2013-09

Energy efficient group context aware sensor management strategy for tactical operations

Graves, Samantha A.

Monterey, California: Naval Postgraduate School

http://hdl.handle.net/10945/37633

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

ENERGY EFFICIENT GROUP CONTEXT AWARE SENSOR MANAGEMENT STRATEGY FOR TACTICAL OPERATIONS

by

Samantha A. Graves

September 2013

Thesis Advisor: Gurminder Singh
Thesis Co-Advisor: John Gibson

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** September 2013	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE
ENERGY EFFICIENT GROUP CONTEXT AWARE SENSOR MANAGEMENT STRATEGY FOR TACTICAL OPERATIONS

5. FUNDING NUMBERS

6. AUTHOR(S) Samantha A. Graves

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

Shared situational awareness (SSA) is essential to success and safety in tactical environments, whether fighting wars or providing relief during disasters and humanitarian catastrophes. The increased availability of sensors in mobile devices offers groundbreaking opportunities for continuous context-aware applications that are capable of responding to the operating conditions of users and their environment. However, continuous context-aware applications involve high-energy consumption. A key challenge in tactical environments is to make the most effective use of scarce resources.

There are numerous approaches for reducing energy consumption of continuous context-aware applications. This thesis examines two methods: Sensor Substitution and Triggering (SENST) and Acquisitional Context Engine (ACE). The goal of this thesis is to explore the capabilities and limitations of SENST and ACE for group context-awareness and provide a group energy-efficient sensor management strategy that enhances the dissemination of SSA in tactical environments.

14. SUBJECT TERMS Group Context Aware Applications, Sensor Management, Continuous Context Sensing, Energy Efficiency

15. NUMBER OF PAGES 99

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

ENERGY EFFICIENT GROUP CONTEXT AWARE SENSOR MANAGEMENT STRATEGY FOR TACTICAL OPERATIONS

Samantha A. Graves
Captain, United States Marine Corps
B.S., University of South Carolina, 2007

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2013

Author: Samantha A. Graves

Approved by: Gurminder Singh
Thesis Advisor

John H. Gibson
Thesis Co-Advisor

Peter Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Shared situational awareness (SSA) is essential to success and safety in tactical environments, whether fighting wars or providing relief during disasters and humanitarian catastrophes. The increased availability of sensors in mobile devices offers groundbreaking opportunities for continuous context-aware applications that are capable of responding to the operating conditions of users and their environment. However, continuous context-aware applications involve high-energy consumption. A key challenge in tactical environments is to make the most effective use of scarce resources.

There are numerous approaches for reducing energy consumption of continuous context-aware applications. This thesis examines two methods: Sensor Substitution and Triggering (SENST) and Acquisitional Context Engine (ACE). The goal of this thesis is to explore the capabilities and limitations of SENST and ACE for group context-awareness and provide a group energy-efficient sensor management strategy that enhances the dissemination of SSA in tactical environments.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ACE	Acquisitional Context Engine
API	application programming interface
AUMF	Authorization for Use of Military Force
BFT	Blue Force Tracker
C2	command and control
CJTF-76	Combined Joint Task Force 76
CL	critical level
COP	common operational picture
CPU	central processing unit
DOD	Department of Defense
DRDC	Defence Research and Development Canada
FOB	forward operating base
GEM	Group Energy-Efficient Management
GPS	global positioning system
GUI	graphical user interface
HA/DR	humanitarian assistance/disaster relief
ICE-B	International Conference on e-Business
IED	improvised explosive device
I/O	input/output
iOS	iPhone Operating System
ISE	Information Security to the Edge
LAN	local area network
LT	less than
MVC	model-view-controller
OCHA	Office for the Coordination of Humanitarian Affairs
OEF	Operation Enduring Freedom
OODA	observe, orient, decide, and act
OODB	object-oriented database
OS	operating system
PRT	Provincial Reconstruction Team

SDK	software development kit
SENST	Sensor Substitution and Triggering
SOP	standard operating procedure
SR	stabilization and reconstruction
SSA	shared situational awareness
TEPCO	Tokyo Electric Power Company
USAID	United States Agency International Development
USGS	United States Geological Survey
UN	United Nations

ACKNOWLEDGMENTS

This thesis would not have been possible without the help of several individuals who contributed to the preparation and completion of this study. I offer my regards and blessings to all of those who supported me in any respect during the completion of this thesis.

First and foremost, I am grateful to Dr. Gurminder Singh and Mr. John Gibson, who have provided outstanding guidance and expert advice. Mr. Gibson, thank you for motivating me to pursue the network and mobile systems track. I know all the knowledge and skills I gained will be useful in my upcoming assignments. Dr. Singh, thank you for introducing me to this topic and area of work. Your passion and enthusiasm excited me. Thank you both for keeping me on track and helping me hurdle all obstacles.

I would like to thank several excellent professors at Naval Postgraduate School: Dr. Michael Atkinson, Dr. Mikhail Auguston, Mr. Arijit Das, Dr. George Dinolt, Mr. William Fast, Dr. Ralucca Gera, Dr. Theodore Huffmire, Dr. Gilbert Lundy, Dr. Craig Martell, Mr. Loren Peitso, Dr. Neil Rowe, CDR Alan Shaffer, Dr. Dennis Volpano, Mr. Daniel Warren, and Lt Col Joel Young (Ret.). Thank you for making this experience so rewarding. The studies from my numerous classes contributed to the completion of this thesis.

I owe my deepest gratitude to my husband, Rodney, and children, Rodney II and Samiya. Thank you for your continued encouragement, understanding, and patience over the past two years. I love you.

Last but not least, thank you dear God, for answering my prayers and giving me strength to carry on when I felt defeated and overcome.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. MOTIVATION

War is a violent struggle between opposing forces to attain some ideological, religious, or political objective. It is a place characterized by fear and chaos, which intensifies uncertainty and fuels unpredictable behavior. Carl Von Clausewitz, a Prussian military analyst, invented the term "Fog of War."[1] It is used to describe uncertainty in situational awareness. "Fog of War" can relate to uncertainty about the enemy (forces or objectives) or the environment (weather or terrain).[2] This uncertainty is compounded by the fact that military intelligence gathered is almost always incomplete.

The challenge of Command and Control (C2) is getting relevant information when and where it is needed. The C2 process is one of acquiring information, planning a course of action, and then implementing that planned course of action. This can be explained using Colonel John Boyd's concept of Observe, Orient, Decide, and Act (OODA) loop. (See Figure 1.)

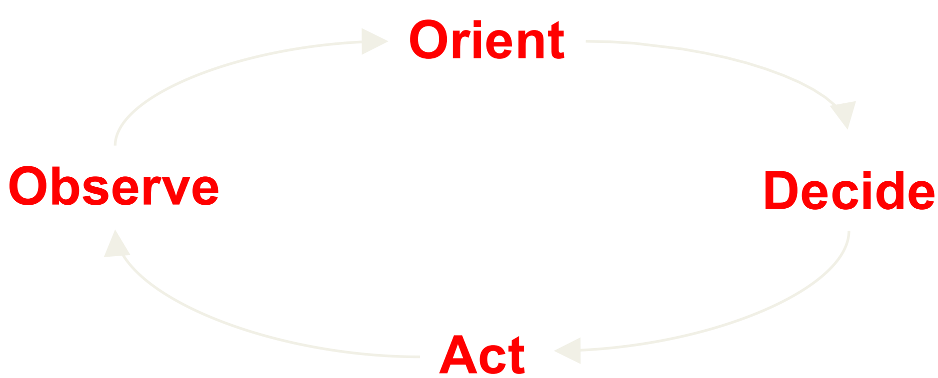


Figure 1. OODA Loop

[1] Carl V. Clausewitz, *On War* (Princeton, NJ: Princeton University Press, 1976).

[2] Ibid.

The warfighter who can process the OODA loop cycle more quickly than the opponent gains the advantage. By developing a system that quickly gathers or observes information and then processes or orients the information, decisions can be made and acted upon faster. That is to say, the OODA loop becomes smaller. The reason bad decisions are made is often not because the warfighter is a bad decision maker, but because the warfighter fails to place the gathered information into its proper context. Lifting the "Fog of War" is a critical component to military strategy and ensures warfighters can make effective, timely decisions that positively shape the battlefield.

Every warfighter possesses elements of the common operational picture (COP), but only by placing these elements together can a complete picture be formed; typically, military forces are deployed in teams and operate in dynamic environments that are semi-structured. In today's military, warfighters are using hand-held mobile devices in tactical environments configured with specific applications and information to help meet mission requirements. A mobile device's onboard sensors can be used to gather information for analysis of the warfighter and the warfighter's situation. The continuous sensing of the mobile device's sensors represents the main source of real-time data, which can help lift the "Fog of War".

Currently, most context-aware applications focus on the individual warfighter. Yet, military forces, which deploy in teams, must communicate and coordinate to complete tasks and accomplish the mission. Group context-aware applications allow the integration of relative, contextual information from the commander, warfighter, and nearby team members to support a team executing a mission.[3] The power of group context-aware applications is that it improves data collection and synergistically enhances the comprehension of the environment.

[3] Grace Lewis, Marc Novakouski, and Enrique Sánchez, "A Reference Architecture for Group-Context-Aware Mobile Applications," *Mobile Computing, Applications, and Services 2012* LNIST 110 (2013): 44–63.

Although group context-aware applications enhance shared situational awareness (SSA), applications use sensors that have a strong impact on the overall device power consumption. Warfighters need a wide range of sensors to gather the most complete and relevant information in which to develop the COP. To capture and analyze the warfighter's state and environment requires operating all or most existing sensors in the mobile device continuously. This continuous running of sensors drains the device battery quickly. A key challenge in military operations is to maximize the use of its limited resources. Therefore, it is imperative to construct a framework to utilize sensors efficiently while capturing information to consume less power. Sensor management concerns the control and coordination of sensing resources in order to optimize the energy consumption of sensors while gathering the most accurate, real-time information.[4] As such, sensor management is an essential strategy in the development of energy-efficient group context-aware applications that are key enablers of SSA.

B. BACKGROUND

Warfighters depend on timely and accurate information to perform tasks and, more importantly, stay alive. Unfortunately, the "Fog of War" often obscures valuable information. Technological advances offer new opportunities to warfighters operating in tactical environments, such as handheld mobile devices with integrated sensors that provide accurate, real-time contextual information. The development of group context-aware mobile applications makes it possible for warfighters to attain and maintain SSA.

While advantages of continuous sensing are evident, high-energy consumption of context-aware applications is a key challenge. Considerable

---

[4] A. Benaskeur and H. Irandoust, *Sensor Management for Tactical Surveillance Operations*, Technical Memorandum 2006-767 (Valcartier, Canada: Defence Research and Development Canada (DRDC), 2006).

research has been conducted to address this challenge, and this has led to the development of sensor management strategies that reduce the sensing high-energy costs without sacrificing accuracy. Sensor substitution and triggering (SENST) is a dual approach that is based on: 1) sensor substitution, which is prompted when observations made from lower resource consumption sensors can be used to obtain information and 2) sensor triggering, which measures environmental changes through sensors with a low energy consumption and uses energy-intensive methods only if changes imply an update.[5] An additional approach for sensor management is Acquisitional Context Engine (ACE). ACE is a middleware that supports continuous context-aware applications while mitigating sensing costs by inferring contexts; it provides the user's context to running applications and dynamically learns relationships among context attributes.[6]

Implementing group context-aware mobile applications provides the war fighting team with the capability to quickly sense emerging group context and form a more accurate COP. Although context-aware applications are high-energy consumers, smart sensor management strategies can reduce power consumption while providing warfighters accurate, real-time data.

C. PURPOSE

This thesis serves as a basis for understanding the value of group context-aware applications for military forces operating in tactical environments. It explores the context-awareness for individuals and groups and demonstrates how group context-awareness can help develop a COP. It examines two methods for sensor management, SENST and ACE, in detail and discusses the

5 Maximilian Schirmer and Hagen Höpfner, "SENST: Approaches for Reducing the Consumption of Smartphone-Based Context Recognition," *CONTEXT 2011* LNAI 6967 (2011): 250–263.

6 Suman Nath, "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing," *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

advantages and disadvantages of each method. Finally, it provides a group energy-efficient management strategy, which fully leverages SSA.

D. ORGANIZATION

The following chapters in this thesis are arranged and discussed as follows:

Chapter II (Context-Awareness) introduces the concept of context-awareness and explores individual and group context-awareness as they relate to group context-aware mobile applications. Two case studies are reviewed to demonstrate the value of group context-aware applications in tactical environments. It also discusses a reference architecture, which leverages group context-aware applications in tactical environments.

Chapter III (Energy-Efficient Sensor Management Strategies) examines two sensor management strategies, SENST and ACE, and describes the advantages and disadvantages of each as related to military forces in tactical environments.

Chapter IV (Group Energy-Efficient Management Framework and Implementation) presents a conceptual framework that aims to assist in the development of energy-efficient group context-aware applications.

Finally, Chapter V (Conclusions and Future Work) provides an overall summary of what group context-aware applications provide to military forces in tactical environments and how such applications can be implemented at reduced power consumption. It also presents suggestions on how sensor management can be improved upon and highlights future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. CONTEXT-AWARENESS

Modern mobile devices are equipped with a multitude of pre-installed sensors, which detect user context and provide data about activities and events in the real world. Sensors provide mobile devices with the ability to monitor and learn about the user. To interpret these data, context-aware applications are engaged. This chapter explores context-awareness for individuals as well as groups and demonstrates the utility of group context-aware applications in tactical environments.

A. INDIVIDUAL CONTEXT-AWARENESS

1. Context

Context has no strict definition and its interpretation is often associated with its application. This research defines *context* using Anind Kumar Dey's definition:

> *Context* is 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 applications themselves.[7]

Following this definition, there is myriad information that can describe a situation. Dey identifies four primary types of context information that are most important to characterize the situation of an entity; these are location, identity, time, and activity.[8] The context interpreted from these elements can also be used to obtain other sources of contextual information. For example, if an individual's identity is known, then that individual's date of birth or email address can be derived.[9]

---

[7] Anid K. Dey, "Understanding and Using Context," *Personal and Ubiquitous Computing* 5 (2001): 4–7.

[8] A. Benaskeur and H. Irandoust, *Sensor Management for Tactical Surveillance Operations*, 1–3.

[9] Dey, "Understanding and Using Context," 4–7.

Warfighters operating in dynamic settings require context[10] from the physical and electronic environments. Location, date and time, individual activities, team member interaction, resources, and enemy movement all provide insight into a warfighter's situation at a specific moment. The key objective in generating military context is to accurately sense and interpret context to provide useful intelligence.

2. Context-Awareness

Considering individual context is an important issue in today's military as the use of mobile devices has increased in tactical environments. Using Dey's definition for context-aware systems:

> A system is *context-aware* if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task.[11]

Dey identifies three categories of features that a context-aware application can support: 1) *presentation* of information and services to a user, 2) *automatic execution* of a service for a user, and 3) *tagging* of context to information to support later retrieval.[12] (See Figure 2.)

Context-aware computing has produced a number of application prototypes, frameworks, middleware systems and models for describing context. The goal of context-aware computing is to allow applications to perform seamless adaptations when pre-defined context-based situations occur making user's lives easier.

---

[10] Military context refers to any information, which characterizes a situation between military team members to derive a common operational picture.

[11] Dey, "Understanding and Using Context," 4–7.

[12] Ibid.

