Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 129-140

RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

Preventive customized strategies for saving energy on smartphones

Ahmed Sameh and Abdullah Almasri

Prince Sultan University, P.O.Box 66833 Riyadh KSA

Abstract: Current smartphones power saving strategies can be improved towards saving more power and/or gain more user satisfaction only if they start following "preventive" and/or user customized power saving plans. We develop a number of preventive power saving strategies which will help in saving the battery power without the need of using the power of the same battery for detecting abusers (detective strategies). A key disadvantage about current power saving strategies is the layer by layer power saving management approach which takes place on several smartphones with lack of coordination between these layers. This also can be improved tremendously if there is a cross-layer middle ware that helps the layers to share data regarding power management on a smartphone. In order to implement this, the paper proposes an "ontology" that works as a cross layer middleware replacing current individual layers management strategy that is done layer by layer. The Ontology is used as a power-cognizant distributed middleware among the various layers to allow better monitoring and flexible controlling of smartphone components power consumption.

Keywords: Ontology, Mobile Apps, OWL, SPARQL, Preventive, Power Saving

1. Introduction

Research done on the user-application layer focus on providing or supporting energy-aware adaptation of applications as means to extend battery life and reduce energy consumption. There are three main directions investigated: measuring and profiling of application energy use, adaptive algorithms and adaptive scheduling and configuration. Measuring and profiling of application energy usage is done by taking into account profiling models for energy-intensive applications [2] [3] [6] or benchmarks of battery dissipation rates [7]. Approaches relying on benchmarks of battery dissipation rates typically aim to predict the power consumption of a given application based on known characteristics of the battery itself and known drain-rate curves for different instruction types used in the application [7]. For example, one study empirically determined benchmark battery dissipation rates for integer register operations (IReg), integer loads and stores (IMem), floating point register operations (FPReg), as well as floating point loads and stores (FPMem) [7]. Moreover, the same study investigated the effect on battery drain rates of full memory access for those operations, as opposed to the situation when only the cache is accessed [6].

Battery drain rates for different operations are combined and considered together with known characteristics of the battery in order to allow runtime systems and dynamic compilers to predict with reasonable accuracy the energy consumption of different applications [7]. Such prediction can be useful as a first step towards optimizing energy use of applications. Other studies focus on models that can be used to accurately represent energy usage on mobile devices. For instance, one study investigated the energy usage patterns of mobile devices when decoding and displaying video data in the MPEG format. Reporting that energy usage in this case can be accurately represented by polynomial models with good measures of least square fit [3]. Another study takes a slightly different approach, proposing a scheme to infer real life usage patterns that are then used to successfully predict future energy usage and to inform optimal energy-saving profiles [2]. In this case, the usage patterns are based on repeated observations of device operation and are broader in scope than the previous study, taking into consideration not only the use of video data, but all running applications and their CPU load, as well as other energy-intensive aspects such as display and network. Based on the usage patterns observed, different energy saving profiles are created, including actions such as reducing brightness when the battery is critically low or transferring data in bulk over a high speed network when the battery level is below 50%. Building on some of the results of the above approaches, adaptive algorithms go one step further and aim to provide concrete parameters that can be changed in order to improve energy efficiency of applications in different contexts. For example, a video downloading algorithm has been shown to reduce its energy consumption by around 60% when parameters such as buffer size, low water mark and socket-reading are tuned for energy efficiency [9]. In another example, a middleware application is proposed to manage energy-intensive requests from mobile devices and serve them in the most efficient way possible, effectively increasing battery life almost three times, as illustrated in Figure 3. This is achieved by providing more energy-efficient bulk data transmission and adaptive GPS receiver reading to minimize energy consumption without affecting perceived performance [4]. Similar to adaptive algorithms, adaptive scheduling and configuration also builds on the results of measuring and profiling of applications for energy usage. However, by contrast to adaptive algorithms, the adaptive scheduling and configuration approach works on a different level, external to the applications themselves. For example, a CPU scheduling approach uses online profiling combined with dynamic voltage scaling to decide when and for how long to execute energy-intensive applications [6]. The proposed approach is essentially a soft real-time CPU scheduler called GRACE-OS, which uses known distributions of application cycle demands of different multimedia codecs, in order to schedule processes so that CPU idle time is minimized and CPU busy time in lower-power speeds is maximized [1]. The results of an empirical evaluation using an HP laptop and several multimedia codecs showed energy savings of up to 72% over more traditional scheduling approaches [11]. A different adaptive approach uses an overseer system to adapt applications to mobile environments and to tune them for optimal energy-use through their interfaces, without changes to the code itself [5]. The overseer system, called “Puppeteer”, assumes that applications export well-defined interfaces through which their behavior can be modified in order to save energy [12]. In addition, “Puppeteer” also acts as an intermediate layer between applications and servers, controlling data exchanges so that they are more energy efficient. For example, “Puppeteer” can provide a PowerPoint application with distilled versions of remote presentations, purposefully excluding from the presentation images that can increase significantly the amount of energy used for document transfer [5].

