

iPower: An Energy Conservation System for Intelligent Buildings by Wireless Sensor Networks

Yu-Chee Tseng, You-Chiun Wang, and Lun-Wu Yeh

*Department of Computer Science
National Chiao-Tung University, Hsin-Chu, 30010, Taiwan
E-mail: { yctsen, wangyc, lwyeh }@cs.nctu.edu.tw*

ABSTRACT

Wireless sensor networks (WSNs) provide a convenient way to monitor the physical environment. Exploiting the context-aware capability of WSN to achieve energy conservation in intelligent buildings is an attractive direction. We therefore propose an intelligent and personalized energy-conservation system by wireless sensor networks (iPower). This system consists of some WSNs connected to a control server, some power-line control devices, and some user identification devices. A WSN is deployed in each room to monitor the usage of electric appliances and to help the control server to determine if there are electric appliances that can be turned off to reduce unnecessary energy consumption. The iPower system is quite intelligent and can adapt to personal need by automatically adjusting electric appliances to satisfy users' requirements. The design and implementation details of iPower are reported in this paper.

Keywords: context awareness, energy conservation, smart home/office, wireless communication, wireless sensor network.

1: INTRODUCTION

The discovery of electricity is one of the most important milestones in human history. Electricity is so essential in our daily life that people cannot live without it. However, today, energy has been over used and energy conservation has become a global problem that everyone has to be conscious of. According to the report [3], more than one third of energy consumption is spent on HVAC systems, which include heating, ventilating, air conditioning, lighting, and other related equipments. According to experiences, a large portion of energy consumed by HVAC systems is due to improper use of electric appliances. Therefore, how to exploit the context information of an environment to automatically control the usage of electric appliances in that environment has a great potential to reduce the waste of energy by HVAC systems.

In this work, we propose an *intelligent and personalized energy-conservation system by wireless sensor networks (iPower)* to reduce energy consumption of HVAC systems by exploiting the context-aware capability of sensors. In the iPower system, WSNs are

deployed in rooms of a building to collect information of the environment. Such information is reported to a control server to determine whether to turn off those unnecessary electric appliances in the building. Such systems need to be designed with user friendliness in mind to minimize the involvement of users in making decisions. As an example, when sensor nodes detect a low temperature or a high brightness in a likely unoccupied room, they can report to the server that the electric appliances in that room (e.g., air conditioners or lights) could be turned off. The server then sends an alarm signal to notify people in the room that the electric appliances could be turned off shortly. If there are still users in that room, they can signal the system that these appliances should not be turned off by triggering some events (such as making some voices, changing the light reading of any sensor, or moving any furniture attached with sensors). If there is no such intentional event made by human being detected in a predefined amount of time, the server will turn off the electric appliances in the room through some power-line control devices. In this way, the iPower system can conserve energy even if users are not wearing any particular badge.

In the iPower system, we also provide *personalized services* in which electric appliances can be automatically adjusted to satisfy users' preferences. In particular, each user can create a profile to describe his/her favorite temperature and brightness. Such users are considered priority users and need to carry user identification devices so that our system can retrieve their profiles. When there are priority users in a room, the server will adjust the air conditioners and lights in that room according to the profiles of these users.

The rest of this paper is organized as follows. Section 2 reviews some related works. Section 3 discusses the design of our iPower system. Section 4 presents the implementation details. Section 5 concludes this paper.

2: RELATED WORKS

WSNs have been widely used for providing context information in smart space/environment applications. How to automatically control electric appliances based on users' locations and requirements has been intensively discussed for smart homes/offices. The work in [7] considers a ubiquitous computing architecture in

which electric appliances are all controlled by a SIP (session initiation protocol) [6] server, under which architecture users can make calls to communicate with the SIP server to control their electric appliances. In the MavHome system [2], the mobility pattern of a user in a house can be computed, which is forwarded to the system to provide advanced services (e.g., controlling the corresponding electric appliances) in the predicted location that the user is moving toward. In Semantic Space [9], the authors propose some semantics to describe the environment, which can be used to query the status of the environment where users are located. The work in [4] proposes a context-aware smart house by sensors, and electric appliances can be automatically adjusted according to the environmental information collected from sensors. The issue of energy conservation, which is very critical to our environment, has not been well addressed, which motivates us to investigate the related issues.