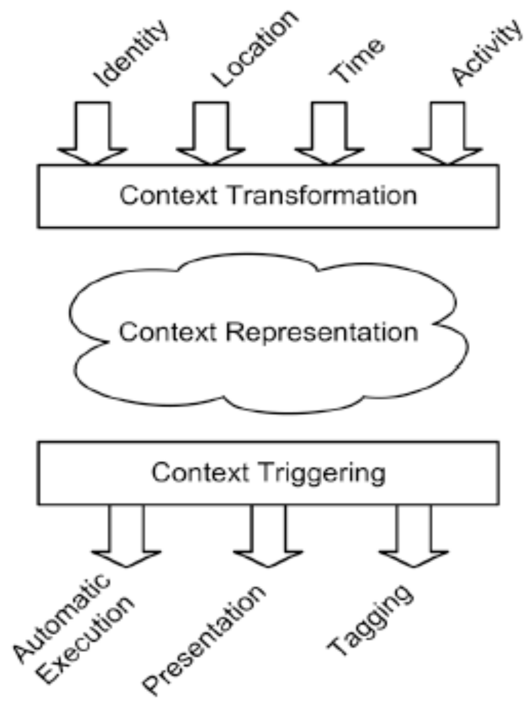


Figure 2. Layers of a Context-aware System[13]

3. Mobile Context-Awareness

Much of the research relating to context-aware mobile computing belongs to the field of ubiquitous computing.[14] It aims to enable mobile devices to provide better services for users by applying available contextual information. It enhances computer use by making multiple computers available throughout the physical environment. The use of mobile computing devices by warfighters is becoming more and more prominent in tactical environments. A context-aware mobile application uses contextual information to modify its behavior, adapt its user interface, or filter data accordingly.[15]

---

[13] Source: Alois Ferscha, Clemens Holzmann, and Stefan Oppl, “Context Awareness for Group Interaction Support” (paper presented at the annual meeting for the International Workshop on Mobility Management and Wireless Access Protocols, Philadelphia, PA, October 1, 2004).

[14] Mark Weiser, “The Computer for the Twenty-first Century,” *Scientific American* 265 (1991): 94−104.

[15] Lewis, Novakouski, and Sánchez, “A Reference Architecture,” 44−63.

Some context-aware applications need to monitor user context continuously or frequently. The high performance, mobility, and power of mobile devices allow warfighters to collect valuable intelligence directly. By utilizing a rapidly growing set of embedded sensors such as microphone, digital compass, global positioning system (GPS), gyroscope, camera, ambient light sensor, accelerometer, proximity sensor, barometer, air quality, chemicals, and radiation it is possible to understand the warfighter's situation. Today's mobile devices are equipped with an increasing range of sensing, computational, storage and communication resources enabling continuous sensing applications.

B. GROUP CONTEXT-AWARENESS

Individual context provides inadequate intelligence, representing only a partial view of the overall operational environment. Group context-awareness first considers individual context and then relates that information to the group context. Awareness of individual and group activities is essential to successful collaboration and interaction of military members operating in tactical environments. Group context-aware applications must have an awareness and understanding of all group members' activities. This awareness evaluates individual actions with respect to the tasks and goals of the group. Then information related to these factors is shared among group members.

An issue for warfighters using group context-awareness applications is that military context is dynamic. Warfighters can maneuver rapidly throughout an environment and require information at anytime and anywhere.[16] Group members must be able to unfailingly communicate with all other team members. Individuals must be capable of reliably passing relevant information while receiving context information about the group's current situation. Thus, continuous sensing is essential for warfighters operating in tactical environments. Warfighters are often deployed with limited resources for an undetermined

[16] Ibid.

amount of time. A key challenge of continuous sensing applications is to manage those resources efficiently and effectively.

Although group context-aware mobile applications have not been widely explored, such applications may provide versatile solutions for meeting military intelligence requirements. With improved processes of collecting correlations and patterns of large scale sensing data sets from mobile applications and the correct machine learning techniques to analyze the data, it is possible to discover patterns and details about individuals as well as groups. Thereby helping warfighters understand their own state as well as the state of the team.[17]

C. GROUP CONTEXT-AWARE APPLICATIONS IN TACTICAL EVNVIRONMENTS

Shared situational awareness is vital in decision making for the support of dispersed groups operating in tactical environments. Warfighters need access to real-time, accurate information. Success and failure of providing critical information can influence the outcome of joint operations. The military is becoming increasingly involved in small-scale engagements throughout the world. When providing relief, support, and assistance to a natural or manmade humanitarian disaster, warfighters must act quickly to minimize the impact of such disasters on health, sanitation, shelter, and basic needs. Group context-aware applications can support group communication in a multitude of situations and locations. The cases of military involvement in humanitarian assistance/disaster relief (HA/DR) and combat operations help to illustrate the utility of group context-awareness.

1. Case Study #1: Humanitarian Assistance / Disaster Relief

On March 11, 2011, Japan experienced a devastating chain of events that led to a national state of emergency. The magnitude 9.0 Tohoku Earthquake, the fourth largest in the world since 1900, struck near the northeast coast of Honshu,

---

[17] Ibid.

Japan.[18] The effects of the calamity multiplied as the earthquake triggered a tsunami. Within 45 minutes of the earthquake, Japan was hit with a series of tsunami waves that reached 15 meters high. The waves destroyed everything in their path and reached as far as six kilometers inland. Entire towns were washed away. The combined earthquake and tsunami caused a major nuclear energy crisis at the Fukushima Daiichi Nuclear Power Station when the damaged facility suffered "cooling system failures, fires, and explosions" that led to a large quantity of radioactive material being released.[19]

The tsunami impact reached from Chiba to Aomori and hit Iwate, Miyagi, and Fukushima the hardest.[20] Vital infrastructure was damaged or destroyed; evacuation centers lacked food, water, medicine, medical staff, and heating; and almost one million people were without power.[21] Fatalities surpassed 15,000 and the destruction and damage caused displacement of approximately 130,000 people. (See Table 1.) During a televised news conference, Japan's Prime Minister Naoto Kan stated, "in the 65 years after the end of World War II, this is the toughest and most difficult crisis for Japan."[22] The devastation propelled an international humanitarian aid effort when immediately following the disaster, the Japanese government requested humanitarian assistance from the United Nations (UN).

---

[18] United States Geological Survey (USGS), "USGS Updates Magnitude of Japan's 2011 Tohoku Earthquake to 9.0," *USGS,* March 14, 2011, http://www.usgs.gov/newsroom/article.asp?ID=2727&from=rss_home.

[19] Karren Parrish, "U.S. Leaders Recall Japan Disasters, Relief Efforts," *American Forces Press Service*, March 11, 2012, http://www.army.mil/article/75490/U_S__leaders_recall_Japan_disasters__relief_efforts/.

[20] United States Geological Survey (USGS), "Magnitude 9.0 - Near the East Coast of Honshu, Japan," *USGS,* March 11, 2011, http://earthquake.usgs.gov/earthquakes/eqinthenews/2011/usc0001xgp/#summary.

[21] United Nations Office for the Coordination of Humanitarian Affairs (OCHA*), Japan Earthquake & Tsunami*, Situation Report No. 7 (United Nations OCHA, Asia Pacific: OCHA, 2011).

[22] CNN wire staff, "Anxiety in Japan Grows as Death Toll Steadily Climbs," CNN World, March 14, 2011, http://www.cnn.com/2011/WORLD/asiapcf/03/13/japan.quake/index.html.

Japan 2011 Earthquake & Tsunami Statistics

Fatalities	***20,896***
Missing	***4,647***
Injured	***5,314***
Displaced	***130,927***
Buildings	***332,395***
Roads	***2,126***
Bridges	***56***
Railways	***26***

Table 1. Japan 2011 Earthquake & Tsunami Statistics [23]

a. Operation Tomodachi

The first several hours after a major disaster constitute a period of confusion and uncertainty. Limited situational awareness is available to determine what resources are needed and "the Fog" is compounded by continuously changing conditions. Logistics is crucial; as responders are challenged with getting necessary resources to the locations they are needed as quickly as possible.

[23] Data taken from Source: USGS, "Honshu, Japan."

The United States has almost 40,000 military troops composed of all services (Army, Navy, Marine Corps, and Air Force) stationed in Japan. These troops regularly participate with the Japanese Self Defense Force[24] (SDF) in various exercises, including humanitarian assistance and disaster relief.[25] The years of joint training allowed allied forces to coordinate efforts almost immediately to reinforce Japanese responders, and United States forces were deployed and providing critical support within 24 hours of the crisis.[26]

The United States' relief effort was designated as Operation Tomodachi. Efforts focused on "transport of relief supplies, SDF personnel, and equipment; surveillance of the affected area to search for stranded victims; and restoration of critical infrastructure, such as damaged airfields."[27] As the nuclear crisis intensified, United States' efforts included "on-the-ground expertise, decontamination of assets, monitoring of contamination of food and water, aerial detection capability, high-pressure water pumps, fire trucks, and protective gear from radioactivity."[28] Ground activities were focused on assisting the disaster victims by working with locals to clear key transport points, establishing relief hubs, restoring access roads, and delivering humanitarian supplies to refugee centers.

Urban Search and Rescue (USAR) teams trained in search, rescue, and medical services were deployed. Each team brought a great deal of search and rescue equipment, to include canines, and was prepared to operate in any type of construction or building to locate victims embedded in debris. Trained specialists "assess[ed] damage, determine[d] needs, provide[d] feedback

---

[24] Note: Although Japan is prohibited from maintaining military forces, the Japanese Self Defense Force is an all-volunteer force that helps preserve peace, public order, and Japan's independence and safety.

[25] Andrew Feickert and Emma Chanlett-Avery, *Japan 2011 Earthquake: U.S. Department of Defense (DOD) Response*, CRS Report R41690 (Washington, DC: Library of Congress, Research Service, June 2, 2011).

[26] Ibid.

[27] Ibid.

[28] Ibid.

to local officials, and conduct[ed] hazardous material surveys and evaluations of affected areas."[29]

As a result of radiation exposure, more than 700,000 civilians and warfighters have filed a lawsuit against the Tokyo Electric Power Company (TEPCO).[30] The *Stars and Stripe* stated that plaintiffs claim they have suffered a number of health ailments such as headaches, cancer, tumors, and gynecological bleeding; warfighters may have been exposed to radiation for as long as one month.[31]

When used appropriately, group context-aware applications have the capability to enhance SSA in HA/DR efforts such as this. Damaged and destroyed infrastructure can be mapped to develop safe logistical routes to deliver vital supplies. As infrastructure and services are restored, group members receive the most accurate, relevant information. Moreover, as supplies are distributed, applications can monitor supply inventories and submit requests for additional support as needed. Additionally, sensors can detect contamination and radiation and immediately alert group members to discovered health dangers.

2. Case Study #2: Combat Operations

In 1996, the Taliban, an Islamic extremist group, captured Afghanistan's capital city, Kabul, and held control of the national government. A strict Islamic law characterized Taliban rule. Television, music, and dancing were prohibited. Women were not allowed to work, go school, or play sports and were required to wear a burqa, a dress that covers the body from head-to-toe. Violators of

---

[29] *The Future of Japan: Hearing Before the Subcommittee on Asia and the Pacific of the Committee on Foreign Affairs House of Representatives*, 112th Cong. 1 (2011) (statement of Battalion Chief Robert J. Zoldos II, Program Manager, U.S.A.-1/VA-TF1, Urban Search & Rescue, Fairfax County Fire and Rescue Department).

[30] Matthew M. Burke, "Lawsuit expands over radiation exposure during Fukushima disaster," *Stars and Stripes*, http://www.stripes.com/news/lawsuit-expands-over-radiation-exposure-during-fukushima-disaster-1.211889, accessed 15 September 2013.

[31] Ibid.

Taliban law were severely punished. It was common for the Taliban to amputate body parts, rape, murder, or stone presumed offenders. The Taliban slaughtered thousands on the basis of ethnic differences or suspicion and provided a safe haven for radical organization, al Qaeda.

On September 11, 2001, al Qaeda conducted a series of coordinated attacks against the United States. Terrorists hijacked four passenger airlines; two planes crashed into the World Trade Center, one into the Pentagon, and another crashed in Pennsylvania. The attacks resulted in the deaths of almost 3,000 people. Three days later Congress passed the Authorization for Use of Military Force Against Terrorists (AUMF)[32], and within two months, the United States launched Operation Enduring Freedom (OEF). The military objectives included the destruction of terrorist camps and infrastructure and capture of al Qaeda leaders in Afghanistan.

a. Operation Enduring Freedom

OEF was launched on October 7, 2001. During the initial stage, a relatively small footprint of ground troops was deployed. Major combat operations consisted primarily of air strikes and the launching of cruise and Tomahawk missiles. In December 2001, the Taliban regime surrendered and the high-intensity warfighting phase ended. Although, deployed forces did not capture al Qaeda leader, Osama bin Laden, the al Qaeda network was dispersed and the terrorist regime was removed from power in Afghanistan.

Subsequently, the United States transitioned military efforts to stabilization operations. Failed and failing states provide a haven for extremists and breed organized crime. Thus, the United States' primary aim in Afghanistan was to create conditions for the establishment of enduring democratic governance and trustworthy national military. Military missions in Afghanistan range from humanitarian to special operations. President George W. Bush

[32] Note: AUMF, also referred to as Public Law 107-40, was passed by Congress to grant the President authority to use force against those who contributed to the September 11, 2001, attacks.

delivered a speech emphasizing the need to create opportunities for freedom of expression and economic growth in failed or failing states to reduce the support for terrorist groups. In his address, Bush stated that

> the security of our nation depends on the advance of liberty in other nations. On September 11, 2001, we saw that problems originating in a failed and oppressive state 7,000 miles away could bring murder and destruction to our country. We saw that dictatorships shelter terrorists, feed resentment and radicalism, and threaten the security of free nations.[33]

One tactic employed in OEF was to conduct stabilization and reconstruction (SR) operations with the use of Provincial Reconstruction Teams (PRTs). Stabilization operations include "counterinsurgency operations, peace operations, security assistance, and combating terrorism."[34] Reconstruction activities include "training civil administrators, improving essential services and public safety, supporting civil society and self-determination, and promoting the rule of law and economic development."[35]

The United States developed the concept of PRTs in 2002. PRTs are civilian-military teams located in hostile areas of Afghanistan who deploy counterinsurgency operations against al Qaeda and Taliban forces. The concept of PRTs was "to expand legitimacy of the central government to the regions and enhance security by supporting security sector reform and facilitating the reconstruction process."[36] PRTs are composed of personnel with various experiences to include: Civil Affairs, Medical, Military Police, Intelligence, and Infantry. PRTs are critical to removing internal hiding places and sanctuaries for insurgents. Teams work closely with the provincial government to determine projects and reconstruction activities for the local population to gain support.

---

33 President George W. Bush, "President Discusses Freedom and Democracy in Iraq" (Presidential address, Foundation for the Defense of Democracies, Washington, D.C., March, 13, 2006).

34 Michael J. McNerney, "Stabilization and Reconstruction in Afghanistan: Are PRTs a Model or a Muddle?" *Parameters* 35 (Winter 2005): 32−46.

35 Ibid.

36 Ibid.

PRTs facilitate construction of schools, government buildings, health clinics and community centers. Each team provides its own force protection when operating outside the PRT's base. Although a PRT's primary effort is nonlethal supportive effects, teams can support tactical units within their operating area.[37]

The potential benefits of using group context-aware applications in combat operations are only limited by available resources. Group context-aware applications can track the location of other team members as well as the enemy. Units on patrol can share and update maps to generate an accurate area map. Furthermore, applications can use historical trends to predict enemy patterns of behavior.

