Online Material
Companion to *Architectural Tactics for Cyber-Foraging: Results of a Systematic Literature Review*
Journal of Systems and Software

A Systematic Literature Review on Architectures for Cyber-Foraging Systems

Grace A. Lewis and Patricia Lago
VU University Amsterdam

May 26, 2015

Abstract

Mobile devices have become for many the preferred way of interacting with the Internet, social media and the enterprise. However, mobile devices still do not have the computing power and battery life that will allow them to perform effectively over long periods of time, or for executing applications that require extensive communication, computation, or low latency. Cyber-foraging is a technique to enable mobile devices to extend their computing power and storage by offloading computation or data to more powerful servers located in the cloud or in single-hop proximity. This article presents the protocol that was followed in the execution of a systematic literature review (SLR) on software architectures that support cyber-foraging. It is an online companion for the article titled Architectural Tactics for Cyber-Foraging: Results of a Systematic Literature Review published by the Journal of Systems and Software.

1 Introduction

Mobile Cloud Computing (MCC) refers to the combination of mobile devices and cloud computing in which cloud resources perform computing-intensive tasks and store massive amounts of data [1]. Increased mobile device capabilities, combined with better network coverage and speeds, have enabled MCC such that mobile devices have become for many the preferred form for interacting with the Internet, social media, and the enterprise. However, mobile devices still offer less computational power than conventional desktop or server computers, and limited battery life remains a problem especially for computation- and communication-intensive applications.

Cyber-foraging [2] is an area of work within MCC that leverages external resources (i.e., cloud or local servers; the latter often called surrogates) to augment the computation and storage capabilities of resource-limited mobile devices while extending their battery life. There are two main forms of cyber-foraging. One is computation offload, which is the offload of expensive computation in order to extend battery life and increase computational capability. The second is data staging to improve data transfers between mobile devices and the cloud by temporarily staging data in transit.

The goal of this article is to present the details of the research protocol that was followed in the Systematic Literature Review (SLR) to discover software architectures that support cyber-foraging.

2 Research Protocol

To identify work related to architectures for cyber-foraging an SLR was conducted following the guidelines proposed in [3] and [4]. The research question, search strategy, inclusion and exclusion criteria, and validation method are presented in the following subsections.

2.1 Research Question

The goal of the SLR is to identify work in cyber-foraging with a software architecture perspective. To achieve this goal, the following research question is defined:

What software architecture and design strategies for cyber-foraging from mobile devices can be identified in the literature?

2.2 Search Strategy

Three main keywords can be built from the research question: cyber-foraging, mobile devices, and software architecture. Each of these keywords has a set of related synonyms and alternative spellings. Based on these keywords and their related terms the following basic search string was defined:

(cyber foraging OR cyber-foraging OR code offload OR code offloading OR computation offload OR computation offloading OR data offload OR data staging) AND (mobile OR handheld OR smartphone) AND (software architecture OR software design OR system architecture)

The main data source was Google Scholar.[1] Snowballing was used to complement the set of primary studies. The advantage of using Google Scholar was that it included studies that are outside of software engineering, such as computer engineering and computer science, which is where many of the studies on cyber-foraging currently come from. The downside is that it returns many

[1] `http://scholar.google.com/`

results which are irrelevant because it performs a full-text search and because there is no control process that ensures that all results are valid (i.e., are academic or industrial publications). To make sure that all relevant studies were identified, the dates were left open even though the term cyber-foraging was coined in 2001.

Details of each study were recorded using JabRef.[2] Separate JabRef databases were created for each round of the primary study identification process.

2.3 Inclusion and Exclusion Criteria

The inclusion and exclusion criteria shown in Table 1 were defined and applied to the search results.

2.4 Validation

The protocol was validated by executing it on Google Scholar without snowballing. The goal was to determine if it was rigorous enough and to improve it where necessary. The results of multiple iterations of the search string were checked against a set of 17 known relevant studies in the field of cyber-foraging. This set was validated by an expert in the field. The search string was adjusted accordingly until it returned all 17 relevant studies either directly or as one of the references (first-level snowballing). The inclusion/exclusion criteria were reviewed during the process to ensure that the results were representative of software architecture and design of cyber-foraging systems.