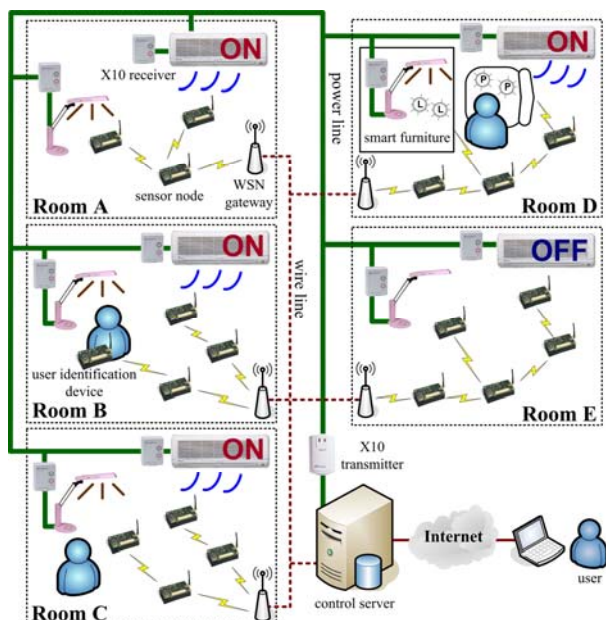


Figure 1. System architecture of iPower.

3: DESIGN OF THE IPOWERSYSTEM

3.1: SYSTEM ARCHITECTURE

The architecture of our iPower system is illustrated in Fig. 1, which includes some sensor nodes, several WSN gateways, a control server, some power-line control devices, and user identifications devices. Below, we describe the functions of each component.

- **Sensor nodes:** In each room, we deploy sensor nodes to monitor the environment. These sensor nodes will form a multi-hop WSN to collect information in the environment. In our current prototype, three types of sensing data can be collected, including light, sound, and temperature. An *event* is defined when the sensory input is higher or lower than a predefined threshold. To

conserve the energy of sensor nodes, reporting of events is reactive, in the sense that a sensor node will report its sensing data only when some predefined events occur. Different events can be combined to define a room's condition. For example, a low temperature (or a high brightness) together with some sound events in a room may indicate that the corresponding electrical appliances are turned on to serve users in that room; some sound events and change of the light degree read by sensors may indicate that users in that room are moving around; and a low temperature (or a high brightness) with no sound event for a certain amount of time may indicate that the air conditioners (or lights) in that room are unnecessarily turned on because no one is in the room. In the future, we plan to include more sensors to provide more intelligence to our iPower system. For example, as shown in Fig. 2, a smart study desk may include some pressure sensors underneath the cushion of a chair and some light sensors nearby the lamp on the desk. When someone is sitting on the chair, such an event can be detected by the pressure sensors, and the system can adjust the lamp according to the light degree nearby the lamp. When the user leaves the chair, the pressure sensors can detect the disappearance of the user and make energy conservation decision by notifying the server to turn off the lamp.

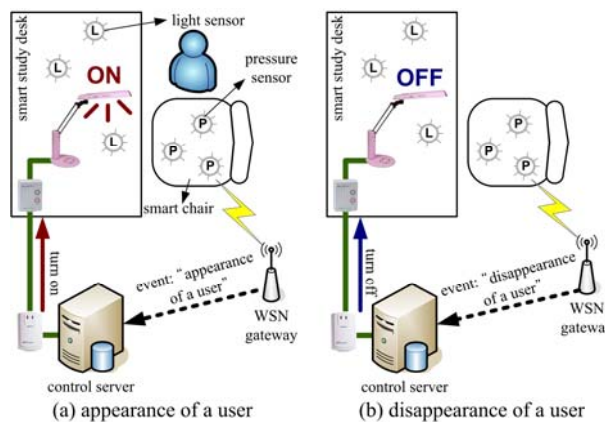


Figure 2. The "smart study desk" scenario.