On the operating system layer, approaches to energy saving in mobile devices can be broadly grouped in two classes: centralized and decentralized approaches. Centralized approaches consider energy as a system-wide resource that is to be managed globally for optimal results. As a result, centralized approaches usually propose a framework or global model of energy consumption that is used to schedule application processes and manage CPU allocation for best results overall [10] [11]. By contrast, decentralized approaches tend to focus on tuning or otherwise improving the energy efficiency of separate, individual applications or activities, such as wireless network management [12]. A survey of existing Android power saving apps in Google store found two main decentralized approaches for energy saving at the operating system layer: CPU frequency scaling and controlling smartphone features such as brightness levels, Wi-Fi, Bluetooth or GPS (turning it off for reduced energy consumption) [13]. Both approaches essentially propose to reduce energy consumption by limiting either performance or available features. Moreover, this survey notes that the approaches have several significant limitations, including fixed profiles that do not take into account individual users’ needs or use patterns, potential security risks due to root access on the phone, and the static nature of approaches which do not evolve to adapt to use contexts [13]. Most of the above limitations are addressed by a different decentralized approach that reportedly reduced power consumption [12]. This is achieved through a combination of flexible self-tuning power management for wireless networks based on intent of applications (whether they transfer data and what maximum delay they are willing to tolerate), observed access patterns, energy usage of the platform and characteristics of the physical network interface [12]. This approach is motivated by the observation that existing power management schemes such as used in 802.11b are ineffective and can sometimes result in significant degradation of performance due to repeated delays introduced by turning off the network interface [9].

On the network layer, approaches for reducing energy consumption focus mainly on ensuring predictable delivery of data packets [14] and supporting computation off-loading [6] or protocol design for lowering energy consumption [11]. Some of the approaches on network layer provide concrete energy-saving techniques, while other approaches serve as pre-requisites, by providing reliable estimates of energy consumption for various tasks in order to enable more energy-efficient task scheduling and execution. Predictable delivery of data packets focuses primarily on ensuring that data packets are received by mobile devices in bursts, so that the network interface card can be transitioned to low-energy consumption sleep mode between those bursts. One approach uses a transparent proxy between mobile devices and remote locations, as a means to control traffic and ensure predictable delivery. The proxy buffers data to create bursts of packets, dynamically generates optimal transmission schedules and maintains separate connections to the client and the server [8]. Energy savings through this approach are reported to be within 10-15% of the optimal case, without any significant loss of data transmission reliability [8].

The first approach focuses specifically on reducing the power consumption of LCD displays, while maintaining a similar quality of the display. This is done through several techniques that target various energy-intensive features of a LCD display, such as refresh, color depth control and luminance of the display. The energy consumed by those features is reduced through adaptable techniques such as variable duty-ratio refresh, dynamic color depth control and backlight luminance dimming [2]. Each of those techniques is shown to provide significant reductions in power consumption of the device, with their aggregated result reportedly extending battery life between 20% and 38%, depending on the type of application used [2]. The duty-ratio refresh technique refers to reducing refresh rates of LCD displays as a means to reduce energy consumption due to both frame buffer and LCD panel bus, which are considered to be responsible together for approximately 20% of energy use overall. Empirical tests show that the refresh rates can be reduced up to 50% without any flicker of the image becoming noticeable. This new organization allows more energy-efficient decreasing of color depth through the shutting down of the LCD when only 8-bit depth is required. Empirical results show that the energy saved depends however on the applications that are running, since savings are higher for videos and image-intensive applications and smaller for text viewers.