Table 1: Inclusion and Exclusion Criteria

Inclusion Criteria	**Exclusion Criteria**
A study that proposes software architectures for cyber-foraging. *Rationale:* We are interested in studies that present software architecture and design of cyber-foraging systems in which mobile components, surrogate/server components, and offload elements are clearly defined. *Example:* A study that presents the software architecture and design of a cyber-foraging system for both the mobile as well as the surrogate/server and clearly defines what computation or data is being offloaded.	A study that proposes a cyber-foraging system that does not present software architecture and design details. *Rationale:* If the study does not present architecture and design details, it does not contain information that can be abstracted into general architecture patterns and tactics. *Examples:* A study that presents a cyber-foraging solution that discusses only the benefits of the solution and does not contain software architecture details will not be included. A study that surveys cyber-foraging solutions that have already been presented in other studies and does not propose a new cyber-foraging solution will not be included. A study that only discusses an offload algorithm and not a complete solution for cyber-foraging will not be included.

Continued on next page

[2] `http://jabref.sourceforge.net/`

Table 1 – *Continued from previous page*

Inclusion Criteria	**Exclusion Criteria**
A study that proposes a cyber-foraging system for computation offload or data staging in which the mobile device is augmenting its computing power by using surrogates such as cloud resources. *Rationale:* A cyber-foraging system leverages surrogates to perform computation that would make sense to execute locally but if executed on the mobile device would drain resources or not provide adequate performance, or to stage data in transit to and from cloud resources and mobile devices. *Example:* A study that presents a cyber-foraging solution that uses surrogates to offload expensive computation or to store data temporarily until centralized resources become available.	A study that proposes a system in which mobile devices simply access cloud services or in which computation is partitioned across similar nodes. *Rationale:* A system that simply uses cloud services as parts of its functionality or that distributes computation among other mobile devices is not a case of cyber-foraging because it is not leveraging a more powerful surrogate to extend its computing power. *Example:* A study that presents a mobile cloud solution in which cloud services are accessed from mobile devices simply to fulfill part of its functionality or a study that represents distribution of computation across a mobile ad hoc network (MANET).
A study that proposes solutions based on open technologies that contain enough detail to abstract the main software architecture components. *Rationale:* Studies that rely on open technologies are more likely to present solution details. *Example:* A study that presents software architecture views for a cyber-foraging solution that relies on open or readily-available technologies will be included.	A study proposed by a commercial vendor or that relies on proprietary hardware or network protocols. *Rationale:* Studies produced by vendors are unlikely to contain architecture information because it is part of their intellectual property. In addition, characteristics of solutions that rely on specific hardware or protocols will not be able to be abstracted into general architecture patterns and strategies. *Example:* A study that presents a cyber-foraging solution that only works if connected to a vendor's network or that requires special hardware, networking devices or protocols for communication will not be included.
A study that is in the form of a published scientific paper or industrial publication. *Rationale:* A scientific paper focuses on scientific content and follows a process to guarantee a good level of quality. Also, as solutions may have been devised by industrial organizations, broader industrial publications describing such solutions should be included. *Examples:* A study in a refereed journal that is part of a conference or a technical report that follows a standard publication template (i.e., abstract, introduction, description of the problem, proposed solutions, related work and references), a PhD or Masters thesis, or a study in an industrial publication that presents details of a cyber-foraging system or architecture will be included.	A study that is not in the form of a published scientific paper or that is in an industrial publication but only focuses on the commercial benefits of the solution. *Rationale:* Lack of scientific content and rigorous methods can lead to a low-quality outcome. In addition, studies in industrial publications targeted at increasing sales and that only highlight benefits of the solution do not add scientific value to the outcome of the review. *Examples:* Papers that have not been published, scientific papers that do not follow a standard publication template, keynote summaries, tables of contents, collections of abstracts, workshop summaries, project proposals, slide sets, and commercial product brochures will not be included.

3 Identification of Primary Studies

3.1 Round 1

The search string was last entered in Google Scholar on September 17, 2013 and returned 430 results. The complete list of results is available as online material at `http://www.andrew.cmu.edu/user/gritter/InitialStudies-CyberForaging.html`. The studies were evaluated against the inclusion and exclusion criteria *based on the title, abstract, keywords and an initial scan of the study*. The results are shown in Table 2.