- **WSN gateways:** The set of sensor nodes in each room will form a WSN. For each WSN, there is a WSN gateway. A WSN gateway has a wireless interface to talk to sensor nodes and a wire-line interface to talk to the control server. It has four major functionalities: issuing commands to nodes of the WSN, gathering data from nodes of the WSN, reporting the room's condition to the control server, and maintaining the WSN. Specifically, the gateway will notify nodes of the WSN to start collecting environmental information when it receives a *start* command from the server. After gathering sensing reports from the WSN, the

gateway will determine the room's condition and report to the server. To maintain the WSN, the gateway will periodically broadcast *heart-beat* messages to the network. A sensor node receiving such a message will reply an *alive* message to the WSN gateway. If the WSN gateway does not receive any *alive* message from a certain sensor node for a predefined amount of time, it will notify the server that the sensor node may be broken.

- **Control server:** The control server is to collect the system's status (e.g., rooms' conditions and sensors' states) and perform power-saving decisions. It maintains a database of user profiles and periodically checks the states of electric appliances in each room. It will decide whether to turn off an electric appliance in a room according to the sensory data collected from the room. It can also adjust the electric appliances in a room according to the profiles of users in that room. Such decisions or adjustments are achieved by sending commands through the power-line control devices to turn off or adjust electric currents of the corresponding electric appliances. The server also provides user interfaces to allow users to change their profiles and obtain the system's status.
- **Power-line control devices:** The power-line control devices allow us to turn off or adjust the electric currents of appliances in the iPower system. In our current implementation, we adopt the X10 products by SmartHome [8], which contains one X10 transmitter and several X10 receivers. The X10 transmitter can communicate with X10 receivers via power lines. In our iPower system, the X10 transmitter is connected to the control server to transmit the server's commands.
- **User identification devices:** The user identification devices are portable devices that can be carried by users so that the system can determine users' IDs and retrieve their profiles. It can be any identification device. In this work, we simply use the processor board of our sensor platform (without sensors) for user identification. When a user enters a room, his/her user identification device will join the WSN in that room and provide its ID to the server via the WSN gateway.

3.2: ENERGY CONSERVATION SCENARIOS

Next, we give five scenarios to demonstrate how the iPower system works in an intelligent building. Let us consider the five rooms in Fig. 1.

- **Room A: electric appliances are turned on but nobody is in the room.** In this case, since sensor nodes have detected a low temperature, a high brightness, and no sound event for some while, the WSN gateway will report to the control server that this room is *abnormal*, implying that electricity may be wasted in room A. The server will then send

an *alarm* message to room A, which will trigger the beepers attached to sensor nodes. These beeps are to announce that the system will turn off air conditioners and lights in room A in a few minutes. Alternatively, we can blink lights on and off to signal users that appliances in that room will be turned off soon. This is to avoid our system to make wrong decisions. Since there is no one in the room, the server will turn off these appliances after timeout to conserve energy.

- **Room B: electric appliances are turned on and somebody is in the room (with user identification device).** In this case, since the system can detect that the room is occupied, energy conservation commands will not be issued. So these electric appliances will remain on.
- **Room C: electric appliances are turned on and somebody is in the room (without user identification device).** In this case, energy conservation commands will be given depending on whether some events (such as sound events) indicating that the room is occupied can be detected or not. If there are such events, these electric appliances will remain on. If there are no sound events for a while, some signals (such as beeps or blinking lights) will be triggered to warn users in that room. In response, users can do some actions to signal the system that the room is occupied (such as making some noise by clapping, covering any sensor with a light sensor to change its light reading, or switching on or off any electrical appliance that is under control of the iPower system). As long as any of such events can be detected, the server can realize that the room is still in use and thus will not turn off the electric appliances. Note that to reduce bothering users too much, the interval to warn users will be increased in an exponential manner after each intentional event being generated by users in that room. Further, after several warning signals without success, the system will stop trying (to make energy conservation decisions) for a long period of time.
- **Room D: electric appliances are turned on in a room with smart furniture.** If there is smart furniture in the room, they can help detect the existence of people in the room. For example, if there is a person sitting on the smart chair, the system will keep on reporting that someone is on the chair, so no energy conservation commands will be issued. If the smart furniture is not in use, then the scenario in room C may apply.
- **Room E: electric appliances are turned off.** In this case, the WSN gateway will report to the server that the room is *normal* so that the server will not issue any energy-saving command.