3. Benefits of Group Context-Aware Applications

In HA/DR efforts and combat operations, warfighters are deployed in teams or groups that operate in dynamic environments. Group communication is vital to mission accomplishment and the survivability of group members. The information that is critical is determined by what the individuals and the group as a whole can directly and indirectly sense. Mission-oriented efforts can be classified into seven categories: tracking, mapping, reporting, health, force protection, basic services, and resources/logistics. Group context-aware applications can enable the integration of information to support these efforts.

a. Tracking

Position-based applications can assist warfighters track individuals and groups. In combat operations, warfighters benefit by knowing the location of other team members as well as the enemy. HA/DR responders, such as in the Japan 2011 Earthquake and Tsunami, also benefit from sharing locations of key personnel and resources.

---

[37] Robert J. Pauly Jr. and Robert W. Redding, "Denying Terrorists Sanctuary through Civil Military Operations," in *Countering Terrorism and Insurgency in the 21st Century*, ed. James Forest, 273-297 (Westport, CT: Praeger, 2007).

b. Mapping

Applications capable of mapping are an important aspect to warfighters operating in tactical environments. After a natural disaster, infrastructure can be damaged or destroyed. First responders need accurate maps for logistics and recovery efforts. Combat teams patrolling need annotated maps to identify structures that may be hiding places for insurgents. As roads or bridges are repaired, warfighters would be well served by having the latest evacuation plan and transportation routes.

c. Reporting

Messaging applications can deliver critical information about the group, victims, displaced people, and numerous other situations.

d. Health

Warfighters could benefit from supportive applications that monitor vital signs, radiation, sanitation, and air quality. The ability to share information about health emergencies is critical to alleviate deteriorating health conditions and avert epidemics. Applications can filter information and alert appropriate emergency personnel to help coordinate an effective emergency response.

e. Force Protection

Supportive applications can use historical trends to determine if a group is entering a dangerous area. Combat patrols can identify locations where warfighters are frequently attacked. Disaster responders can identify areas where crime is highest.

f. Basic Services

Applications could track information about availability of electricity, water, sanitation, transportation, and housing. Loss of critical services can be reported allowing for plan modifications. As engineers repair facilities and utility systems, groups can be alerted of restored services.

g. Resources/Logistics

Events may unfold rapidly and requirements change. Food, water, medical supplies, shelter, and ammunition can mean the difference between life and death. Continuous dialog is needed to ensure the right support arrives in a timely manner. Applications can monitor resources as well as generate and receive requests for assistance or resources.

D. A REFERENCE MODEL FOR GROUP CONTEXT-AWARE APPLICATIONS

In 2013, Grace Lewis, Marc Novakouski, and Enrique Sánchez of Carnegie Mellon University Software Engineering Institute developed Information Security to the Edge (ISE) to address the challenges of first responders and warfighters operating at the tactical edge[38]. The ISE architecture supports changes in context data models, context data storage mechanisms, context reasoning engines and rules, sensors, communication mechanisms and context views.

Lewis, Novakouski, and Sánchez identify three desired capabilities of group context-aware applications in hostile environments: 1) capture and store context information on a mobile device in a non-intrusive manner to reduce cognitive overload and without imposing an unreasonable burden on handheld device resources, 2) disseminate context information to group members using whatever communications mechanisms are available at the moment, and 3) integrate local and group context information that is relevant to the individual and mission according to configurable rules.[39] ISE presents a reference architecture for implementing group context-aware mobile applications that enables these capabilities. The ISE architecture follows the concept of Model-View-Controller (MVC).

---

[38] In this research, *tactical edge* is defined as a hostile environment with limited resources. Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

[39] Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

1. Model-View-Controller

MVC is a software architecture pattern that was designed by Trygve Reenskaug in 1979 to solve the problem of representing complex real-world systems.[40] The objective of MVC is to develop applications in a modular way, support the development of graphical user interfaces (GUIs), and use object sharing to endorse software reusability. MVC architecture has three distinct elements: model, view, and controller.

a. Model Element

The model is the core element of MVC and is used to manage information. It manages the performance and data of the application and notifies the view when its state changes.

b. View Element

A view controls the representation of data by displaying all or a fraction of the data. It specifies how data should be presented and displays the state of the model, typically with a user interface element.

c. Controller Element

A controller facilitates changes to the state of the model or view. It operates by accepting input from the user. Then, based on the input, instructs the model and view to perform actions.

2. Information Security to the Edge Architecture

In the ISE architecture, the model is the Application Data in the Input/Output (I/O) Layer, the controller is the Application Layer, and the view is the User Interface Layer (see Figure 3).

---

40 Matjaž Debevc, Andreas Holzinger, and Karl Heinz Struggl, "Applying Model-View-Controller (MVC) in Design and Development of Information Systems: An Example of Smart Assistive Script Breakdown in an e-Business Application" (paper presented at the Proceedings of the 2010 International Conference on e-Business (ICE-B), July 26–28, 2010).

a. User Interface Layer

The *User Interface Layer* is the collection of views of context data.[41]

[41] Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

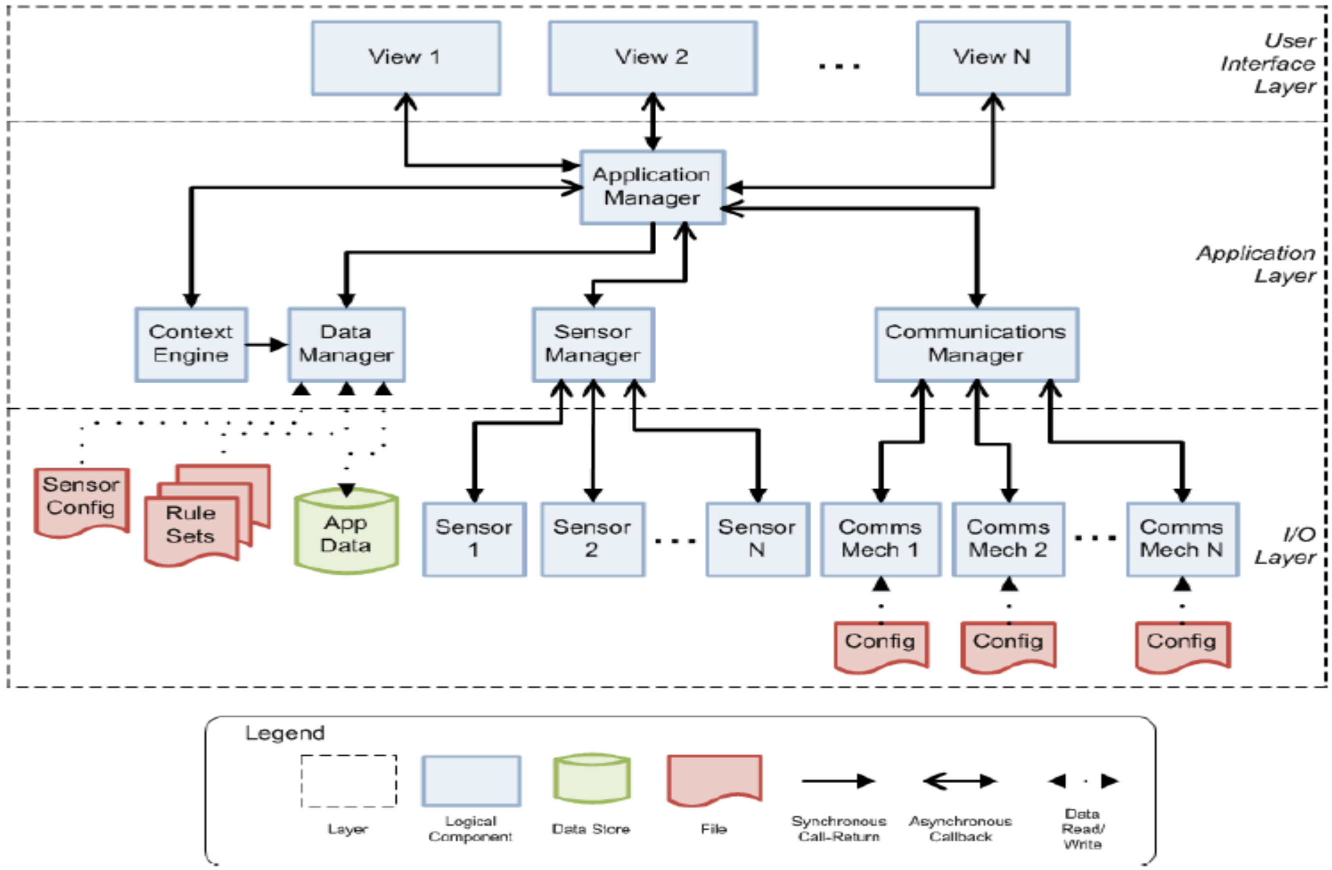


Figure 1. ISE Reference Architecture [1]

[1] Source: Lewis, Novakouski, and Sánchez, “A Reference Architecture,” 44–63.

a. Application Layer

The *Application Layer* is the core of the system. Based on individual and group context, these components manage context and create events:[1]

(1) All system activity is routed through the *Application Manager*, which distributes events to appropriate components.

(2) The *Context Engine* processes all context information used by ISE to detect events as they occur. Device sensors report new data and context data is received from group members.

(3) The *Sensor Manager* receives data from the sensors. It also controls the sampling rate and change thresholds for each sensor.

(4) Inbound and outbound messages are passed through the *Communications Manager*.

(5) The *Data Manager* performs all create, retrieve, update, and delete operations on context data and application data. It manages all access to the sensor configuration file and the context rule sets.

b. Input/Output Layer

The I/O Layer components interact directly with files, databases, sensors and communication services:[2]

(1) *Sensor Config* file contains the default sensor configuration information for all active sensors.

(2) *Rule Sets* are files that contain rules about which the context engine reasons. All group context-aware mobile applications have a generic default rule set, but rule sets can be created or modified as needs change.

(3) *App Data* is the physical storage.

---

[1] Lewis, Novakouski, and Sánchez, “A Reference Architecture,” 44–63.

[2] Ibid.

(4) *Sensor 1* to *Sensor N* correspond to the components that receive data from sensors. All sensors implement the same interface to allow addition of new sensors to the system.

(5) *Comms Mech 1* to *Comms Mech N* correspond to the communication mechanisms that are used to send and receive data within the group. Each communication mechanism has a configuration and all implement the same interface so that communication mechanisms can be easily added to the system.

3. Information Security to the Edge – Group Context-Awareness

a. Context Model

(1) Logical Data Model

ISE expands Dey's definition of entity. The Person entity is changed to People and is divided into three subcategories: individuals, groups, and organizations. Groups and organizations can be divided into subcategories.[45]

[45] Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

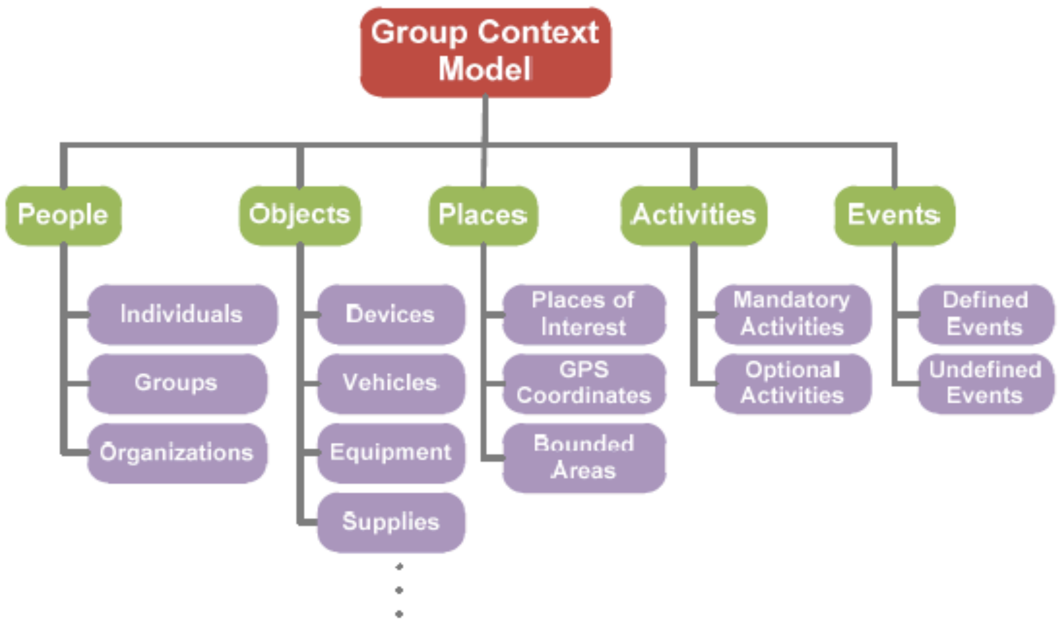


Figure 4. Group Context Model [46]

(2) Physical Data Model

The physical data model is implemented using an Object-Oriented Database (OODB) approach to allow expansion of the data model. Warfighters, who operate in tactical environments, will benefit from a data model that adapts to varied situations. Memory scalability is a concern because OODB holds most of its data in memory.[47] Warfighters require continuous context sensing, which produces a large sensing data set. This model manages this concern by regulating sensor usage and only sharing relevant data between group members.[48]

b. Context Sensors

The ISE model uses a standardized interface that all onboard sensors can use. This limits the application's effort to integrate a new sensor to: 1) the development of the sensor control logic, 2) a few lines of code in the

---

[46] Source: Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

[47] Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

[48] Ibid.

application itself, and 3) a new line in a configuration file.[49] The sensor management capability manages sensor status reporting. All sensors are started as bound services and processes gathering sensor information are decoupled from the main application while following the defined service life cycle.[50] If a sensor service stalls, the process management infrastructure will limit the impact on the primary application (see Figure 5).

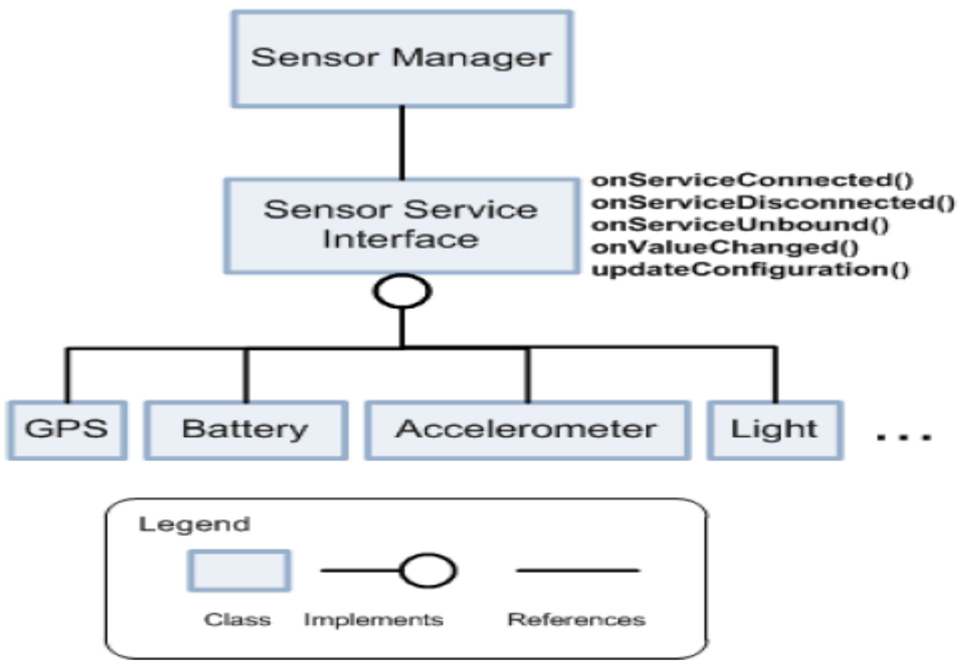


Figure 5. Sensor Service Interface [51]

c. *Context Dissemination*

This model supports the addition of new communication mechanisms. The communication service interface shown in Figure 6 provides general communication methods and callbacks that increase flexibility allowing the application to use any available channel.