Table 2: Round 1 Results

Result	Studies	Description
Yes	91	Studies that met the inclusion and exclusion criteria based on the title, abstract, keywords and an initial scan of the study
No	297	Studies that did not meet the inclusion and exclusion criteria based on the title, abstract, keywords and an initial scan of the study
Maybe	23	Studies that did not fully meet the inclusion criteria based on the title, abstract, keywords and an initial scan of the study, but that warranted a full read due to the coverage of software architecture
Duplicate	18	Studies that were identical to other studies or were a subset of a larger study by the same author(s) (e.g., a paper that was cross-listed or a paper that is explicitly a chapter of a PhD or Masters thesis, in which case we included the thesis because it is the superset)
Plagiarism	1	Study that was copied from a conference paper that we co-authored in 2013.
TOTAL	**430**	
TOTAL FOR ROUND 2	**114**	Studies with Result = Yes and Result = Maybe

3.2 Round 2

The studies with Result = Yes and Result = Maybe from Round 1 were *fully read* and evaluated against the inclusion and exclusion criteria. The list of studies evaluated in Round 2 is available as online material at `http://www.andrew.cmu.edu/user/gritter/Round2Studies-CyberForaging.html`. The results of the evaluation are shown in Table 3.

Table 3: Round 2 Results

Result	Studies	Description
Yes	50	Studies that met the inclusion and exclusion criteria based on fully reading the study
No	62	Studies that did not meet the inclusion and exclusion criteria after reading the study in full
Duplicate	12	Studies that were a subset of a larger study by the same author(s) (e.g., a paper that after a full read was determined to be part of a PhD or Masters thesis or a shorter version of a study that reports the same results from a software architecture perspective)
TOTAL	**114**	

3.3 Final Round

The references in each study with Result = Yes from Round 2 were evaluated against the inclusion criteria based on title, abstract and keywords as an initial round of snowballing. Those that passed based on this initial scan were fully read and included if they fully met the inclusion criteria. The results are shown in Table 4.

Table 4: Final Round Results

Result	Studies	Description
Direct	50	Studies with Result = Yes from Round 2
Snowballing	8	Studies that correspond to references in the *Direct* results that met the inclusion and exclusion criteria based on fully reading the study
TOTAL	**58**	

The list of 58 primary studies is presented in Table 5. The *Primary Study* column contains the reference for the study. The *Type* column is the type of study which can be BC (Book Chapter), CP (Conference Paper), DD (Doctoral Dissertation), JA (Journal Article), MT (Masters Thesis), or TR (Technical Report). *System Name* refers to the name of the cyber-foraging system that is described in the study. The *Form* is the form of cyber-foraging which can be CO (Computation Offload) or (DS = Data Staging). The *Domain or Use Case* refers to the targeted domain or use case for the system. Finally, the *Source* column is the source of the study which is either GS (Google Scholar) or S (Snowballing).

Table 5: Primary Studies

Primary Study	Type	System Name	Form	Domain or Use Case	Source
Ahnn2013 [5]	JA	mHealthMon	CO	Healthcare	GS
Angin2013 [6]	JA	Mobile Agents	CO	Java applications	GS
Armstrong2006 [7]	CP	Edge Proxy	DS	Web page updates	GS

Continued on next page

Table 5 – *Continued from previous page*

Primary Study	Type	System Name	Form	Domain or Use Case	Source
Aucinas2012 [8]	CP	Clone-to-Clone (C2C)	CO	Intelligent transport systems, Mobile multiplayer online games	GS
Bahrami2006 [9]	CP	Mobile Information Access Architecture for Occasionally-Connected Computing	DS	Occasionally-connected operations	GS
Balan2007 [10]	CP	Chroma	CO	Mobile interactive resource-intensive applications	S
Chang2011 [11]	JA	Collaborative Applications	CO	Speech recognition	GS
Chen2004 [12]	JA	Computation and Compilation Offload	CO	Image and video processing	GS
Cheng2013 [13]	TR	Cloud Media Services	CO	Hybrid Broadcast Broadband TV (HBB-TV)	GS
Chu2004 [14]	JA	Roam	CO	Seamless applications	GS
Chun2009 [15]	CP	CloneCloud	CO	Mobile applications in general	GS
Cuervo2012 [16]	DD	MAUI (Mobile Assistance Using Infrastructure)	CO	Operations that consume and produce small amounts of information compared to their computational requirements	GS
		Kahawai	CO	Graphics applications that require high-end GPU rendering	