3.3: SYSTEM OPERATIONS AND MESSAGE FLOWS

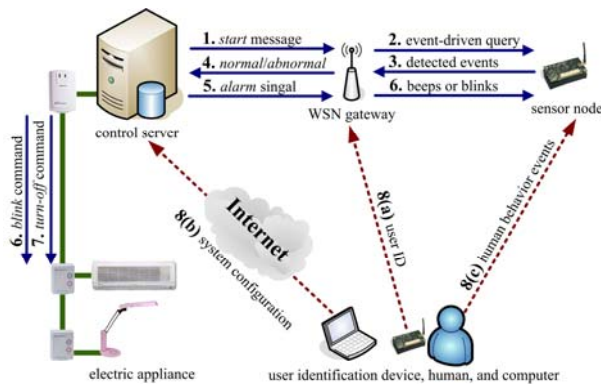


Figure 3. Message flows in the iPower system.

Fig. 3 illustrates the message flow and the interaction of system components in the iPower system. The details are discussed below.

1. The control server begins checking the usage of electric appliances in a room periodically or at predefined time according to the system configuration file. The server will then send a *start* message to the WSN gateway in that room.
2. On receiving the *start* message from the server, the WSN gateway will notify its sensor nodes by issuing some event-driven queries to begin collecting information of the environment. The WSN gateway then sets a timer and waits for sensing reports from sensor nodes.
3. When a sensor node detects any event (such as a low temperature or a high brightness), it will report its sensing data to the WSN gateway.
4. If the WSN gateway receives any sensing report and any human behavior report from step 3 before its timer expires, it can determine the room's status according to the following rules:
 - If any piece of smart furniture reports that someone is using it, the WSN gateway will report a *normal* status to the server. However, if it is reported that a user leaves any smart furniture, the WSN gateway will repeat steps 2–4 to check the room's status again.
 - If sensors report any human behavior (such as a sound event or the change of light readings), the WSN gateway will report a *normal* status to the server. However, it will also notify the existence of people to the server and thus the system will check this room's status later on.
 - Otherwise, the WSN gateway reports an *abnormal* status to the server to indicate that the electric appliances in the room may be turned on unnecessarily.
5. When the server receives an *abnormal* report from the WSN gateway, it will warn the people (if any) in the corresponding room by sending an alarm message to the WSN gateway.

6. Once receiving the *alarm* message, the WSN gateway will command its sensor nodes to turn their buzzers to generate a beeping sound. Alternatively, the server can send a *blink* command to the X10 receiver to blink any light on and off for a short period of time. These actions are used to notify people in the room that the server will turn off the electric appliances after ten minutes.
7. If the server does not receive any human behavior event from the room after ten minutes, it knows that there is no one in that room and will turn off the electric appliances by sending a *turn-off* command to the X10 receivers in that room.
8. If there is any user in the room hearing the beeping sound or seeing blinked light, he/she can notify the server that the room is still in use by one of the following three methods:
 - (a) If the user has carried a user identification device, the device will directly inform the server (via the WSN gateway) his/her ID. In this case, the user does not need to take any action.
 - (b) If the user can access the Internet, he/she can log in the web page of the iPower system to set up the next checking time of this room so that the server will not disturb the user before he/she leaves the room.
 - (c) Otherwise, the user can make some intentional events by changing the room's environment, such as making some noise by clapping or turning off and then turning on any light. In this way, sensor nodes will detect an unusual sound or a change of light degree and thus report these events to the WSN gateway.

Based on these reports, the WSN gateway will notify the existence of users to the server and the system will back off and check the room's status later on. The next checking time can be set manually by users, by any default value (such as one hour), or in any typical exponential backoff mechanism.

3.4: PERSONALIZED SERVICES AND USER PROFILES

The iPower system also provides personalized services in which electric appliances can be automatically adjusted to satisfy users' preference. In particular, each user can specify his/her favorite temperature and brightness. When a user enters a room, the iPower system can adjust the air conditioners and lights to meet the user's preference. To achieve this goal, the user has to create his/her profile in the server's database and carry a user identification device when entering our system. The user's location is determined by the WSN gateway which collects the user's ID.

In our current implementation, we follow the XML (extensible markup language) [10] to describe user's profiles. The current definition is shown in Fig. 4. The profile includes user's ID, name, and several attributes with the user's favorite temperature and brightness.

```

- <User>
  <Id>001</Id>
  <Name>Yu-Chee Tseng</Name>
  - <Attribute>
    <Name>temperature</Name>
    <Value type="Float"/>
    <Range max="25" min="18">true</Range>
  </Attribute>
  - <Attribute>
    <Name>light</Name>
    <Value type="Float">50</Value>
    <Range max="0" min="0">false</Range>
  </Attribute>
</User>

```

Figure 4. An example of user profile.