---

49 Lewis, Novakouski, and Sánchez, “A Reference Architecture,” 44–63.

50 Ibid.

51 Source: Lewis, Novakouski, and Sánchez, “A Reference Architecture,” 44–63.

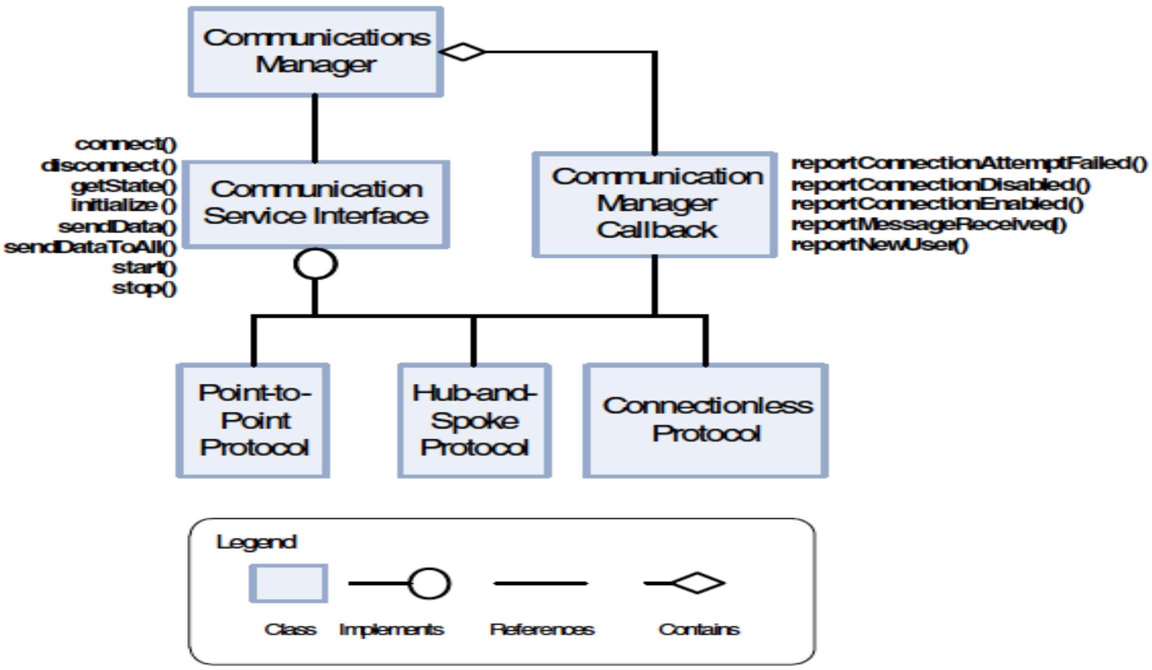


Figure 6. Communication Service Interface [52]

d. Context Engine

ISE encodes all rules in rules sets. Rule sets can be changed at design time or runtime. Each rule is evaluated against a specific data type and contains a *RuleName*, *TriggerDataType*, *Conditionals*, and *Actions*:[53]

The *TriggerDataType* field specifies the triggering data type, and only rules that are triggered by the data type of the data item are executed.

Conditional statements are included in the rule's *Conditionals* field. The use of semi-colons enables conditionals to be combined.

If a rule is evaluated to be true, the *Action* field specifies the actions that an application is allowed take to include sending information to members of the group.

52 Source: Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44–63.

53 Lewis, Novakouski, and Sánchez, "A Reference Architecture," 44_63.

A sample rule is shown in Figure 7. The rule, *BatteryLow*, uses the battery sensor to report changes in the battery level by triggering a *BatterySensorDataUpdateEvent*. The rule observes the *batteryCharge* field to detect if the battery level is less than (*LT*) thirty percent. If the rule evaluates to True, then two actions occur: 1) *Alert 1* is created and 2) *BatteryLowEvent* is declared.

```
<Rule
 RuleName = "BatteryLow"
 TriggerDataType = "BatterySensorDataUpdateEvent"
 Conditionals = "TRIGGER.batteryCharge LT 30"
 Actions = "ALERT 1; DECLARE BatteryLowEvent"
/>
```

Figure 7. Sample Context Rule [54]

4. Evaluation

ISE is an extensible strategy that addresses the challenges of group context-awareness for warfighters operating in tactical environments. The architecture supports the incorporation of various sensors and communication mechanisms. Rule sets direct the context engine on contextual decisions and allow groups to share relevant information to enhance situational awareness and develop a more accurate COP. One challenge that ISE fails to address is resource scarcity. In a comment posted to the Carnegie Mellon Software Engineering Institute Blog on May 21, 2012, Marc Novakouski stated that ISE does not exploit an energy-efficient method for sensing which could save power and bandwidth.

---

[54] Source: Lewis, Novakouski, and Sánchez, “A Reference Architecture,” 44–63.

E. SUMMARY

Warfighters operating in dynamic settings require context information to provide insight into a situation at a specific moment. The essential technique in generating military context is to accurately sense and interpret context to provide useful data. Group context-aware mobile applications provide solutions to help warfighters meet mission goals. Discovering patterns and details about individuals, as well as groups, help warfighters understand their own state as well as the state of the group. SSA is vital in decision making for the support of dispersed groups operating in tactical environments. The military is becoming increasingly involved in small-scale engagements throughout the world. Group context-aware applications can support group communication in HA/DR and combat operations. ISE addresses the challenges of first responders and warfighters operating at the tactical edge and supports group context-aware mobile applications.

III. ENERGY-EFFICIENT SENSOR MANAGEMENT STRATEGIES

Context-aware applications are resource intensive, continuously acquiring data from the sensors of the device. This presents several challenges for mobile context-aware applications that are limited to a great extent by mobile devices' relative lack of memory scalability, high energy consumption, and low processing power. Although great strides have been made to improve memory scalability and processing power, progress in battery technology has remained slow.[55] Consequently, there is a need for adaptive mechanisms, which enable mobile devices to adapt in dynamic environments and collect group context. The literature in the area of energy awareness provides various approaches for reducing energy consumption of sensors. This chapter explores two sensor management strategies, SENST and ACE, and proposes Group Energy-Efficient Management (GEM) as an energy-efficient sensor management model for group context sensing.

A. SENSOR SUBSTITUTION AND TRIGGERING (SENST)

SENST is a dual approach that uses sensor substitution and sensor triggering to "balance the energy consumption of various sensor-driven context recognition alternatives."[56] Sensor substitution reduces energy consumption by replacing energy-intensive sensor with sensors that require less energy whenever context can be determined without sacrificing accuracy. Sensor triggering takes advantage of the different energy consumption characteristics of sensors to activate or inhibit sensors in the determination of context. Sensor triggering saves energy by using energy-intensive sensors only when required.

---

[55] Gurminder Singh, "Processor Technology for Mobile Devices," (class lecture, Mobile Devices from Naval Postgraduate School, Monterey, CA, September 2012).

[56] Schirmer and Höpfner, "SENST," 250–263.

Maximilian Schirmer and Hagen Höpfner of Mobile Media Group developed SENST to address the energy consumption constraints of smartphones. However, the approaches presented in SENST can be applied to the diverse devices that are capable of ubiquitous computing. This approach places sensors into four categories: technology that originally serves as communication interfaces, environment sensors, multimedia features of smartphones, and software sensors.[57]

Wireless Local Area Networks (LANs), Bluetooth, and digital cellular networks are *Communication Interfaces* can be used to determine a mobile device's location by triangulation. These communication interfaces can be used as substitutes to location sensors. Environment conditions, energy resources, and sensor availability determine the application of sensor substitution. Communication Interfaces have high energy consumption (see Table 2).

Environment Sensors include GPS sensors, ambient light sensors, proximity sensors, accelerometers, and the gyroscope. GPS sensors provide precise locations; however, GPS location acquisition takes time and uses a large amount of energy. Communication interfaces can provide data to GPS sensors and shorten the location acquisition time. Ambient light sensors detect ambient light levels and can be used to indicate indoor/outdoor activity. The proximity sensor recognizes the presence of objects. This sensor can be used for gesture recognition. The accelerometer measures the device's acceleration and can be used to indicate activities of the device. Gyroscope sensors provide accurate values for the device's position in space, which help identify position or orientation changes.

[57] Schirmer and Höpfner, "SENST," 250–263.

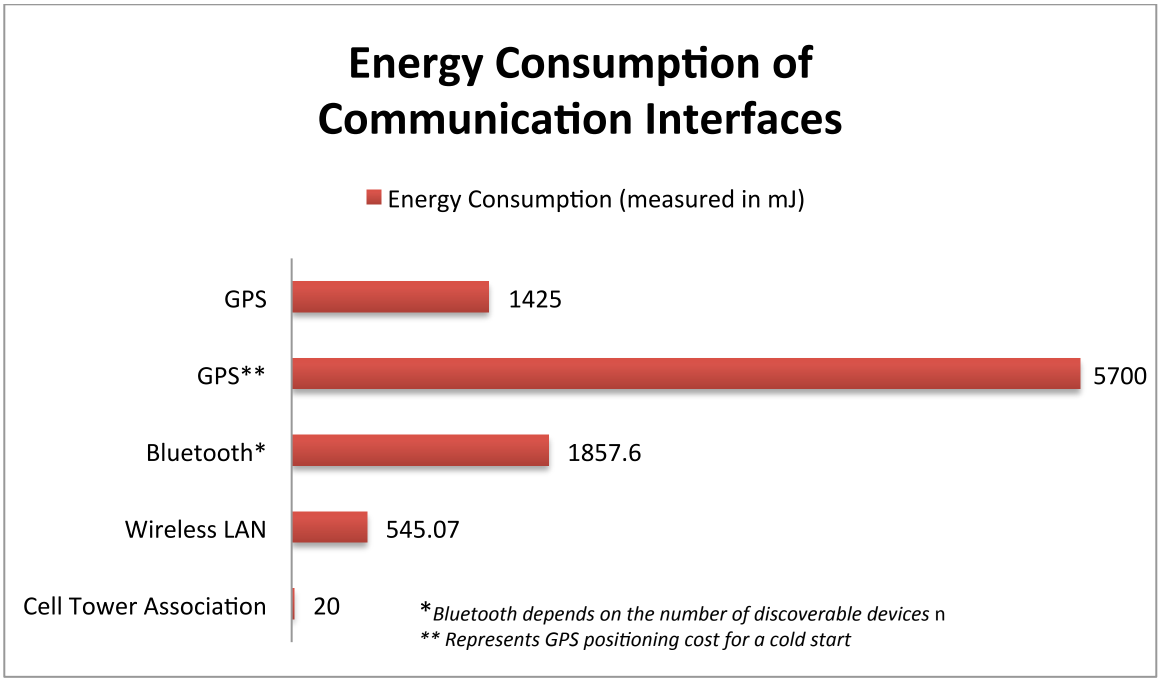


Table 2. Communication Interfaces Energy Consumption [58]

Multimedia Features, such as microphones and cameras, can be used as sensors to draw conclusions about the location of the device, identify people in the vicinity, detect and identify known objects, or determine if the user is near the device.

Software Sensors draw conclusions by using data stored and processed by applications on the device to determine context.

1. Sensor Substitution

There are various ways for a mobile device to determine context. Often applications can use different sensors to acquire data, but these sensors have different energy consumptions.[59] For example, location may be determined by GPS or through triangulation. Although GPS is more precise, it requires more energy. However, some applications only require approximate locations. By

[58] Data taken from Source: Schirmer and Höpfner, "SENST," 250–263.

[59] Schirmer and Höpfner, "SENST," 250–263.

measuring the *appropriateness*[60] of alternative sensors, SENST can determine the best sensor to use in a given scenario.

SENST uses an algorithm that receives the set of available sensors, requested context element, and the minimal accepted appropriateness to choose the most energy-efficient sensor substitution (see Appendix A). A major shortcoming of this algorithm is that it does not include software sensors, which hold a large source of data that defines military context.

2. Sensor Triggering

Sensor triggering aims to use sensors with the lowest energy costs to determine environmental changes and to use high cost sensors when changes signal a need to update context values.[61] Positive and negative triggering of sensors indicates changes. According to Maximilian Schirmer and Hagen Höpfner, "If a device is not moving, then the location cannot change."[62] This is an example of how sensor triggering measures changes through sensors. If the accelerometer sensor does not register movement, then the device does not need to acquire the location.

SENST monitors sensors by polling sensor changes. While some sensors, such as proximity and ambient light sensors, use interrupt requests to signify changes, others must be frequently polled to monitor changes. SENST uses an algorithm that looks at the set of available sensors, available positive and negative triggers, requested context element, and the time to wait between loops to monitor sensors. (See Appendix B.) SENST triggers sensors only when a certain threshold value is reached. As a result, SENST may not always show the most accurate context values.

---

60 In this research, the *appropriateness* of a certain context element's value depends on the probability that the value is correct for the given application. Source: Schirmer and Höpfner, "SENST," 250–263.

61 Schirmer and Höpfner, "SENST," 250–263.

62 Ibid.

3. Evaluation

The combination of sensor substitution and sensor triggering may reduce energy consumption of mobile device applications. Sensor substitution is a viable approach to acquiring context with lower cost alternatives in appropriate scenarios. Sensor triggering polls sensors to determine changes and only uses high cost sensors when changes signal a need to update context values.

There are drawbacks to this solution. Sensor substitution does not include software sensors in the set of available sensors. Furthermore, sensor triggering only occurs when a threshold value is reached. Consequently, this method may produce stale results.

Evaluation of SENST was conducted on the Apple iPhone Operating System (iOS) 4.3.1 platform, running on an Apple iPhone 3GS. The trigger method, a naïve polling method, and the continuous positioning method of the Apple iOS 4 Software Development Kit (SDK) were implemented on a predefined 1.4km course (see Appendix C). The energy consumption and accuracy of each method were compared. Results indicate the most significant decrease in energy consumption was realized using the sensor trigger method (see Appendix D). However, the sensor triggering method collected fewer coordinates along the course and the average accuracy was lower than the polling and SDK method (see Appendix E).

B. ACQUISTIONAL CONTEXT ENGINE (ACE)

In 2012, Suman Nath of Microsoft Research presented ACE, an energy efficient approach that provides context to applications while exploiting correlation of known context attributes. ACE is a middleware[63] that dynamically learns relationships of context attributes. It exploits the correlation of learned attributes to infer desired context attributes by using less energy intensive

63 *Middleware* is an event notification infrastructure system. It provides a multilevel conversion and transformation process where sets of rules are applied on the raw data to infer context elements, Source: Nath, “ACE,” 1472–1486.

context attributes to infer energy intensive context attributes through *inference caching* and *speculative sensing*.

1. ACE Components

ACE consists of the following components: Contexters, Raw Sensor Data Cache, Rule Miner, Inference Cache, and Sensing Planner (see Figure 6).[64]

Contexters are a collection of modules, which determine the value of a context attribute by acquiring data from necessary sensors. A Contexter requires two pieces of information: the name of the attribute that needs to be determined and its energy cost.