By contrast to the above approach, the Currency framework has a broader scope, as it does not focus solely on one energy-intensive component of a device. Instead, the Currentcy framework is explicitly designed to manage energy use at the level of the entire system. More precisely, energy is managed in the Currency framework based on energy goals defined for the entire system (such as desired active time until battery depletion) and a unified approach to resource allocation across applications and components, which is defined through an overall allocation policy, a task-level allocation policy and Currentcy accounting schemes [2]. Essentially, in the Currentcy framework, energy is modeled as a currency and the approach to resource allocation defines the way in which the price of various resources is calculated in this energy-currency. When considering both the above approaches, there are a few main differences with respect to scope and goals. While the first approach focuses on only one component of a device (although seen as part of the system), the second approach focuses on the system as a whole. Moreover, while the first approach aims directly to increase battery life without significant degradation of the quality of service, the second approach has the much more flexible goal to guarantee a desired battery life with as little degradation of quality of service as possible

Another limitation of existing approaches is that they tend to be device centric, meaning that they focus explicitly on the energy concerns that can be observed on one specific device such as the network adapter. For example, several approaches propose to lower energy consumption of wireless network adapters by ensuring predictable delivery of data packets [8] [10]. However, such device centric approaches fail to consider wider issues that can be observed only at a system level. Such issues include for instance network congestion and mobility requirements of the user, which can both affect energy consumption and the type of adaptations that are acceptable. The device centric characteristic of existing approaches also means that most adaptations are essentially reactive: changes are made in response to observed inefficiencies in energy use, but there is little done to prevent energy issues in the first place. For example, several approaches on the user-application layer rely on measurements and profiling of applications to decide on the most efficient scheduling and configuration [2] [8] [9]. However, this approach does not offer any opportunity for preventing inefficient energy use by various applications.

Overall, it can be argued that the main limitation of existing approaches is the fact that they lack a generalized global framework that would enable co-ordination between different system layers, a system-centric rather than device-centric approach to energy management and a pro-active adaptation to different situations. This is partially due to the fact that most approaches focus on providing very specific improvements, such as more efficient functioning of the wireless network adapter or more efficient scheduling and configuration of applications [5] [8] [9]. However, despite the benefits of those specific improvements, they cannot deliver their full benefits unless they are integrated together in a generalized global framework that considers every relevant aspect of a given system.

2. Ontology-Based Middleware for Cross-layer power consumption knowledge exchange

The increasing popularity of ubiquitous, interconnected computing devices such as laptops, PDAs, and 3G mobile phones is fostering the emergence of environments where people access their personal information, corporate data and public resources “anytime” and “anywhere”. Context, an important concept that can be any information regarding the situation and environment, is often exploited by human beings for communications and actions. Context-aware services have been proposed in the computing field to adapt system behaviors based on the retrieved context data. Although different context-aware systems have been presented in literature, this paper moves one step further by building an adaptive cross-layers power management system with context-awareness using ontology-based models. The proposed system can adapt its behaviors according to the changes of various context data. First, it chooses the appropriate power saving mode based on the retrieved static context data such as user’s profile, location, time, and usage forecast. Then, dynamic adaption is performed to greatly improve power saving by adapting to various operational data such as varying wireless channel quality, available energy of the end equipment, network congestion and application Quality of Services (QoS). To verify the effectiveness of proposed system, a test bench and its experimental results are also described.

The proposed solution consists of designing ontology to be used as a power-cognizant distributed middleware. This is intended to provide the following main benefits:

1. Allowing dynamic adaptation to global changes in the system or in the environment.
2. Enabling co-ordination of energy-aware adaptations at all system levels.

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 129-140
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

3. Maximizing the utility of a low-power device, based on a flexible definition of this utility (such as desired power savings, battery time or application quality of service).
4. Allow the evaluation and comparison of cross layer adaption techniques that focus on different definitions of utility of a mobile handheld device. For instance, the framework should allow the evaluation and comparison of approaches that focus on performance vs. those that focus on quality of service vs. those that focus on power tradeoffs.