3.5: PROTOCOL STACK

To implement the iPower system, we design a protocol stack as shown in Fig. 5, which consists of the following layers:

- **User layer:** The user layer defines the rules that a user can access the system through the user interface. Here we consider two types of users: *administrators* and *end users*. An administrator can add or remove equipments (e.g., electric appliances, sensors, and power-line control devices) in the system, change their attributes and profiles, and manage end users. An end user can create and modify his/her user profile.
- **Service layer:** The service layer defines the rules by which the system provides and manages its services. Here we follow the rules defined in *OSGi (open service gateway initiatives)* [5], which is a service-oriented architecture for networked systems. An OSGi service platform provides a standardized, component-oriented computing environment for cooperating networked services. Using this architecture can help reduce complexity to build and maintain applications. Following OSGi, the service layer is separated into *service component* and *service management*, where the former defines the services provided by the system, while the latter provides a management mechanism to maintain these services. In our current implementation, three service components are defined, including *profile setting service*, *device controller service*, and *sensor handler service*. The profile setting service is used to create and modify a profile, while the device controller service and sensor handler service are used to control the power-line control devices and sensor nodes, respectively. To

manage the services, a new service component must first *register* to the server and the administrator can obtain the statuses of all service components in the system by the *service discovery* mechanism.

- **Profile layer:** The profile layer maintains all user profiles and other profiles for sensors, power-line control devices, and rules. The sensor profiles describe the locations (in which room) and sensing types of sensors. The device profiles describe the electric appliances controlled by the power-line control devices. Finally, the rule profiles define how our system components interact with each other. All profiles are depicted by XML.
- **Sensor layer:** The sensor layer controls the actions of sensor nodes. These actions include executing commands from the WSN gateway (such as to detect events and to generate beeping sounds) and reporting sensing data to the WSN gateway.
- **Actuator layer:** The actuator layer controls the actions of power-line control devices, including turning off and adjusting the electric currents of appliances.

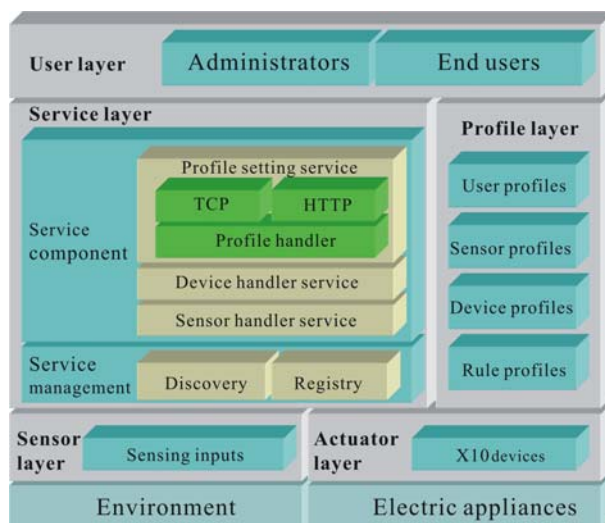


Figure 5. Protocol stack of the iPower system.

4: IMPLEMENTATION DETAILS

4.1: HARDWARE SPECIFICATION

We use the MICAz Motes [1] as sensor nodes. The MICAz is a 2.4 GHz, IEEE 802.15.4-compliant Mote module that enables low-power operations and offers a data rate of 250 kbps with a DSSS radio. Each sensor node has a sensing board that can collect sensing data including light, sound, and temperature from the environment. More sensing capabilities can be added by replacing a node's sensing board. Each sensor node also

has a buzzer so that it can generate a beeping sound when being commanded by the WSN gateway.

For the power-line control devices, we adopt the X10 products by SmartHome [8]. The X10 devices consists of X10 transmitters and X10 receivers. They can communicate with each other by the X10 communication protocol, which encodes messages on the electric signal with a frequency of 60 Hz. With the X10 communication protocol, an X10 transmitter can send commands to an X10 receiver through a power line. To control electric appliances, we connect an X10 transmitter to the computer via an RS-232 interface and connect electric appliances with X10 receivers, as shown in Fig. 6. Each X10 receiver has a unique address and at most 256 addresses can be selected.