Raw Sensor Data Cache is a standard cache, which stores the values of context attributes.

Rule Miner maintains the user's context history of time stamped tuples for a predefined expiration period. It automatically learns relationships among various Boolean context attributes and generates context rules. Each rule has a minimum support and confidence percentage, which can be modified to achieve an acceptable level of accuracy. The rule miner requires daily offloading of all context tuples to a remote server to compute rules. Incremental updates on the mobile device are required to delete existing rules that are no longer valid.

The *Inference Cache* functions as a traditional cache with a *Get/Put* interface.

When a requested context attribute is not located in the cache, the *Sensing Planner* uses speculative sensing to find the sequence of proxy attributes to determine the value of the requested context attribute using the least expensive sensor.

---

64 Nath, "ACE," 1472−1486.

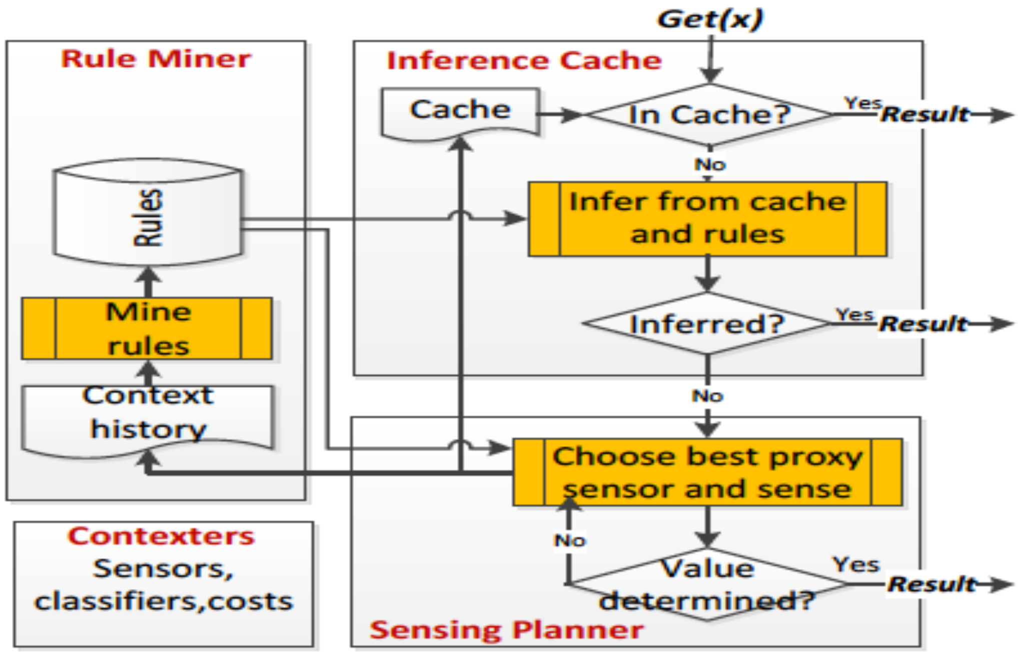


Figure 8. Workflow in ACE [65]

2. Inference Caching

The Inference Cache functions like a traditional cache. It allows ACE to infer requested context attributes from known context attributes without acquiring sensor data. The Inference Cache provides a *Get/Put* interface. A *Put(X,v)* places a context attribute, *X*, in the cache for a predetermined amount of time, *v*. A *Get(X)* returns the value of *X* if the value is in the cache or can be inferred from context rules and cached values of other context attributes.

ACE constructs expression trees to represent the Boolean expression. The expression tree is a Boolean *AND-OR* tree, where a non-leaf node presents an *AND* or *OR* operation on the value of its child and a leaf node represents a tuple. The expression tree for a tuple evaluates to *True* if the tuple holds according to context rules. See Figure 7 for an expression tree for the tuple *Indoor* = *True*. The Inference Cache maintains one expression tree for each

[65] Source: Nath, “ACE,” 1472–1486.

tuple. If the expression tree cannot determine the value of a context attribute to be *True* or *False*, then the Sensing Planner is invoked.

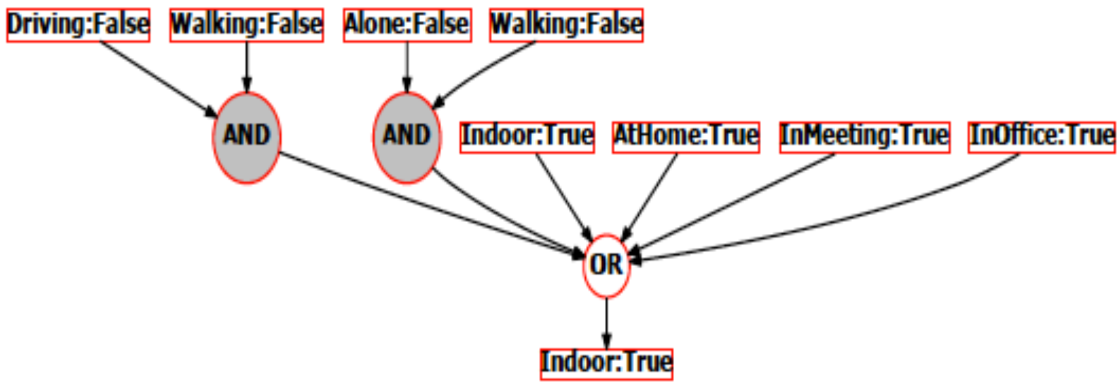


Figure 9. An Expression Tree for Indoor = True, shown upside down [66]

3. Speculative Sensing

If the Inference Cache fails to determine the value of a context attribute, then ACE uses speculative sensing to discover additional proxy attributes. ACE attempts to exploit a low-cost attribute to determine the value of an expensive attribute. A sensing plan is developed to determine an optimal ordering based on energy cost, C_i, and the likelihood that an attribute will return a *True* value, P_i, (see Appendix F).

The algorithm to evaluate a Boolean expression with minimum cost works well for a small number of attributes. However, when the number of attributes is large, the sensing cost to find the optimal plan may be more than the cost to directly acquire the requested attribute. In such cases, ACE uses a heuristic algorithm to rank the attributes based on C_i and P_i (see Appendix G).

4. Evaluation

Overall, ACE is capable of reducing the energy consumption of mobile devices. It mitigates sensing costs by inferring context attributes from known attributes. ACE learns relationships of expensive and low-cost context attributes

[66] Source: Nath, "ACE," 1472–1486.

automatically. In providing context information to applications, ACE can reduce sensing costs for continuous context sensing application.

Although ACE presents an energy-efficient solution, it has several shortcomings. If a user deviates from expected behavior or context values become invalid, ACE will occasionally infer the context incorrectly and return erroneous results. This can be dangerous to warfighters who require a high level of accuracy. Although the Rule Miner can mitigate inaccuracy, it cannot eliminate the problem. This could introduce bugs that are difficult to debug. In addition, ACE lacks temporal correlation across context attributes. If a warfighter is on foot patrol three miles from the forward operating base (FOB), then it can be inferred that the warfighter will not be able to return to the FOB for approximately one hour.

Evaluation of ACE was performed on a Samsung Focus phone running Windows Phone 7.5 Operating System (OS). Two datasets and three applications were used for evaluating the sensing energy consumption of ACE. Results concluded that the Inference Cache can reduce sensing energy consumption by 220% (see Appendix H) and the Sensing Planner can further reduce the energy consumption by an additional 200% (see Appendix I). Overall, results prove that applications running on ACE consume 420% less energy than a baseline cache (see Appendix J).

C. GROUP ENERGY-EFFICIENT MANAGEMENT (GEM)

Although SENST and ACE present energy-efficient solutions for mobile devices, both are designed to reduce the power consumption of individual devices. However, warfighters operating in tactical environments typically work in groups. They often work in high-risk environments with limited resources. Group context-aware applications are providing valuable, life-saving information about users and the physical environment. Despite advances in group context-awareness, mobile devices are still limited in size and capacity. Motivated by this fact, Group Energy-Efficient Management (GEM) is presented to address energy-efficiency and resource scarcity issues of warfighters working in groups in high-risk environments.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. GROUP ENERGY-EFFICIENT MANAGEMENT (GEM) FRAMEWORK AND IMPLEMENTATION

Sensor technologies have a strong impact on the energy consumption of mobile devices. Considerable research has been conducted in the field of context awareness in mobile environments to address energy limitations. Numerous techniques, such as SENST and ACE, have been developed to minimize the use of energy-intensive sensors for individual context-awareness. With the emerging development of group context-aware applications, contextual information from individuals and team members can be shared to provide situational awareness. However, limited research has been conducted on leveraging group context-awareness to collectively manage scarce resources such as storage, bandwidth, and battery power. Inspired by the advances of Lewis, Novakouski, and Sánchez's reference architecture for group context-aware applications to address the issue of scarce resources in tactical environments, GEM is designed as a conceptual framework that aims to assist in the development of energy-efficient group context-aware applications. GEM is distinctly different from other energy-efficiency strategies in that the focus is on using group resource integration to increase energy and resource efficiency.

A. REQUIREMENTS FOR HANDLING GROUP CONTEXT AND RESOURCE INTEGRATION

Group context-aware applications are an attractive means to develop and share situational awareness among group members. Current energy-efficiency strategies maximize the individual's energy consumption. Yet, this is insufficient for warfighters operating in tactical environments. Warfighters require energy-efficient strategies that benefit the group as well as the individual. This is one of many ways in which GEM distinguishes itself from other approaches. While prior research focuses on optimizing energy consumption on a per device basis, GEM aims to optimize energy consumption for a group of devices. Even though GEM focuses on optimizing group resources, individual resources can and should still be optimized to the fullest extent possible. Existing energy-efficiency approaches, such as SENST or ACE, manage individual device resource optimization and complement GEM's energy-efficiency strategy.

To solve the warfighters' energy consumption issues, group energy-efficient strategies must address the following challenges in networks comprised of mobile devices: 1) resource discovery, 2) context storage/interpretation, and 3) transparency/sufficient communication.

1. Resource Discovery

Mobile devices possess limited resources, and context sensing, especially if it is continuous, can be resource intensive. The power source is a battery with limited energy capacity. In tactical environments, mobile device maintenance[67] is impractical. There are many ways to enhance individual management of scarce resources. For a group to communicate with a sensor, the group must know where the sensor is located and what communication mechanism to use to reach the sensor. To effectively manage group resources, each group member must advertise available resources and notify the group of significant changes.

2. Context Storage/Interpretation

Context information must be stored to allow applications to access it when needed. Individual group members may use group resources to store context information as necessary to maximize resource utilization. In addition, all members of a group may require GPS information for imagery, maps, or etc. Energy-efficient group context-awareness can limit group energy consumption by sharing context acquired by high energy cost sensors, such as GPS, throughout the group. To ensure applications can access stored context information, context history must be maintained for all group members.

[67] In this thesis, mobile device maintenance is defined as replacement or recharge of power source.

3. Transparency/Sufficient Communication

When dealing with group context, multiple mobile devices are being used to sense context. However, the fact that communication is distributed must be transparent to the group. Capricious availability of resources due to environmental obstructions, device failures, and communication mechanism failures/inconsistencies can affect group communications. Environmental obstructions can delay or prevent the transmission of data. Temperature, wind, dust, sunlight, and rain may make it impossible to accurately use device sensors that observe and report the environment conditions. As available communication mechanisms change, contact with other group members is affected. Automatic configuration of communication mechanisms is essential to determine how context information can best be communicated throughout the group.

B. GEM LOGICAL COMPONENTS

GEM presents a framework that complements the substantial realm of sensors and supports the ability to collect and distribute group contextual information. To address group context-aware application issues, the following components are used in the conceptual framework (see Figure 10):

1. Group Manager

The *Group Manager* is responsible for the seamless integration of group resources. When a change occurs, the group manager evaluates the effect of the change on the group members. As resources and communication mechanisms update, the assignment of the role of group manager may change. The group manager receives time-stamped context information from group members and sends it to the context engine. Multiple group managers can be assigned and linked to create sub-groups within a group or to provide redundancy. The group manager is chosen using a weighted scoring of available resources. In the event of a tie, the device with the most recent time stamp will be chosen.

2. Group Member

Each *Group Member* accepts data from device sensors and manages the energy efficiency of the individual device using a combination of SENST and ACE methods. When context can be gathered from group members at a lower energy cost, then the group member pulls the information from the group manager. Group members send acquired context to the group manager unless an alert is triggered which requires immediate notification of all group members. This eliminates the need to send multicast messages throughout the group, which wastes bandwidth and valuable resources. All messages are time stamped and set to a default time-to-live to avoid disseminating stale data due to loss of communication mechanisms or inherent delays.

3. Group Configuration File

The *Group Configuration* file contains default configuration information for communication interfaces, sensors, and security. The group configuration file is a set of rules and configurations that support mission requirements and can be modified as needed.

4. Communication Manager

The *Communication Manager* tracks the communication mechanisms available to pass context throughout the group. It focuses on using the lowest cost route to communicate between group members.

5. Resource Manager

The *Resource Manager* manages the distribution of group resources. The resource manager and communication manager work to develop the most cost effective delivery and storage of context.

6. Context Engine

The *Context Engine* processes all group context information.

7. **Resource File**

The *Resource* file is the file that contains the battery life, storage capacity, and central processing unit (CPU) information.

8. **Sensors File**

The *Sensors* file is a file that contains sensor configuration information on all available sensors.

9. **Neighbors Table**

The *Neighbors* table is a file that contains network routing information.

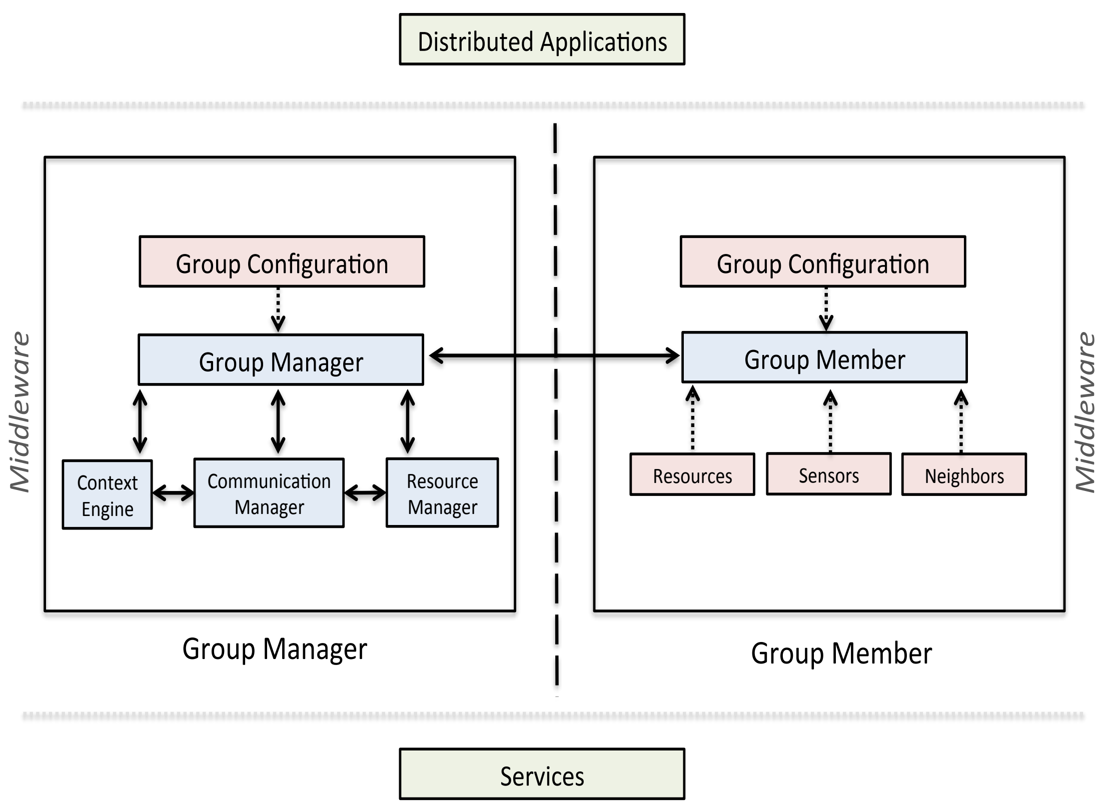


Figure 10. Workflow in GEM

C. HOW TO USE GEM

In GEM, every *Group Member* is assigned a basic role. However, each group has an assigned *Group Manager*, which is responsible for integration and management of group resources. The choice of the group manager is based on remaining resources and the energy cost to communicate with other group members. (See Figure 11.)