Continued on next page

Table 5 – *Continued from previous page*

Primary Study	Type	System Name	Form	Domain or Use Case	Source
Duga2011 [17]	MT	HPC-as-a-Service	CO	High-Performance Computing (HPC)	GS
Endt2011 [18]	BC	OpenCL-Enabled Kernels	CO	Automotive	GS
Esteves2011 [19]	CP	Real Options Analysis	CO	Mobile applications in general	GS
Fjellheim2005 [20]	JA	3DMA	CO	Context-aware applications	GS
Flinn2002 [21]	CP	Spectra	CO	Mobile interactive resource-intensive applications	S
Flinn2003 [22]	CP	Trusted and Unmanaged Data Staging Surrogates	DS	Distributed filesystems	GS
Giurgiu2009 [23]	CP	AlfredO	CO	Typical three-tiered applications implemented as OSGi[3] bundles for each tier	S
Goyal2011 [24]	DD	Collective Surrogates	CO	Mobile applications in general	GS
Guan2008 [25]	DD	Grid-enhanced mobile devices	CO	Ambient intelligence	GS
Ha2011 [26]	TR	Cloudlets	CO	Computation-intensive applications in hostile environments	GS
Hung2011 [27]	JA	Virtual Phone	CO	Mobile applications in general	GS
Imai2012 [28]	MT	Single-Server Offloading	CO	Moderately-slow, single-purpose, computation-intensive applications	GS

Continued on next page

[3] The Open Service Gateway Initiative, or OSGi, is a specification and Java framework for developing and dynamically deploying modular software programs and libraries (`http://www.osgi.org`).

Table 5 – *Continued from previous page*

Primary Study	Type	System Name	Form	Domain or Use Case	Source
		Cloud Operating System to Support Multi-Server Offloading	CO	Very computation-intensive mobile applications	
Iyer2012 [29]	CP	Android Extensions	CO, DS	Mobile applications that handle complex computations or large amounts of data	GS
Jarabek2012 [30]	CP	ThinAV	CO	Anti-malware scanning	GS
Kemp2012 [31]	CP	Cuckoo	CO	Mobile applications in general	GS
Kosta2012 [32]	CP	ThinkAir	CO	Mobile applications in general	GS
Kovachev2012 [33]	JA	MACS (Mobile Augmentation Cloud Services)	CO	Mobile applications in general	GS
Kristensen2010 [34]	DD	Scavenger	CO	Image manipulation, continuous speech recognition, augmented reality	GS
Kundu2007 [35]	JA	Telemedik	DS	Healthcare	GS
Kwon2013 [36]	CP	AMCO (Adaptive, Multitarget Cloud Offloading)	CO	Java applications	GS
Lee2012 [37]	BC	MCo	CO	Java applications	GS
Matthews2011 [38]	CP	PowerSense	CO	Telemedicine (Image Processing for Dengue Detection)	GS
Messer2002 [39]	CP	AIDE	CO	Java applications	GS
Messinger2013 [40]	TR	Application Virtualization on Cloudlets	CO	Mobile applications in general	GS

