a new context is added to the knowledge base, the system first examines whether such context can be inferred using only ontological reasoning known as the Jena rules. If inconsistent data is found by the context reasoning system, theorist assumption based reasoning is used to resolve inconsistencies.

Reasoning can be achieved using some well-known artificial intelligence tools like fuzzy-logic and neural networks. However, unlike other artificial and knowledge based systems, location aware devices need to integrate location information in the reasoning component. Examples of such systems are the Nexus spatial world model and the Augmented World Query language (AWQL).

Reasoning with context information also requires dealing with uncertainty. Bayesian reasoning, Dempster-Shaffer techniques, fuzzy logic, have all been used to deal with uncertainty (Gu et. al 2004, Wu 2003, Byun and Chevest 2003).

Privacy and Security: Key issues of privacy and security will also have to be considered for location aware systems. Users should be able to manage their personal information and decide who gets access to what information and when.

Loke (2007) has contended that three issues influence the security model in context aware devices. These are:

Security levels will have to *adaptable*: Adaptable security means security will have to change based not only on the individual and the rights based on the individual but also on the situation they are currently in. For example, an off duty police officer in a social situation may not have access to individual information.

Security levels will also have to *traceable* not only to individuals and devices but also to the situation in which certain acts were performed and whether those acts were in the context in which they were performed.

Both adaptability and traceability may lead to finer grained security, where levels of security will be determined by the context itself.

Hager (2004) uses six types of contexts for security. These are:

Security level: This is set by the user. If the user sets a higher security level, a stronger security protocol will be used.

Energy: Low battery life leads to the use of a more energy efficient algorithm.

Location: Security levels may be set based on the location. If both the devices are at the home location, it may be deemed to be very secure. If one device is at an unknown location, it is considered very insecure and a very high level of security may be applied.

Communications: Proper design of link parameters like, throughput, link capacity etc.

Object Size: Larger files will require greater processing time, so more efficient security algorithms may be required.

User Interactions: The longer two parties communicate with each other; the greater will be the need for stronger security.

CONCLUSION

This research has investigated uses of context aware systems in healthcare and critically analyzed some of the architectural aspects of context aware systems. Successfully addressing these issues will be of paramount importance if context aware systems are to be widely used in healthcare. These include the hardware, data storage, knowledge and ontological representations, reasoning and uncertainty, as well as privacy and security. In the design of context aware systems these components will have to be considered for an effective and efficient system.

REFERENCES

1. Abowd, 1997: Context-awareness in wearable and ubiquitous computing, <http://smartech.gatech.edu/jspui/bitstream/1853/3531/1/97-11.pdf>
2. Byun, H.E., and Chevest, K., "Supporting Proactive Intelligent Behavior: The Problem of Uncertainty", Proceedings of the Workshop on User Modeling for Ubiquitous Computing", Pittsburgh Pennsylvania. (2003).
3. Centre for Pervasive Healthcare, <http://www.pervasivehealthcare.dk/>
4. http://www.cisco.com/web/strategy/docs/healthcare/CLA_HealthcareSolution.pdf
5. Dey, A.K., Salber, D., and Abowd, G.D., "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", Human-Computer Interaction Journal, Vol. 16, (2-4), (2001), pp. 97-166.
6. Gu et. al, "A Bayesian Approach for Dealing with Uncertain Contexts", Advances in Pervasive Computing, Austrian Computer Society, Vol. 176, (2004)
7. Hager, C.T., Context Aware and Adaptive Security for Wireless Networks, Ph.D. Thesis, Virginia Polytechnic Institute and State University, (2004).
8. Kuwahara et. al., "Verifying Nursing Activities Based on Workflow Model", Context Aware Computing and Self Managing Systems", W. Dargie (ed.), CRC Press, Boca Raton, Florida, (2009), pp. 15-41.
9. Loke, S., Context-Aware Pervasive Systems, Architectures for a New Breed of Applications, Auerbach Publications, Boca Raton, Florida, (2007).
10. Schilit, B., Adams, N. Want, R. "Context-Aware Computing Applications" 1st International Workshop on Mobile Computing Systems and Applications. (1994) pp. 85-90
11. Strang, T., and Popien-Linnhoff, "A Context Modeling Survey", 1st International Workshop on Advanced Context Modeling, Reasoning, and Management, Nottingham, England (2004).
12. Viterbo, et. al., "Managing Distributed and Heterogeneous Context for Ambient Intelligence", Context Aware Computing and Self Managing Systems", W. Dargie (ed.), CRC Press, Boca Raton, Florida, (2009), pp. 79-128.
13. Wu, H., Sensor Data Fusion for Context-Aware Computing Using Dempster-Shafer Theory", Ph.D. Thesis, The Robotics Institute, Carnegie Mellon University (2003).

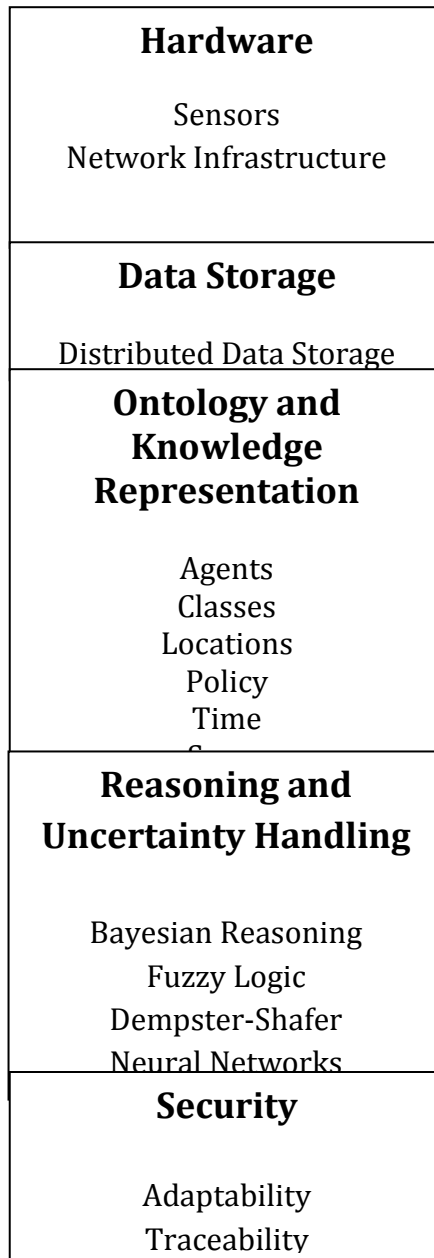


Figure 1: Architecture for Context Aware Systems