The group manager receives information about the resources, sensors, and connected neighbors of each member using a *push-pull* method. Each group member forwards data about the available resources, sensors, and neighbors to the group manager. Group members send aperiodic updates to the group manager when there is a significant change in resources, labeled as delta (Δ), or reportable change in sensors or neighbors since the last report to the group manager. As the group manager's resources diminish and hit a predefined critical level, *CL*, the group manager role is transferred to another group member that possesses more resources and cost effective communication mechanisms. As communication mechanisms change, the group manager will choose the most energy efficient route to communicate with group members.

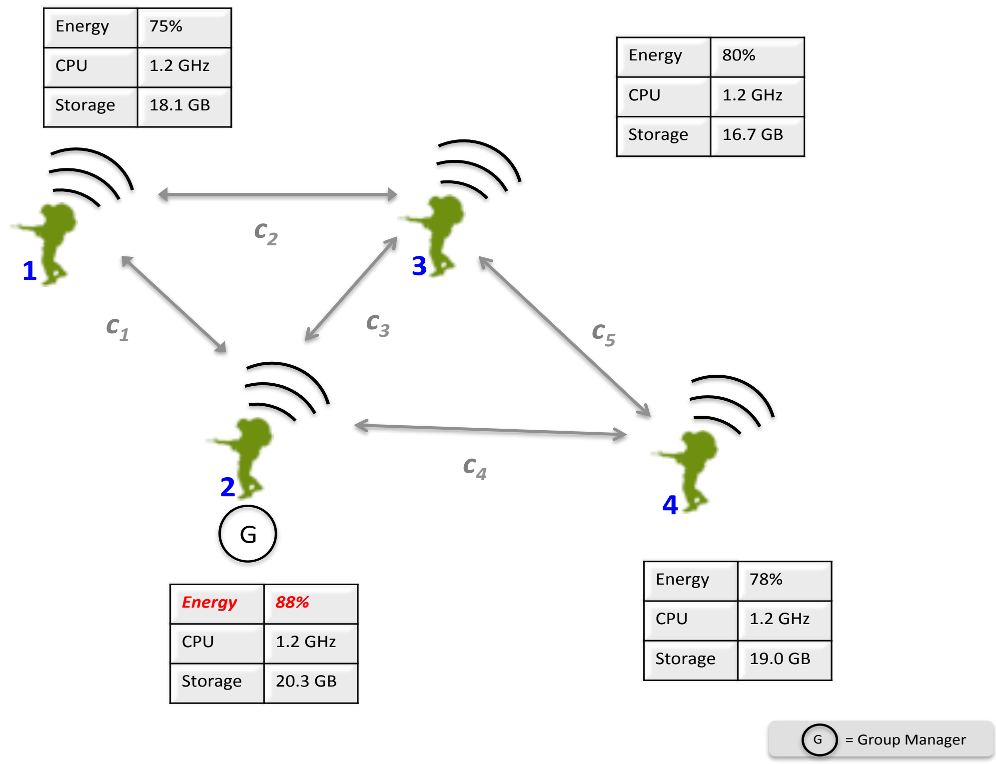


Figure 11. Resources of Group Members

D. A USE CASE IN TACTICAL ENVIRONMENTS

Following is a use-case that illustrates the potential of group context awareness for a tactical operation:

> Afghanistan has reached a critical transition point where there is broad support for the national government. Elections for the National Assembly and provincial councils occurred with few hostile acts, and Afghans were able to elect representatives at the local level. However, widespread corruption and continued violence within the provinces threaten the legitimacy of the national government and could reverse these gains. As the operational center of gravity for SR shifts to the provinces, PRTs are leading stability operations.[68]

> Combined Joint Task Force 76 (CJTF-76) is located at Bagram Airfield, Afghanistan. The CJTF is responsible for all United States led PRTs, including

[68] Scenario introduction generated from United States Agency International Development (USAID) PRTs in Afghanistan Interagency Assessment, 2006.

Bagram PRT which comprises 50 to 100 personnel. Bagram PRT conducts stability operations to deter or defeat the re-emergence of insurgent activities in the Parwan Province. The strategic objective is to facilitate stabilization, reconstruction, and development of government institutions.

Recently, in an effort by the Taliban to impede reconstruction and stability, Taliban fighters targeted the newly constructed Gogamanda Bridge and road, detonating an improvised explosive device (IED) on the bridge. The new bridge and road connect more than 6,000 Afghan families, which had been previously separated by the Gogamanda River, to Kabul. The Bagram PRT is dispatched to evaluate the scene.

Within moments of notification, PRT Commander, Lieutenant Colonel Riley, prepares to send a squad-sized unit to the site. All team members are equipped with mobile devices. Prior to departure, all mobile devices are loaded with group configurations. The critical level and Δ are set to 50 and 10 respectively. (Figure 12 shows the input parameters for the group manager and provides pseudocode for monitoring group resources.) The resources of all group devices are monitored. If a resource of a device reaches a critical level, the device is flagged and the group manager is alerted. In this example, critical resources are identified as battery power and storage capacity. GEM assumes that devices are operating in hostile environments and are unable to restore depleted resources. Therefore, once a device is flagged, it remains flags for the duration of the mission. All devices within the group identify its resources, sensors, and directly connected neighbors. As devices are linked, a group manager is selected based on the information reported by each device. The squad, or group, consists of three fire teams, or subgroups. Each subgroup has a designated group manager. (See Figure 13.)

Algorithm 1. Pseudocode Implementation for Group Manager

```
#Input:
CL     = 50                                #Critical Level
Δ      = 10                                #Significant Change
Δi     = Δ /2                              #As Resources are Depleted,
Smaller Changes in Resources May Be Considered Significant
t      = 30                                #Time Between Loops
#Output:   Alert

def batterySensor():
      return batteryPower

def storageSensor():
      return storageCapacity

def monitorResources(flag = False):
      batteryo = batterySensor()
      storageo = storageSensor()

      while True:
            wait t
            checkResources(flag, batteryo, storageo)

def checkResources(flag, batteryo, storageo):
      if flag == True:
            checkChange(Δi, batteryo, storageo)
      elif batterySensor() or storageSensor() <= CL:
            flag = True
            Alert
            return flag
      else:
            checkChange(Δ, batteryo, storageo)

def checkChange(Δ, batteryo, storageo):
      batteryi= batterySensor()
      storagei= storageSensor()

      batteryCheck = batteryo – batteryi
      storageCheck = storageo – storagei
      batteryo = batteryi
      storageo = storagei

      if batteryCheck or storageCheck >= Δ:
            Alert
            return batteryo, storageo
      else:
            return batteryo, storageo
```

Figure 12. Pseudocode for Group Manager

The PRT is debriefed and a convoy quickly departs the FOB heading towards the Gogamanda Bridge. CJTF-76 standard operating procedures (SOPs) require that convoys provide video surveillance of the convoy route. Vehicle mounted cameras are unavailable; therefore surveillance is recorded on mobile devices. Three group members are designated for this task. Video is energy intensive and requires a large amount of storage. A few hours after leaving the base, the convoy reaches the damaged location on the bridge. The group managers have monitored resource activity of all devices. Devices of the group members tasked with recording the convoy route have reported a significant change in resources. However, group manager assignments have remained the same.

As the team arrives at the bridge, one fire team stands guard while PRT members inspect the bridge for undetonated IEDs, damages, or tampering. Photos tagged with GPS location are taken to document the destruction. The group manager provides accurate GPS data to devices thereby saving energy. The cameras used are high quality and the photos quickly consume a large portion of available storage. The group manager is able to successfully distribute the data among the group to maximize storage and increase efficiency.

Sergeant Finley inspects under the bridge, and he locates a large object that looks like an IED. He immediately confirms his discovery using optics and notifies the group and higher headquarters with an alert on the group's mobile devices. The PRT immediately takes action on the information and evacuates the area to a safe distance. The PRT sweeps the surrounding area checking for secondary IEDs and cordon the area. PRT engineers approach the IED, and device sensors detect that the IED has chemical filler. Immediately the group is alerted of the additional danger. The cordon is expanded, and the team moves upwind to high ground away from the release point and control the immediate area until the chemical IED threat can be destroyed. Task Force Gladiator's explosive ordinance disposal (EOD) team arrives and with the use of a tactical EOD robot, the chemical IED is disposed. PRT engineers determine that the bridge received minor cosmetic damages from the detonated IED and no structural damage was done. Ten hours after arriving to the Gogamanda Bridge, the PRT returns to the FOB.

This was a simple, yet realistic scenario of how GEM can benefit groups operating in tactical environments. SSA helps groups quickly develop a more accurate COP. Warfighters benefit from sharing scare resources. By distributing energy intensive sensing, groups can prolong the life of resources.

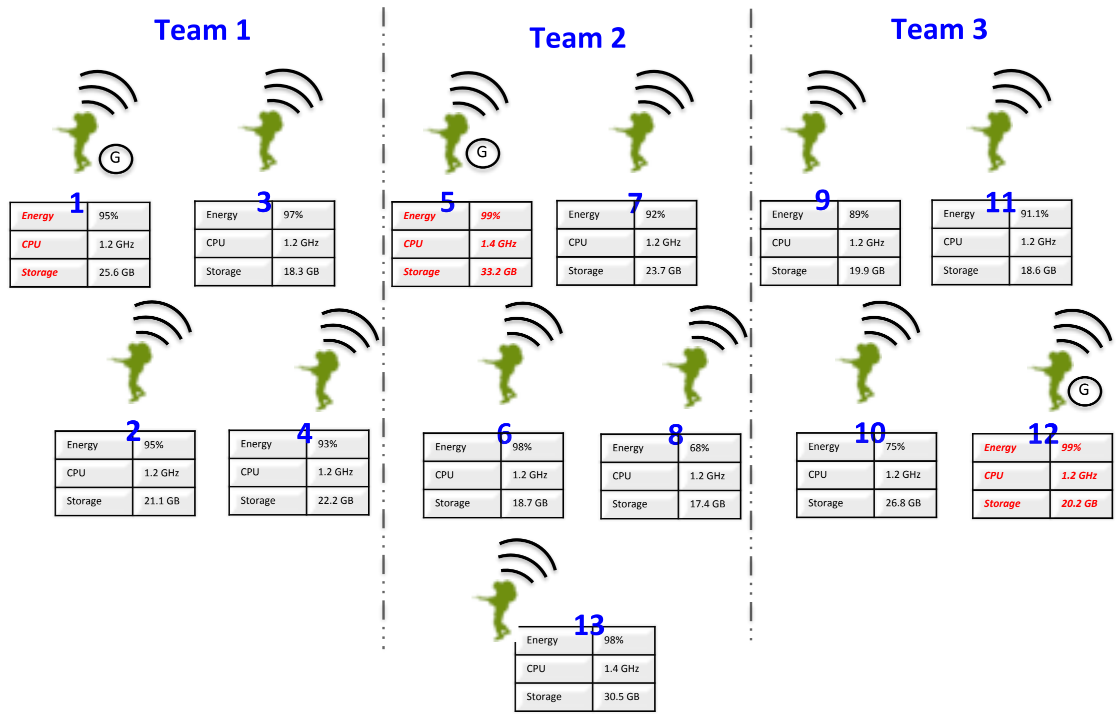


Figure 13. Designated Group Managers

E. ADVANTAGES/DISADVANTAGES OF GEM

GEM is an energy-efficient strategy that allows group context-aware applications to maximize energy efficiency for a group of devices. It allows applications to implement energy saving strategies such as SENST and ACE on each mobile device so that individual device resources are also well managed. In addition, GEM enhances overall energy savings by leveraging and coordinating resources throughout the group. Group context-aware applications are subject to failure due to depleted resources. Implementing a group manager to monitor resources for each group member device prevents premature device failures due to resource exhaustion. GEM not only extends the life of each group member, it optimizes the life of group resources.

However, GEM is only as effective as the data gathered from group members. In addition, with distributed resources, loss of any device communication mechanisms can severely impact the function of the group. Inability to access data on demand prevents group elements from receiving the most accurate, relevant information to form a complete COP.

V. CONCLUSIONS AND FUTURE WORK

A. CONCLUSIONS

Group context-aware applications can help warfighters understand their own state as well as the state of the group enhancing SSA. Our own military experiences show group context-aware applications can benefit warfighters operating in tactical environments. However, group context-aware applications are still in early development. To be effective, most context-aware applications require frequent or continuous context sensing that can be resource intensive. Therefore, efficient management of resources is essential for warfighters who are often deployed with limited resources.

While energy-efficiency for sensing has received some attention, energy-efficient management strategies for groups have not been widely explored.

Several energy-efficiency strategies exist to reduce energy consumption.

GEM is a conceptual platform that can be used to design group context-aware applications. GEM enhances overall energy savings by optimizing the energy resources of devices throughout the group. Implementing a group manager to monitor resources for each group member prevents premature device failures due to resource exhaustion and maximizes group resources.

Resource issues dealing with group context-aware applications on mobile devices will only be resolved by building smart energy-efficiency strategies. The main contribution of our research was to provide a framework for optimizing energy for a group of devices as they provide context information for the group based on sensors embedded in individual devices.

B. AREAS FOR FURTHER RESEARCH

This research was a step towards energy efficient group context-aware applications. GEM supports reduced energy consumption and resource integration; GEM identifies resource consumption issues and presents solutions to address identified issues. Three recommendations are made for future work:

1. Explore Energy Costs of Communication Mechanisms

A key component to the GEM framework is the Communication Manager. As communication mechanisms update, the group manager must identify the lowest cost mechanism to communicate to group members. However, distance between group members can create delays. The number of hops and distance may need to be factored into communication mechanism cost estimates. Studying the total energy cost and delays resulting from communication mechanisms can help derive optimal group performance.

2. Determine Maximum Group Size

The architecture of GEM allows numerous subgroups within a group. However, as the group size grows, the number of hops and distance between devices increases. The result is that communication delays could be significant. Test and evaluation of the recommended size of a group must be conducted to identify how dispersed a group can be before GEM sees a negative impact or diminished savings on energy.

3. Determine Best Energy Management Strategy of Individual Devices Operating Within a Group

Warfighters operate in dynamic environments, and military context may rapidly change. This is compounded by the necessary for real-time, accurate context information. It is established that continuous sensing drains mobile device's battery. Using an energy-efficient strategy that predicts group actions can be advantageous to controlling context queries. Context triggered actions specify how the context-aware application should adapt. Numerous strategies can decrease overall energy consumption of individual devices. However, it has not been determined which strategy or strategies maximize energy consumption of mobile devices used in tactical operations. Therefore, further exploration of energy consumption in support of group context-aware applications should be performed to assess the actual resource consumption impact of these applications.