Continued on next page

Table 5 – *Continued from previous page*

Primary Study	**Type**	**System Name**	**Form**	**Domain or Use Case**	**Source**
Mohapatra2003 [41]	TR	PARM	CO	Mobile applications in general	GS
Ok2007 [42]	CP	Resource Furnishing System	CO	Computation-intensive mobile applications	GS
OSullivan2013 [43]	CP	Cloud Personal Assistant (CPA)	CO	Cloud Services	GS
Park2012 [44]	CP	SOME (Selective Offloading for a Mobile computing Environment)	CO	HTML5 web applications	S
Phokas2013 [45]	CP	Feel The World (FTW)	DS	Participatory sensing applications	GS
Pu2013 [46]	CP	SmartVirtCloud (SmartVC)	CO	Mobile applications in general	GS
Ra2011 [47]	CP	Odessa	CO	Mobile interactive perception applications	S
Rachuri2012 [48]	DD	Smartphone-based social sensing	CO	Social sensing applications	GS
Rahimi2012 [49]	CP	MAPCloud	CO	Rich mobile applications	GS
Satyanarayanan2009 [50]	JA	VM-Based Cloudlets	CO	Computation-intensive mobile applications	S
Shi2013 [51]	TR	IC-Cloud	CO	Mobile applications in general	GS
Silva2008 [52]	CP	SPADE	CO	Mobile applications that perform lengthy tasks	GS
Su2005 [53]	CP	Slingshot	CO	Computation-intensive mobile applications	S
Verbelen2012 [54]	JA	AIOLOS	CO	Complex multimedia applications	S

Continued on next page

Table 5 – *Continued from previous page*

Primary Study	Type	System Name	Form	Domain or Use Case	Source
Xiao2013 [55]	CP	Large-Scale Mobile Crowdsensing	DS	Crowdsensing applications	GS
Yang2008 [56]	JA	Offloading Toolkit and Service	CO	Java applications	GS
Yang2012 [57]	TR	Sonora	DS	Continuous data streams	GS
Yang2013 [1]	JA	Mobile Data Stream Application Framework	CO	Data stream applications	GS
Zhang2009 [58]	CP	Heterogeneous Auto-Offloading Framework for Mobile Web Browsers	CO	Web pages with multimedia content	GS
Zhang2011 [59]	JA	Weblets	CO	Web applications	GS
Zhang2012 [60]	JA	DPartne	CO	Java applications	GS
Zhang2012a [61]	CP	Elastic HTML5	CO	Web applications	GS

4 Categorization of Primary Studies

4.1 Studies Per Type

As shown in Figure 1, most of the primary studies are papers published in conference proceedings (28) followed by journal articles (15). Even though the scope of the search included industry reports, of the six studies identified as Technical Reports, only one comes from industry. The others are from universities (4) and an FP7 project (1). In addition, there were two book chapters, two Masters Theses and five Doctoral Dissertations. This distribution shows that even though the topic is of potential interest to industry, most of the published work in this area comes from academia.

4.2 Studies Per Year

As shown in Figure 2, the number of primary studies per year has grown since the first study dated 2002. This shows that it is indeed a topic of interest, especially in the last three years.