Protégé is a free, open source ontology editor and a knowledge acquisition system [5]. Protégé provides a graphic user interface to define ontologies. It also includes deductive classifiers to validate that models are consistent and to infer new information based on the analysis of ontology. Like Eclipse, Protégé is a framework for which various other projects suggest plugins. This application is written in Java and heavily uses Swing to create the rather complex user interface. In Protégé 7.0, it is possible to manually construct class hierarchy called the asserted hierarchy. The class hierarchy that is automatically computed by the reasoner is called the inferred hierarchy. Moreover, this tool automatically classifies the ontology (and check for inconsistencies). The already mentioned reasons certify the high-quality of the tool. Protégé 7.0, OWL was chosen in order to make the descriptions of concepts, attributes, and instances formal, so as the knowledge can be machine-readable and reasoning-automated.

Figure.1 shows the hierarchal graph of the proposed cross-layer power consumption knowledge exchange ontology. The blue dotted lines show both the research work implemented and contributions added by this paper to on each layer of the smartphone. As figure.1 shows the first layer of the smartphone which is “Users” and its subcategories, this new layer was presented and its subcategories were proposed in paper 4 of this paper. The second layer shows the “Applications” and the Google Play classifications of applications plus the OS. These categories were rated and studied on both papers 3 & 5 of this paper. The fourth layer shows the “Network Connectivity” where a new concept of “lazy algorithm was proposed on paper 6 of this paper. Finally the fifth “Hardware Layer” which was deeply studies and rated as separate components in 3 of this.

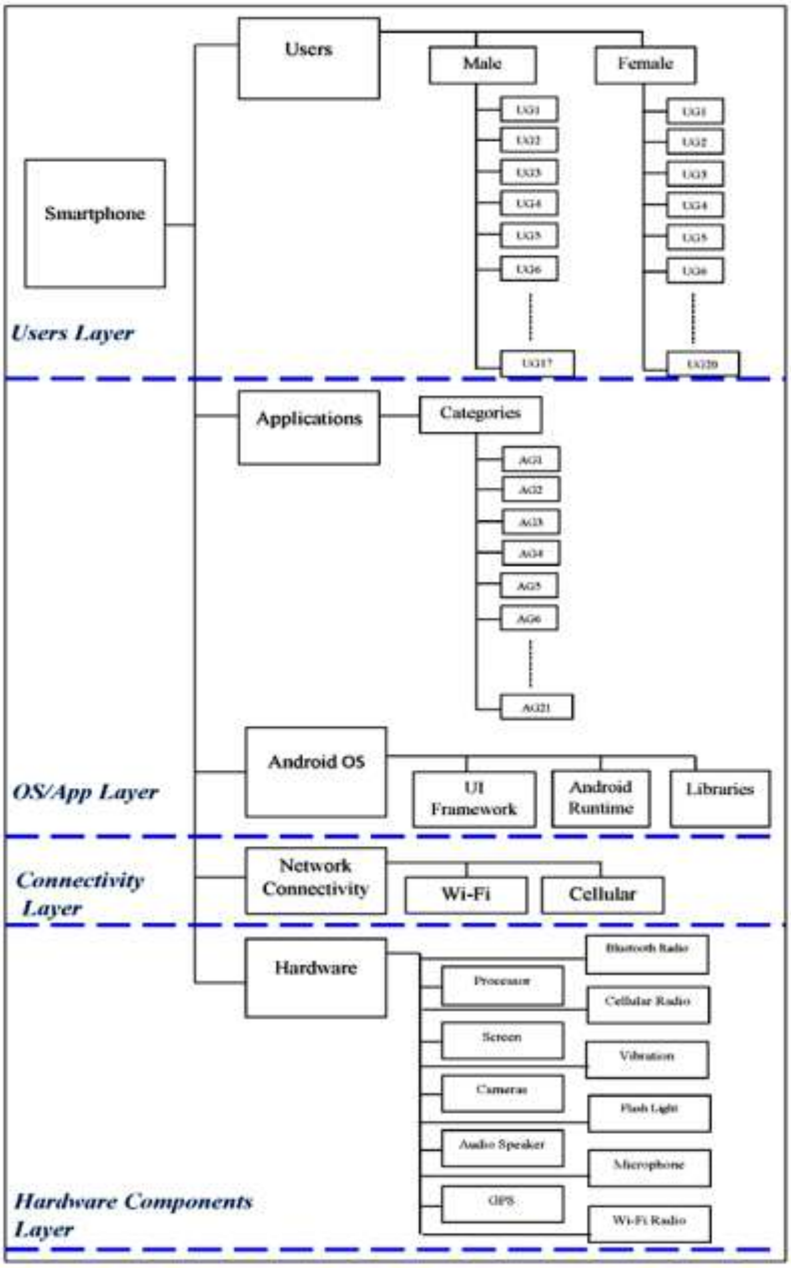
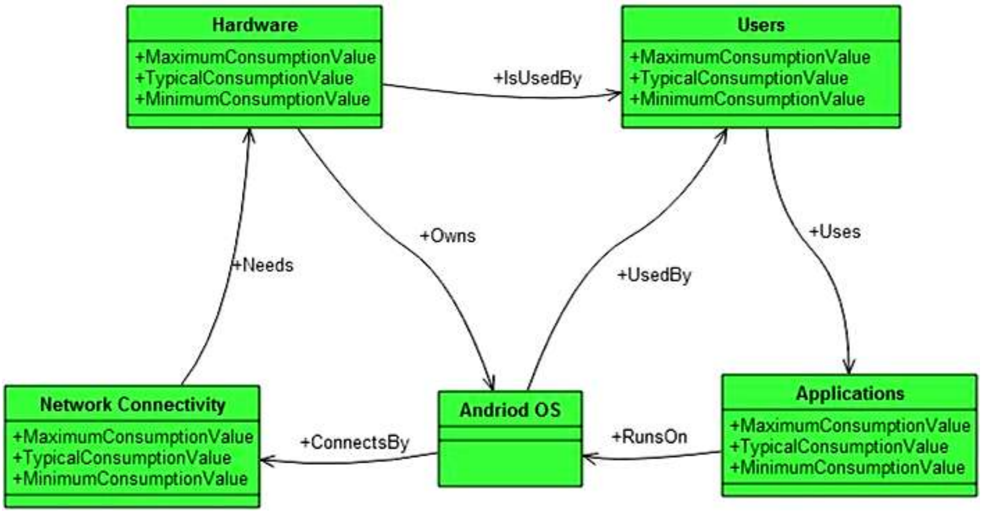