4. Test and Evaluate the Model

The most obvious area for future research is to develop a prototype to test and evaluate the performance of GEM in a real scenario. The framework and case study here are limited in scope. There are some applications, such as ISE, underway that enable group context-awareness. However, most applications are in the infancy stage.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. SENST SENSOR SUBSTITUTION ALGORITHM

Algorithm 1. *Sensor substitution*

Input:	$\mathcal{S}_U^{KE}$	// set of available sensors
	KE	// requested context element
	AV	// minimal accepted appropriateness
Output:	W^{KE}	// Context element value
	FALSE	// Notification in case of an error

def *get_context_element_value_with_substitution(KE,* $\mathcal{S}_U^{KE}$*, AV)*
min_e_sensor=NULL // variable for storing "best" sensor
min_e=-1 // variable for keeping minimal required amount of energy
for each $S \in \mathcal{S}_U^{KE}$ **do** // loop through all alternatives
  if $((min\_e == -1 \vee (min\_e > e(S)) \wedge g(S) \leq AV))$
    then $\{min\_e\_sensor{=}S;\ min\_e{=}e(S);\}$
if *min_e==-1* **then** return(*FALSE*) **else** return($w(v(min\_e\_sensor))$)

Source: Nath. Suman. "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing." *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. SENST SENSOR TRIGGERING ALGORITHM

Algorithm 2. *Sensor triggering using polling*

Input:	$\mathcal{S}_U^{KE}$	// set of available sensors
	$\mathcal{S}_U^{T+KE}$	// available positive triggers
	$\mathcal{S}_U^{T\text{-}KE}$	// available negative triggers
	KE	// requested context element
	x	// time to wait between loop cycles
Output:	W^{KE}	// Context element value
	FALSE	// Notification in case of an error

def *get_context_element_value_with_polling*(KE, $\mathcal{S}_U^{KE}$, $\mathcal{S}_U^{T+KE}$, $\mathcal{S}_U^{T\text{-}KE}$, x)
min_e_sensor=NULL // variable for storing "best" positiv triggering sensor
min_e=-1 // variable for keeping minimal required amount of energy
for each $ST \in \mathcal{S}_U^{T+KE}$ **do** // loop through all alternatives
    if $((min\_e == -1 \lor min\_e > e(ST)))$
        then $\{min\_e\_sensor{=}ST;\ min\_e{=}e(ST);\}$
if *min_e==-1* **then** return(*FALSE*) // no positiv trigger available
else
    trigger_value=v(*min_e_sensor*) // remember initial value
    while *TRUE* **do** // start loop
        wait x seconds
        if $(trigger\_value <<>> v(ST)) \land (\neg\exists NST \in \mathcal{S}_U^{T\text{-}KE} | v(NST){==}TRUE)$
            MEASURE AND ANNOUNCE
        trigger_value=v(*min_e_sensor*) // remember current value
    done

Source: Schirmer, Maximilian and Hagen Höpfner. "SENST: Approaches for Reducing the Consumption of Smartphone-Based Context Recognition." *CONTEXT 2011* LNAI 6967 (2011): 250–263

.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. SENST RECONSTRUCTION ROUTES

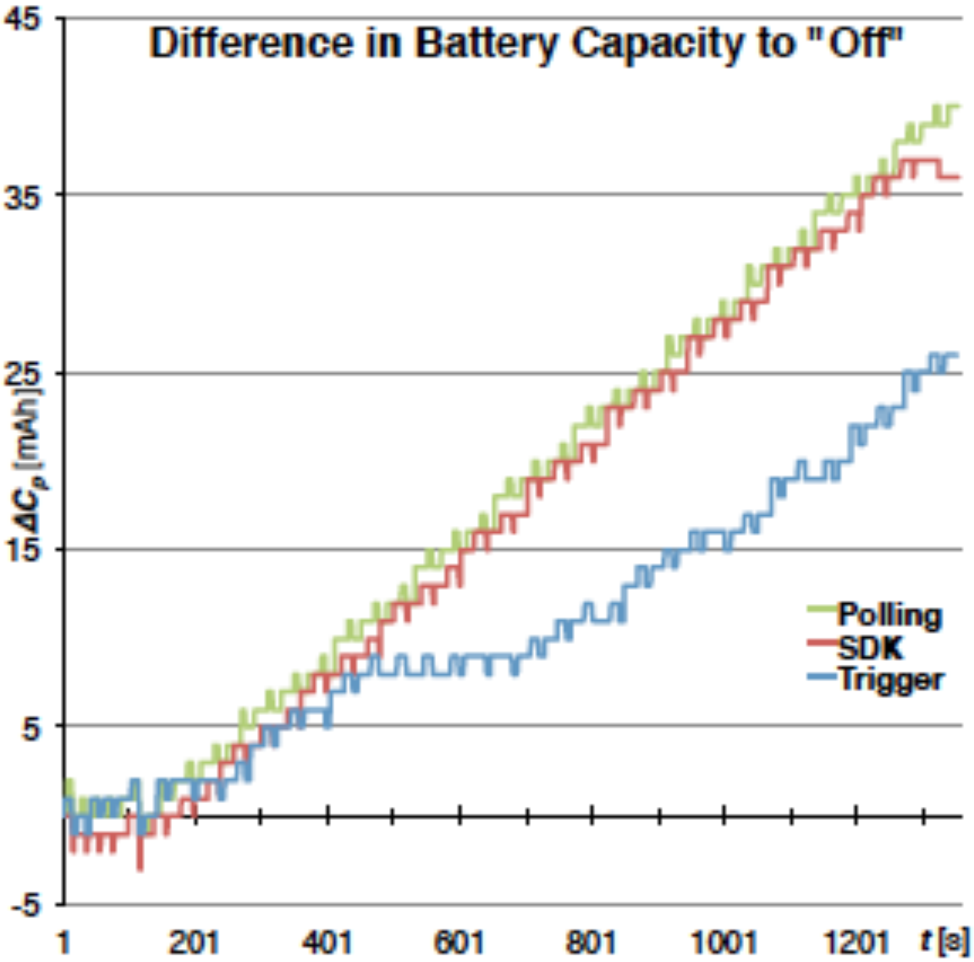
(a) Polling route. (b) SDK route. (c) Triggering route.

In this figure, a red line between coordinates indicates the route; green circles illustrate the accuracy at the coordinate point. The polling and SDK methods produce overlapping coordinates at several positions. Although the triggering method acquires fewer coordinates, it shows few overlaps and includes the most important turns.

Source: Schirmer, Maximilian and Hagen Höpfner. "SENST: Approaches for Reducing the Consumption of Smartphone-Based Context Recognition." *CONTEXT 2011* LNAI 6967 (2011): 250–263

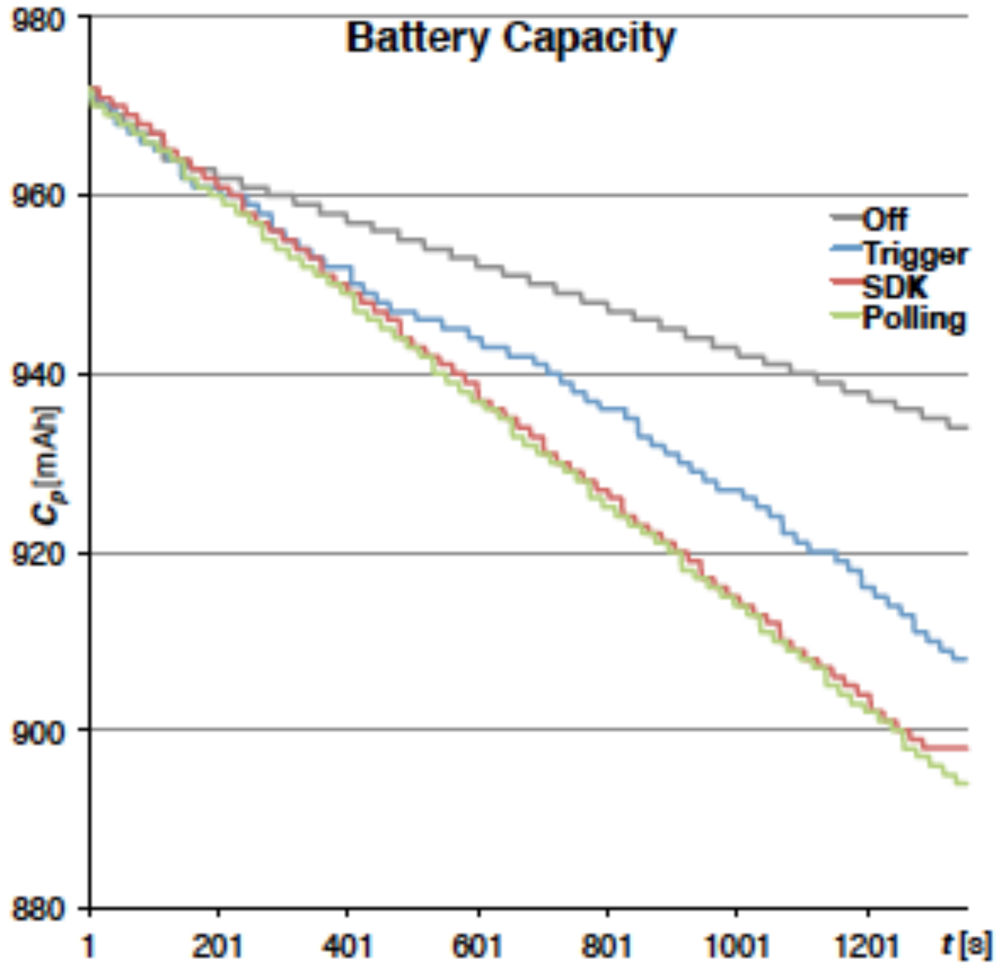
THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. SENST ENERGY EVALUATION CHARTS

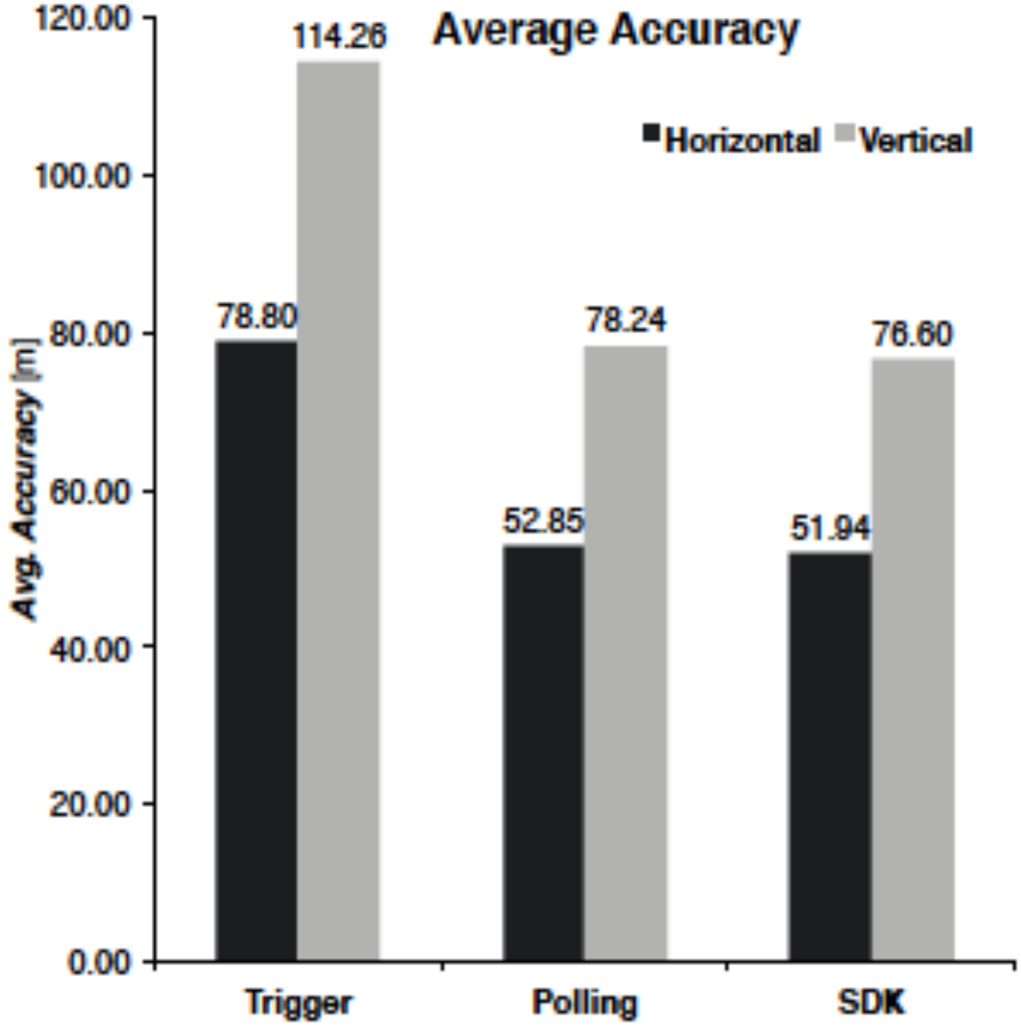


Remaining battery capacity at a time *t* from the baseline condition "Off." This figure indicates that the difference in power consumption of the SDK and polling methods grow linear over time. However, when no GPS positioning is requested, the difference between the "Off" condition and the triggering method remain constant.

Source: Schirmer, Maximilian and Hagen Höpfner. "SENST: Approaches for Reducing the Consumption of Smartphone-Based Context Recognition." *CONTEXT 2011* LNAI 6967 (2011): 250–263

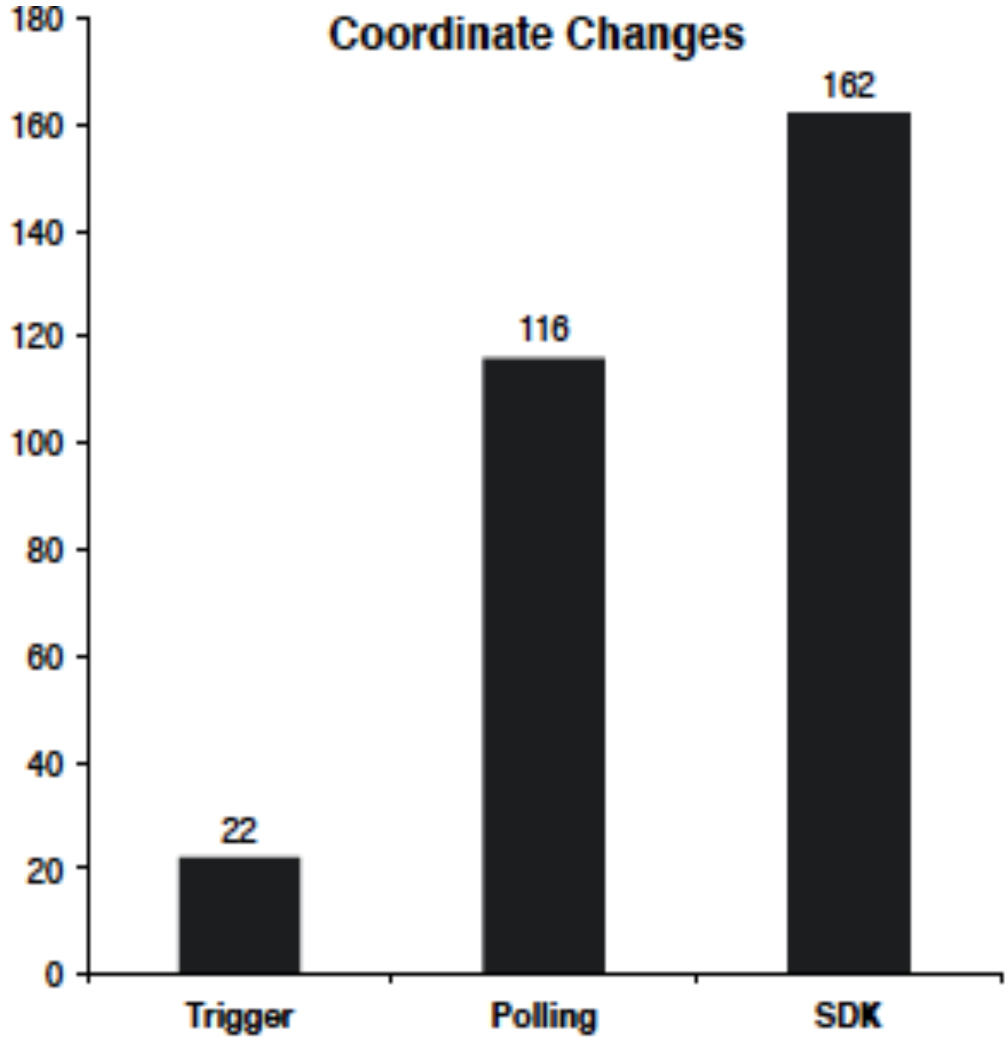

Battery Capacity
980
960
940
920
900
880
C_p [mAh]
1
201
401
601
801
1001
1201
t [s]
Off
Trigger
SDK
Polling

APPENDIX E. SENST ACCURACY CHARTS



Accuracy of acquired geographical location data was compared using horizontal and vertical accuracy.