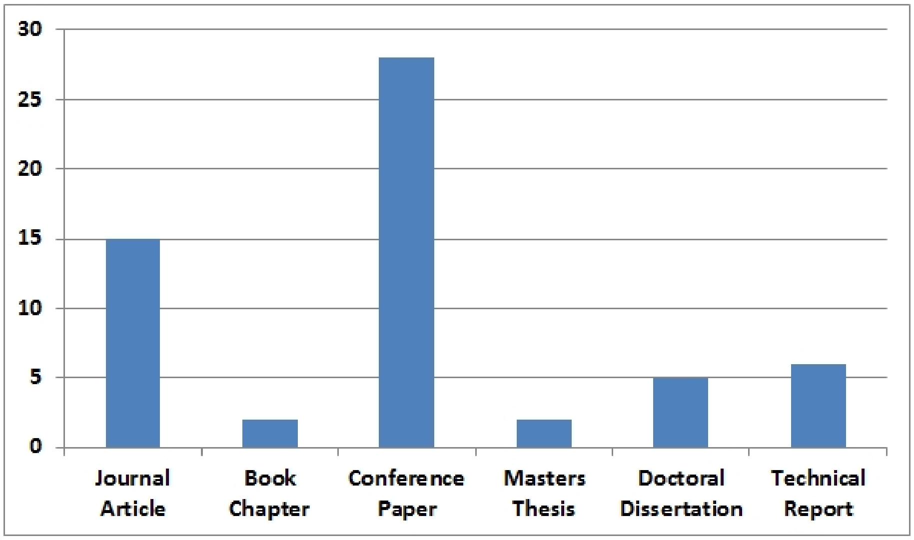

Figure 1: Number of Primary Studies Per Type

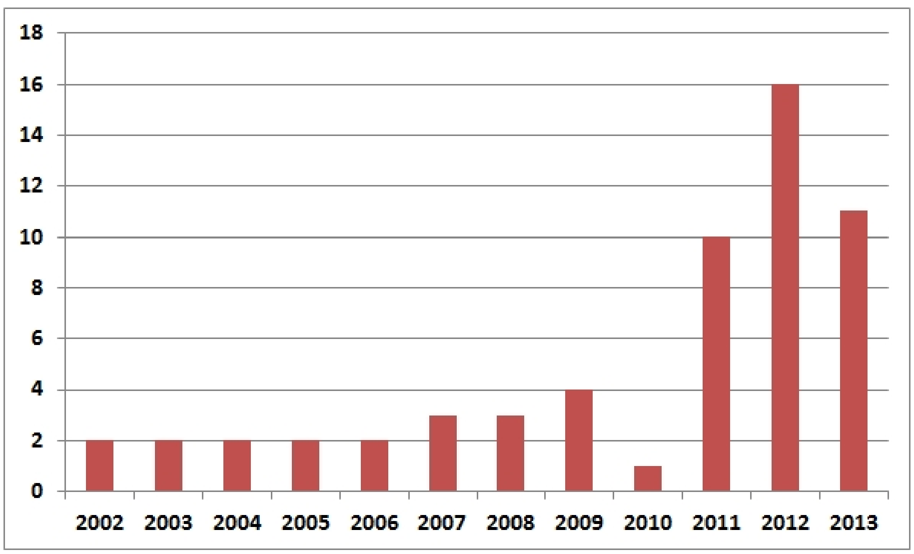

Figure 2: Number of Primary Studies Per Year

5 Threats to Validity

Google Scholar was the single data source for the primary studies, and was complemented by snowballing. The search string was adjusted until it returned the set of studies that was identified by an expert in the field as the set of seminal studies (either direct results or in the references). However, the problem is that if a study is not listed in Google Scholar it will not be returned in the results.

For example, Google Scholar returned [15], which is one of the seminal studies, but did not return a later study on the same system [62].

In addition, the term software architecture (which is part of the search string) was not widely used until the mid 2000s, which is reflected in the years of the studies and Figure 2. This was mitigated by snowballing.

6 Related Work

There are several studies that survey the field of mobile cloud computing and identify cyber-foraging as a research area and challenge, but are not systematic literature reviews and do not have an architecture focus. Abolfazli et al [63] present a survey of cloud-based mobile augmentation (CMA) approaches, one of which is cyber-foraging. One of the challenges stated by this work is the lack of a reference architecture for CMA. Dinh at al [64] present a survey on mobile cloud computing (MCC). Computation offload is discussed as a technique for extending battery lifetime of mobile devices and listed as one of the challenges for MCC. Fernando et al [65] present a more complete survey on MCC. Some of the research that addresses efficient computation offload and distribution to the cloud and how it differs from traditional distributed systems is discussed in this paper. Lomotey at al [66] present an additional survey on MCC and start introducing some of the challenges of ubiquitous cloud computing (UCC), defined as defined as consistency in cloud service access from multiple mobile devices owned by a single user. Computational offloading from mobile nodes to middle-tier servers (i.e., surrogates) is mentioned as one way to overcome energy and latency limitations of offloading to remote clouds in this paper. Kumar et al [67] present a survey on computation offloading but focus primarily on the algorithms used to partition and offload programs in order to improve performance or save energy. Finally, Yu et al [68] present a survey on seamless application mobility, which is the continuous or uninterrupted computing experience as a user moves across devices. Code offloading is mentioned as a future direction for seamless application mobility.

The work that is most similar to ours is by Flinn et al [69] that presents a discussion of representative cyber-foraging systems and their characteristics. However, it is limited to a small number of systems and does not follow a systematic process. To the best of our knowledge, ours is the first systematic literature review related to architectures for cyber-foraging.