Figure. 1. Hierarchal graph of the proposed cross-layer power consumption knowledge exchange ontology

The attributions and relations of mobile phone can be seen on figure. 2 as it also shows the high-level UML diagram of the proposed cross-layer power consumption knowledge exchange ontology:



Attributes:

- MaximumConsumptionValue
- TypicalConsumptionValue
- MinimumConsumptionValue

Relationships:

- ”IsUsedBy” means the relationship between hardware and users,
- “Needs” means the relationship between hardware and network connectivity,
- “Owns” means the relationship between hardware and android OS
- “ConnectsBy” means the relationship between network connectivity and android OS
- “UsedBy” means the relationship between Users and Android OS
- “Uses” means the relationship between Applications and Users
- “RunsOn” means the relationship between Applications and Android OS

Figure. 2 . High-level UML diagram of the proposed cross-layer power consumption knowledge exchange ontology

The individuals of the proposed cross-layer power consumption knowledge exchange ontology are shown in details in the following diagrams where each layer of the high-level diagram is presented separately for easier reading and demonstration proposes.

The following graphs show the individuals of the proposed cross-layer power consumption knowledge exchange ontology for “Users” layer: Next the above proposed ontology is to be put into verifying and validation process which will be done by running a simple query. This will help to verify the output of the query and make sure about the flow of knowledge among the knowledge sequence that was proposed. In order to achieve this an OWL editor is needed which is protégé [6]. The first step is to define the classes, subclasses and attributes in protégé as illustrated in Figure.3.

Figure.3 shows each layer of the smartphone as a class that will be divided into subclasses, it shows each of them separately for easier reading and demonstration proposes.