Figure 6. The X10 power-line control devices.

4.2: USER INTERFACE

We provide a user interface to manage the system and let users create their profiles at the remote control server, as shown in Fig. 7. The user interface has a *monitor*, a *configure*, and an *attribute* areas. The monitor area shows the deployment of sensor nodes and electric appliances controlled by the X10 devices in a room. The configure area provides an interface to let administrators add new equipments in a room, including sensor nodes, X10 devices, and electric appliances. Finally, the attribute area is used to modify the attributes of equipments and the profiles in the system.

5: CONCLUSIONS

In this work, we have proposed an iPower system that is designed for energy conservation in an intelligent building and provision of personalized services for environment control. The iPower system can detect if there is possible waste of electricity by WSNs and then turn off these unnecessary electric appliances via the X10 power-line control devices with a user-friendly design. The iPower system also provides personalized services in which electric appliances can be automatically adjusted to satisfy users' requirements. With the iPower system, we can avoid unnecessary

electric electricity consumption of the HVAC system and thus achieve energy conservation.

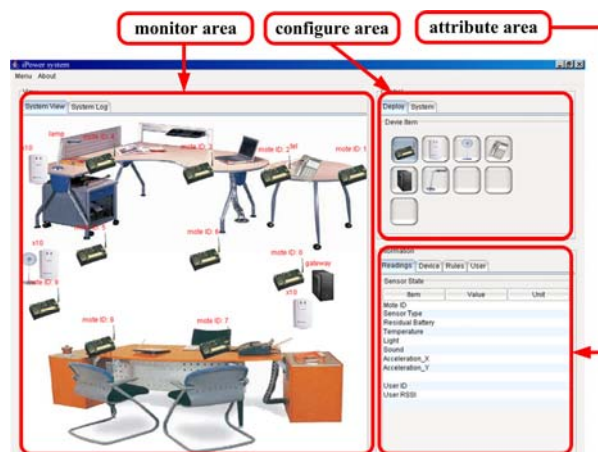


Figure 7. User interface at the control server.

ACKNOWLEDGEMENTS

Y.-C. Tseng's research is co-sponsored by Taiwan's MoE ATU Program, by NSC under grant numbers 93-2752-E-007-001-PAE, 95-2623-7-009-010-ET, 95-2218-E-009-020, 95-2219-E-009-007, 94-2213-E-009-004, and 94-2219-E-007-009, by MOEA under grant number 94-EC-17-A-04-S1-044, by ITRI, Taiwan, and by Intel Inc.

REFERENCES

- [1] Crossbow, "MOTE-KIT2400," <http://www.xbow.com/>.
- [2] S. K. Das, D. J. Cook, A. Battacharya, E. O. H. III, and T. Y. Lin, "The role of prediction algorithms in the mavhome smart home architecture," *IEEE Wireless Communications*, vol. 9, no. 6, pp. 77-84, 2002.
- [3] O. Gassmann, H. Meixner, J. W. G. J. Hesse, and W. Gopel, *Sensors in Intelligent Buildings: Sensors Applications*. Wiley-VCH, 2001.
- [4] S. Helal, W. M. H. E. Zabadani, J. King, Y. Kaddoura, and E. Jansen, "The gator tech smart house: a programmable pervasive space," *IEEE Computer*, vol. 38, no. 3, pp. 50-60, 2005.
- [5] OSGi Alliance, "OSGi - The Dynamic Module System for Java," <http://www.osgi.org/>.
- [6] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," IETF RFC 3261, June 2002.
- [7] H. Schulzrinne, X. Wu, S. Sidiroglou, and S. Berger, "Ubiquitous computing in home networks," *IEEE Communications Magazine*, vol. 41, no. 11, pp. 128-135, 2003.
- [8] SmartHome, "X10-1132B," <http://www.smarthome.com/>.
- [9] X. Wang, J. S. Dong, C. Y. Chin, S. R. Hettiarachchi, and D. Zhang, "Semantic space: an infrastructure for smart spaces," *IEEE Pervasive Computing*, vol. 3, no. 3, pp. 32-39, 2004.
- [10] WorldWideWeb Consortium (W3C), "Extensible Markup Language (XML) 1.0," <http://www.w3.org/TR/REC-xml/>.