Source: Schirmer, Maximilian and Hagen Höpfner. "SENST: Approaches for Reducing the Consumption of Smartphone-Based Context Recognition." *CONTEXT 2011* LNAI 6967 (2011): 250–263


Coordinate Changes
180
160
140
120
100
80
60
40
20
0
162
116
22
Trigger
Polling
SDK

APPENDIX F. ACE OPTIMAL SENSING ALGORITHM

Algorithm 1 Exhaustive search for the optimal sensing plan

```
procedure INIT(X)
    target ← X
    trace, next, result, dpCache ← φ

procedure RESULT
    return result

procedure UPDATE(attrib, value)
    trace ← trace ∪ [attrib = value]
    if attrib = target then
        result ← value

procedure NEXT
    if result = φ then
        return  φ
    (attrib, cost) ← NextHelper(trace)
    return attrib

procedure NEXTHELPER(trace)
    if trace is in dpCache then
        [next, cost] ← dpCache[trace]
        return [next, cost]
    minCost ← ∞
    bestAttrib ← φ
    for all State s ∉ trace  do
        traceT ← trace ∪ {s = true}
        if traceT satisfies expressionTree(target = T) then
            CostT ← 0
        else
            [next, CostT] ← NextHelper(traceT)
        traceF ← trace ∪ {s = false}
        if traceF satisfies expressionTree(target = F) then
            CostF ← 0
        else
            [next, CostF] ← NextHelper(traceF)
        ExpctedCost ← Cost(s) + Prob(s = true) · CostT + Prob(s = false) · CostF
        if ExpectedCost < minCost then
            minCost ← ExpectedCost
            bestAttrib ← s
    dpCache[trace] ← [bestAttrib, minCost]
    return [bestAttrib, minCost]
```

Source: Nath. Suman. “ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing.” *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX G. ACE SENSING ORDER ALGORITHM

Algorithm 2 Generating attribute sensing order

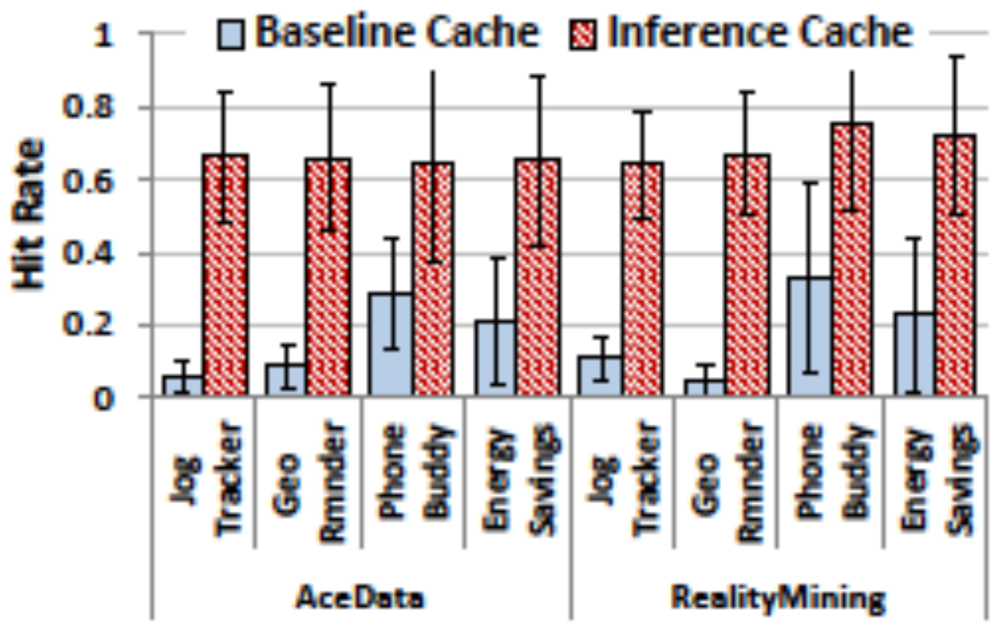
procedure AssignCostAndProb(And-OR Tree Node T)
    if T is a leaf node **then**
        $(\mathcal{C}_T, \mathcal{P}_T) \leftarrow (c_T, p_T)$
        return $(\mathcal{C}_T, \mathcal{P}_T)$
    for all child node N_i of T, $1 \leq i \leq k$ **do**
        $(\mathcal{C}_N, \mathcal{P}_N) \leftarrow$ AssignCostAndProb(N)
        if T is an AND node **then**
            $\mathcal{P}_N \leftarrow 1 - \mathcal{P}_N$
    Sort the child nodes of T in decreasing order of their values of $\mathcal{P}_{N_i}/\mathcal{C}_{N_i}, 1 \leq i \leq k$
    $\mathcal{C}_T \leftarrow \mathcal{C}_{N_1} + \sum_{i=2}^{k} \mathcal{C}_{N_i} \prod_{j=1}^{i-1}(1 - \mathcal{P}_{N_j})$
    if T is an AND node **then**
        $\mathcal{P}_T \leftarrow \prod_{i=1}^{k} \mathcal{P}_{N_i}$
    else
        $\mathcal{P}_T \leftarrow 1 - \prod_{i=1}^{k}(1 - \mathcal{P}_{N_i})$
    return $(\mathcal{C}_T, \mathcal{P}_T)$

procedure FindOrdering(And-Or Tree Node T)
    if T is a leaf node **then**
        $Q.Enqueue(T)$
    for all child N_i of T in ascending order of $\mathcal{P}_{N_i}/\mathcal{C}_{N_i}$ **do**
        FindOrdering(N_i)

Source: Nath. Suman. "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing." *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

THIS PAGE INTENTIONALLY LEFT BLANK

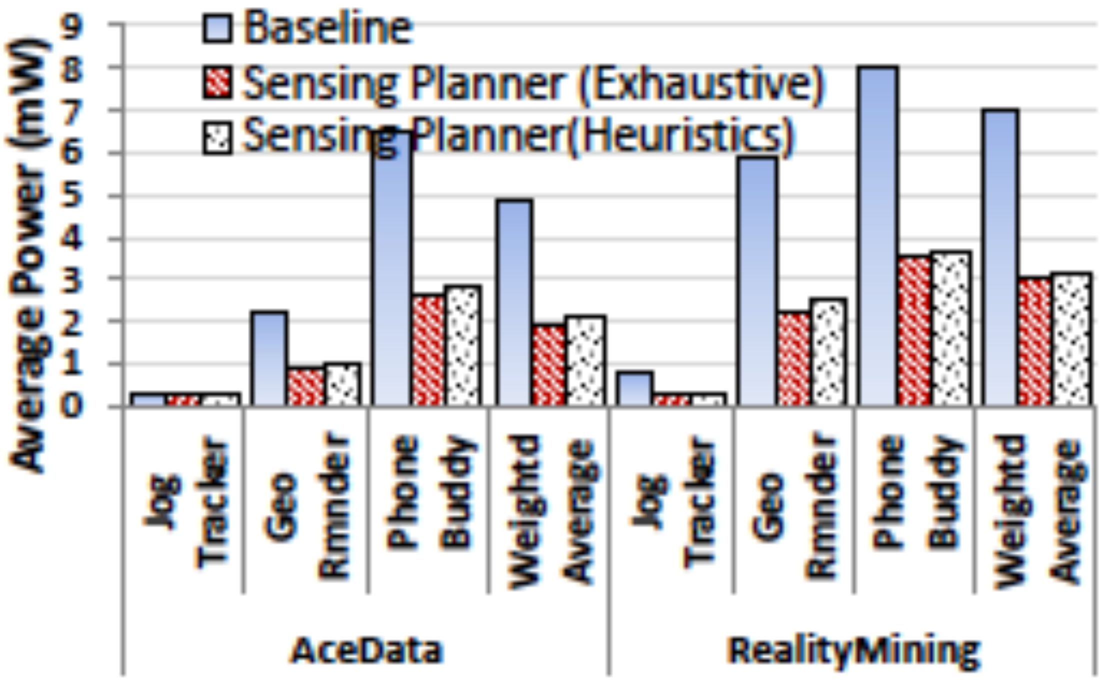
APPENDIX H. HIT RATE AND ENERGY SAVINGS OF INFERENCE CACHE



Source: Nath. Suman. "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing." *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

THIS PAGE INTENTIONALLY LEFT BLANK

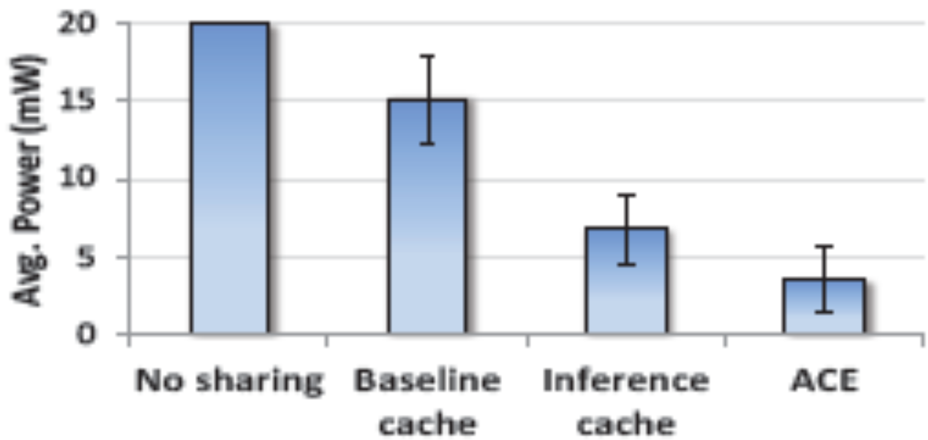
APPENDIX I. ENERGY SAVINGS OF SENSING PLANNER



Source: Nath. Suman. "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing." *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX J. ACE END-TO-END ENERGY SAVINGS



Source: Nath. Suman. "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing." *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Benaskeur, A. and H. Irandoust. *Sensor Management for Tactical Surveillance Operations*. Technical Memorandum 2006-767. Valcartier, Canada: Defence Research and Development Canada, 2006.

Burke, Matthew M. “Lawsuit Expands over Radiation Exposure During Fukushima disaster,” *Stars and Stripes*, March 13, 2013. http://www.stripes.com/news/lawsuit-expands-over-radiation-exposure-during-fukushima-disaster-1.211889.

Clausewitz, Carl V. *On War*. Princeton, NJ: Princeton University Press, 1976.

CNN wire staff. “Anxiety in Japan Grows as Death Toll Steadily Climbs,” *CNN World*, March,14, 2011. http://www.cnn.com/2011/WORLD/asiapcf/03/13/japan.quake/index .html.

Debevc, Matjaž, Andreas Holzinger, and Karl Heinz Struggl. “Applying Model-View-Controller (MVC) in Design and Development of Information Systems: An Example of Smart Assistive Script Breakdown in an e-Business Application.” Paper presented at the proceedings of the 2010 International Conference on e-Business (ICE-B), Athens, Greece, July 26-28, 2010.

Dey, Anind. “Understanding and Using Context.” *Personal and Ubiquitous Computing* 5 no. 1 (2001): 4–7.

Feickert, Andrew and Emma Chanlett-Avery. *Japan 2011 Earthquake: U.S. Department of Defense (DOD) Response*. CRS Report R41690. Washington, DC: Library of Congress, Research Service, June 2, 2011.

Ferscha, Alois, Clemens Holzmann, and Stefan Oppl. “Context Awareness for Group Interaction Support.” Paper presented at the annual meeting for the International Workshop on Mobility Management and Wireless Access Protocols, Philadelphia, PA, October 1, 2004.

Lewis, Grace, Marc Novakouski, and Enrique Sánchez. “A Reference Architecture for Group-Context-Aware Mobile Applications.” *Mobile Computing, Applications, and Services* 2012 LNIST 110 (2013): 44–63.

McNerney, Michael J. "Stabilization and Reconstruction in Afghanistan: Are PRTs a Model Or a Muddle?" *Parameters* 35, no. 4 (05, 2006): 32–46, http://search.proquest.com/docview/198173983?accountid=12702.

Nath. Suman. "ACE: Exploiting Correlation for Energy-Efficient and Continuous Context Sensing." *IEEE Transactions on Mobile Computing* 12 (August 2013): 1472–1486.

Parrish, Karren. "U.S. Leaders Recall Japan Disasters, Relief Efforts." *American Forces Press Service.* March 11, 2012. http://www.army.mil/article/75490/U_S__leaders_recall_Japan_disasters__relief_efforts/.

Pauly Jr., Robert J. and Robert W. Redding. "Denying Terrorists Sanctuary through Civil Military Operations." in *Countering Terrorism and Insurgency in the 21st Century,* edited by James Forest, 273−297. Westport, CT: Praeger, 2007.

Schirmer, Maximilian and Hagen Höpfner. "SENST: Approaches for Reducing the Consumption of Smartphone-Based Context Recognition." *CONTEXT 2011* LNAI 6967 (2011): 250–263.

United Nations Office for the Coordination of Humanitarian Affairs (OCHA*). Japan • Earthquake & Tsunami*. Situation Report No. 7. United Nations OCHA, Asia Pacific: OCHA, 2011.

United States Geological Survey (USGS). "Magnitude 9.0 - Near the East Coast of Honshu, Japan." USGS. March 11, 2011. http://earthquake.usgs.gov/earthquakes/eqinthenews/2011/usc0001xgp/#summary.

———. "USGS Updates Magnitude of Japan's 2011 Tohoku Earthquake to 9.0." USGS. March 14, 2011. http://www.usgs.gov/newsroom/article.asp?ID=2727&from=rss_home.

Weiser, Mark. "The Computer for the Twenty-first Century." *Scientific American* 265 (1991): 94–104.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California