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 129-140

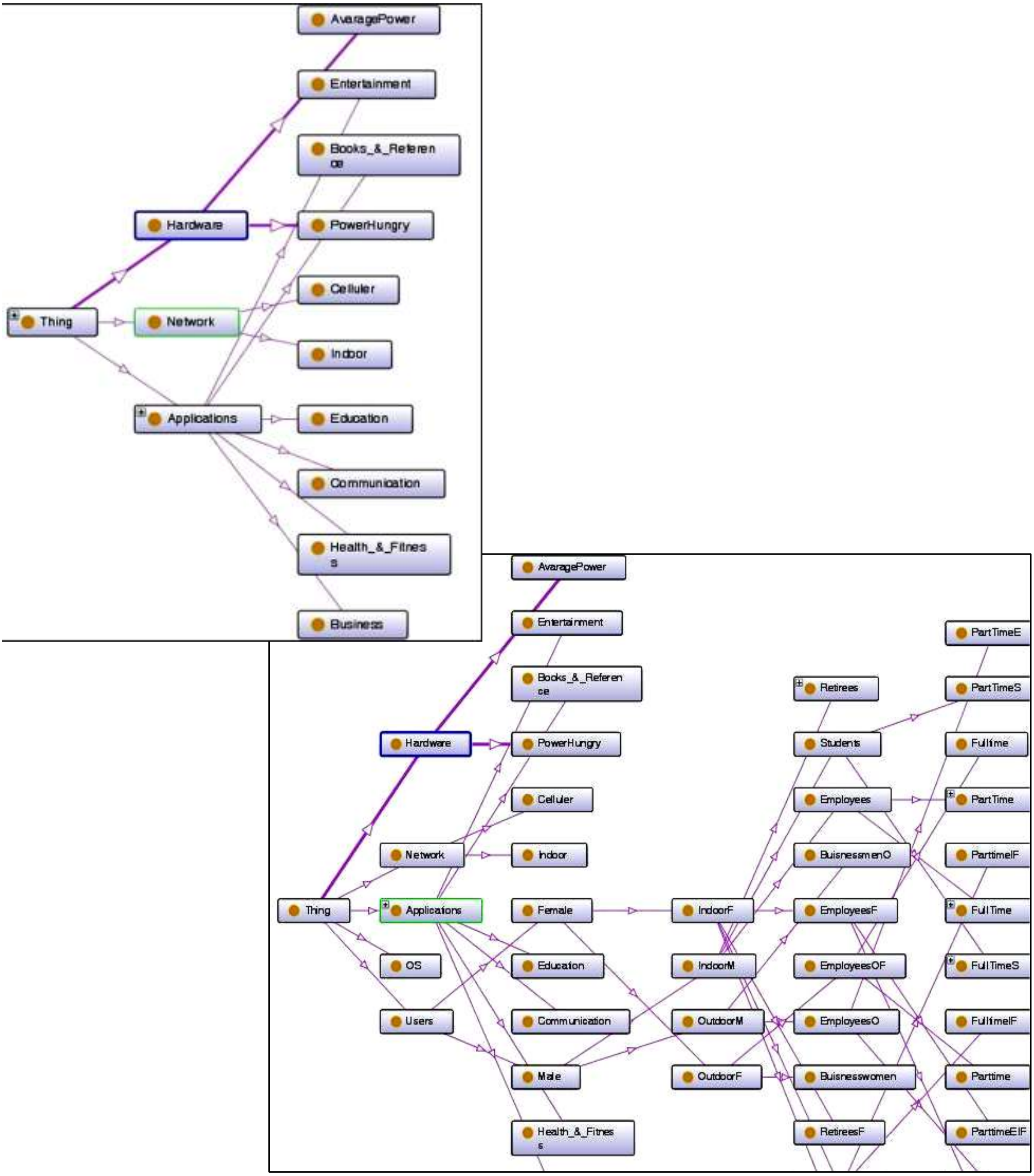


Figure. 3 . UML diagram of the full proposed cross-layer power consumption knowledge exchange ontology after all layers where joined by the previously mentioned relationship.

The previous step is repeated on all classes and subclasses until all objects are defined. After that many helpful features can be used from protégé including generating the code of the designed ontology and also generating the graphs of each class. The following figures show the ontology for each class generated by protégé. The purpose of showing these figures is to demonstrate the hierarchy of the ontology and to have a wide vision about the full proposed solution. The following will show each layer of the smartphone as a class that will be divided into subclasses, and the below figures will show each of them separately for easier reading and demonstration proposes.