References

[1] L. Yang, J. Cao, Y. Yuan, T. Li, A. Han, A. Chan, A framework for partitioning and execution of data stream applications in mobile cloud computing, ACM SIGMETRICS Performance Evaluation Review 40 (4) (2013) 23–32.

[2] M. Satyanarayanan, Pervasive computing: vision and challenges, Personal Communications, IEEE 8 (4) (2001) 10–17.

[3] T. Dyba, T. Dingsoyr, G. Hanssen, Applying systematic reviews to diverse study types: An experience report, in: Empirical Software Engineering and Measurement, 2007. ESEM 2007. First International Symposium on, 2007, pp. 225–234.

[4] B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, Tech. Rep. EBSE 2007-001, Keele University and Durham University Joint Report (2007).

[5] J. Ahnn, M. Potkonjak, mhealthmon: Toward energy-efficient and distributed mobile health monitoring using parallel offloading, Journal of Medical Systems 37 (5) (2013) 1–11.

[6] P. Angin, B. Bhargava, An agent-based optimization framework for mobile-cloud computing, Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA) 4 (2013) 1–17.

[7] T. Armstrong, O. Trescases, C. Amza, E. de Lara, Efficient and transparent dynamic content updates for mobile clients, in: Proceedings of the 4th international conference on Mobile systems, applications and services, ACM, 2006, pp. 56–68.

[8] A. Aucinas, J. Crowcroft, P. Hui, Energy efficient mobile M2M communications, in: Proceedings of ExtremeCom '12, 2012.

[9] A. Bahrami, C. Wang, J. Yuan, A. Hunt, The workflow based architecture for mobile information access in occasionally connected computing, in: Services Computing, 2006. SCC'06. IEEE International Conference on, IEEE, 2006, pp. 406–413.

[10] R. K. Balan, D. Gergle, M. Satyanarayanan, J. Herbsleb, Simplifying cyber foraging for mobile devices, in: Proceedings of the 5th international conference on Mobile systems, applications and services, MobiSys '07, ACM, New York, NY, USA, 2007, pp. 272–285.

[11] Y.-S. Chang, S.-H. Hung, Developing collaborative applications with mobile cloud-a case study of speech recognition, Journal of Internet Services and Information Security (JISIS) 1 (1) (2011) 18–36.

[12] G. Chen, B.-T. Kang, M. Kandemir, N. Vijaykrishnan, M. J. Irwin, R. Chandramouli, Studying energy trade offs in offloading computation/compilation in java-enabled mobile devices, Parallel and Distributed Systems, IEEE Transactions on 15 (9) (2004) 795–809.

[13] B. Cheng, M. Probst, HBB-NEXT I D4.4.1: Intermediate middleware software components for cloud service offloading, Tech. rep., HBB-NEXT Consortium 2013 (2013).

[14] H.-h. Chu, H. Song, C. Wong, S. Kurakake, M. Katagiri, Roam, a seamless application framework, Journal of Systems and Software 69 (3) (2004) 209–226.

[15] B.-G. Chun, P. Maniatis, Augmented smartphone applications through clone cloud execution, in: Proceedings of the 12th conference on Hot topics in operating systems, USENIX Association, 2009, pp. 8–8.

[16] E. Cuervo, Enhancing mobile devices through code offload, Ph.D. thesis, Duke University (2012).

[17] N. Duga, Optimality analysis and middleware design for heterogeneous cloud HPC in mobile devices, Master's thesis, Addis Ababa University (2011).

[18] H. Endt, K. Weckemann, Remote utilization of opencl for flexible computation offloading using embedded ECUs, CE devices and cloud servers, in: Volume 22: Applications, Tools and Techniques on the Road to Exascale Computing, Vol. 22 of Advances in Parallel Computing, IOS Press EBooks, 2011, pp. 133–140, this paper was not available online. I got it from the library.

[19] R. G. Esteves, M. D. McCool, C. Lemieux, Real options for mobile communication management, in: GLOBECOM Workshops (GC Wkshps), 2011 IEEE, IEEE, 2011, pp. 1241–1246.

[20] T. Fjellheim, S. Milliner, M. Dumas, Middleware support for mobile applications, International Journal of Pervasive Computing and Communications 1 (2) (2005) 75–88.