Finally an OWL code was generated from the above figure by using the same ontology editor protégé. The code can be executed as a full functioning solution which will help all the layers of the smartphone to exchange information related to power consumption management. For example now the applications can read the estimated power consumption value of the current smartphone user. Also the hardware components can know the estimated power consumption amount for the installed application. It is recommended that the location of this solution among the different layers of smartphones would be on a memory firmware location. The reason behind this is because of avoiding all phases of detective strategies. The solution will use historical data that gets updated whenever the user updates the smartphone firmware. The generated code will also provide help during the steps of validating and verifying the ontology.

A simple SPARQL [6] query is executed for the phone ontology base by the java programming language in the MyEclipse7.1 development environment. Assuming a smartphone layer queries some parameters or properties of a smartphone layer from the ontology, the layer only needs to send the pattern of the phone as an input for getting the result. For different smartphones layers, different interfaces are given. The key code of SPARQL is as follows. Finally an OWL code was generated and was verified by a simple SPARQL[6] query by java programming language in the MyEclipse7.1 development environment.

3. Scenarios of Applying the Proposed Middleware

3.1 Unifying techniques for switching between network types

When dealing with power consumption, mobile devices cannot disregard environment conditions and user habits. Both internal and external conditions are rapidly changing and may influence the response of the entire system, e.g., switching between network types may cause unpredictable power consumption. In order to puzzle out this issue, we can use the proposed ontology to share knowledge of power data of different network types (e.g. WiFi, 3G, LTE, 4G, Bluetooth) through the proposed middleware. We can then make informed decisions of network switching based on both the internal and external conditions. The proposed ontology middleware can enable all smartphones network types to exchange energy saving adaptations knowledge which will improve the power saving strategies of the smartphones' networking in general. The proposed ontology model will allow monitoring and flexible controlling of the various network types of a smartphone power consumption habits and take switching decisions at the right times.

3.2 Real-Time Location-Aware Smartphone Apps Power Requirements

In terms of processors and memory, these wireless mobile devices are as powerful as the PCs were one decade ago. Therefore, they are perfectly suitable to become the first real-life platforms for ubiquitous computing. For instance, they can be programmed to run location-aware applications that provide people with real-time information relevant to their current places. Deploying such applications in our daily life, however, requires a good understanding of their power requirements in order to ensure that mobile devices can indeed support them. The proposed ontology can provide for a quantitative analysis activity of power consumption for location-aware applications, which builds a test-bed for mobile social computing. Based on this analysis, we can extend these applications to run for longer hours, while updating the user location frequently enough in the ontology to support real-time location-aware communication.

3.3 Power Consumption during Signaling

The explosive growth in mobile communication requirement of high data demand with high QoS led to further increase in power consumption. A lot of computation is due to excess processing of signaling messages which Smartphones perform during calling and/or before call. The earlier measurement studies on mobile phones do not show the power consumption during signaling activities. The use of the proposed ontology can provide a detailed measurements and data of power consumption during various signaling events, whose increase in frequency leads to unwanted battery consumption and increased load in rest of the network. Network activities, namely, network access, call initiation, call termination, and mobility in a coverage hole can all be considered for the measurement. The proposed ontology thus will provide a comparative energy distribution for voice and signaling traffic, and consequently lead to significant impact on energy consumption with low duration of call.

3.4 Applications' Traffic Trace analysis

The explosive number of applications now available on Smartphones and the availability of high-speed mobile broadband connectivity through 4G networks such as LTE have only increased the challenges in improving battery life of these devices. Using the proposed ontology, the 4G power consumption issues through traffic traces and power measurements on LTE device over commercial LTE network can be recorded in the ontology. These traces can be analyzed to understand the traffic characteristics. The ontology will also contain measured power consumption of the various components of the device. It will thus provide the cdf's of the packet size and packet inter-arrival times of background traffic of common applications (e.g., Skype, Google Talk, Twitter etc), and power consumption values and duration of low-power states which are crucial for the innovations and implementation of efficient power optimization mechanisms in LTE networks.

3.5 Parallel Connections (e.g. TCP, 3G, VoIP, etc.) Power Consumption

In this scenario, we analyze the end-to-end communication activities of a modern mobile phone, Nokia N95, to understand how much energy different communication alternatives consume. In particular, using the proposed ontology we can investigate the interactions when multiple connections are used in parallel. Parallel connections save energy but the gains vary depending on the technology. For example from the ontology data we found that TCP downloads during 3G voice calls result into 75%-90% energy savings, TCP downloads during VoIP calls result into 30%-40% savings, and TCP downloads when other TCP streams are active at the same interface result into 0%- 20% savings. These results indicate that there is a significant potential to save energy if applications are engineered to take advantage of this phenomenon. We can use the proposed ontology to share such knowledge through the proposed middleware. We can make informed decisions of network interactions when multiple connections are used in parallel based on both the internal and external conditions.

3.6 Signal Strength, Network Bandwidth, and Power Consumption

Power consumption from wireless data communications contributes a significant part for the overall mobile phone platform power. Compared with cellular data transmission, WiFi transmission is more energy-efficient. Nevertheless, there are scenarios where cellular network is the only option to access the Internet for a number of mobile phones because of limited Wi-Fi access. The proposed ontology proposes a cooperative approach of using WiFi tethering technique to optimize the overall power consumption. Based on a series of real measurements recorded in the ontology, we can characterize the relationship among signal strength, network bandwidth and power consumption, and point out principles of choosing appropriate mobile hotspot nodes. The ontology then can suggest to optimize the overall power consumption, i.e., most suitable hotspot first (MSHF) and most suitable leaf first (MSLF). The proposed ontology results demonstrate that MSHF can acquire 24% power saving and MSLF can achieve up to 29% compared with regular 3G approach.

4. Conclusion

To the best of our knowledge, this work adds a new ontology specification "Smartphone Power Consumption Ontology" to W3C ontology repository. The developed middleware can be ported on any smartphone. More scenarios and experimentations are needed to prove the power of the new middleware. Our initial results support the analogy set between this proposal and previously successful "Smart Home Ontology".

Acknowledgment
The authors would like to recognize the support of the RIC at Prince Sultan University.

References

[1] A. Almasri, Preventive Customized Strategies for Saving Energy on Smartphones, M.Sc. Thesis, Prince Sultan University, 2015

[2] W. Yuan and K. Nahrstedt, 'Energy-Efficient Soft Real-Time CPU Scheduling for Mobile Multimedia Systems', in Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles, SOSP '03, New York, 2003

[3] S.K. Datta, C. Bonnet and N. Nikaein, 'Power Monitor v2: Novel Power Saving Android Application', in 2013 IEEE 17th International Symposium on Consumer Electronics (ISCE), 2013

[4] S. Chakraborty and D.K.Y. Yau, 'Predicting Energy Consumption of MPEG Video Playback on Handhelds', in 202 IEEE International Conference on Multimedia and Expo, 202. ICME '02. Proceedings, 2002, pp.317–2 vol.1

[5] R. Perez-Torres and C. Torres-Huitzil, 'A Power-Aware Middleware for Location Amp; Context Aware Mobile Apps with Cloud Computing Interaction', in 212 World Congress on Information and Communication Technologies (WICT), 2012

[6] J. Flinn, 'Reducing the Energy Usage of Office Applications', in Middleware 201, ed. by Rachid Guerraoui, Lecture Notes in Computer Science, 2218 (2001), pp. 252–72

[7] Yuan and Nahrstedt, ‘Energy-Efficient Soft Real-Time CPU Scheduling for Mobile Multimedia Systems’, Lecture Notes in Computer Science, 2218 (2001),pp. 149–163

[8] https://support.google.com/googleplay/answer/6014972?hl=en

[9] Abogharaf and Naik, ‘Client-Centric Data Streaming on Smartphones’, Mobile and Wireless Networking (MoWNeT), International Conference on Selected Topics, 2013

[10] W. Yuan and others, ‘Design and Evaluation of a Cross-Layer Adaptation Framework for Mobile Multimedia Systems’, 203, Electronic Imaging 2003

[11] H. Zeng ‘ECOSystem: Managing Energy As a First Class Operating System Resource’, in Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems, New York, 2002

[12] http://ziyang.eecs.umich.edu/projects/powertutor/

[13] Bonino and Corno, ‘DogOnt - Ontology Modeling for Intelligent Domotic Environments’ 2008, pp.790–803

[14] Yuan, ‘Design and Evaluation of a Cross-Layer Adaptation Framework for Mobile Multimedia Systems’ SPIE/ACM Multimedia Computing and Networking Conference (MMCN), 2003