[21] J. Flinn, S. Park, M. Satyanarayanan, Balancing performance, energy, and quality in pervasive computing, in: In Proceedings of the 22nd International Conference on. Distributed Computing Systems, 2002, pp. 217–226.

[22] J. Flinn, S. Sinnamohideen, N. Tolia, M. Satyanarayanan, Data staging on untrusted surrogates, in: Proceedings 2nd USENIX Conference on File and Storage Technologies (FAST03), Mar 31-Apr 2, 2003, San Francisco, CA., 2003.

[23] I. Giurgiu, O. Riva, D. Juric, I. Krivulev, G. Alonso, Calling the cloud: Enabling mobile phones as interfaces to cloud applications, in: J. Bacon, B. Cooper (Eds.), Middleware 2009, Vol. 5896 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2009, pp. 83–102.

[24] S. Goyal, A collective approach to harness idle resources of end nodes, Ph.D. thesis, School of Computing, University of Utah (2011).

[25] T. Guan, A system architecture to provide enhanced grid access for mobile devices, Ph.D. thesis, University of Southampton (2008).

[26] K. Ha, G. Lewis, S. Simanta, M. Satyanarayanan, Cloud offload in hostile environments, Tech. rep., Carnegie Mellon University (2011).

[27] S.-H. Hung, J.-P. Shieh, C.-P. Lee, Migrating android applications to the cloud, International Journal of Grid and High Performance Computing (IJGHPC) 3 (2) (2011) 14–28.

[28] S. Imai, Task offloading between smartphones and distributed computational resources, Master's thesis, Rensselaer Polytechnic Institute (2012).

[29] A. N. Iyer, et al., Extending android application programming framework for seamless cloud integration, in: Mobile Services (MS), 2012 IEEE First International Conference on, IEEE, 2012, pp. 96–104.

[30] C. Jarabek, D. Barrera, J. Aycock, ThinAV: truly lightweight mobile cloud-based anti-malware, in: Proceedings of the 28th Annual Computer Security Applications Conference, ACM, 2012, pp. 209–218.

[31] R. Kemp, N. Palmer, T. Kielmann, H. Bal, Cuckoo: a computation offloading framework for smartphones, in: Mobile Computing, Applications, and Services, Springer, 2012, pp. 59–79.

[32] S. Kosta, A. Aucinas, P. Hui, R. Mortier, X. Zhang, Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading, in: INFOCOM, 2012 Proceedings IEEE, IEEE, 2012, pp. 945–953.

[33] D. Kovachev, R. Klamma, Framework for computation offloading in mobile cloud computing, International Journal of Interactive Multimedia and Artificial Intelligence 1 (7) (2012) 6–15.

[34] M. D. Kristensen, Empowering mobile devices through cyber foraging, Ph.D. thesis, Aarhus University (2010).

[35] S. Kundu, J. Mukherjee, A. K. Majumdar, B. Majumdar, S. Sekhar Ray, Algorithms and heuristics for efficient medical information display in PDA, Computers in Biology and Medicine 37 (9) (2007) 1272–1282.

[36] Y.-W. Kwon, E. Tilevich, Reducing the energy consumption of mobile applications behind the scenes, in: Proceedings of the 29th IEEE International Conference on Software Maintenance (ICSM 2013), 2013.

[37] B.-D. Lee, A framework for seamless execution of mobile applications in the cloud, in: Recent Advances in Computer Science and Information Engineering, Springer, 2012, pp. 145–153.

[38] J. Matthews, M. Chang, Z. Feng, R. Srinivas, M. Gerla, PowerSense: power aware dengue diagnosis on mobile phones, in: Proceedings of the First ACM Workshop on Mobile Systems, Applications, and Services for Healthcare, ACM, 2011, p. 6.

[39] A. Messer, I. Greenberg, P. Bernadat, D. Milojicic, D. Chen, T. Giuli, X. Gu, Towards a distributed platform for resource-constrained devices, in: Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on, IEEE, 2002, pp. 43–51.

[40] D. Messinger, G. A. Lewis, Application virtualizaton as a strategy for cyber foraging in resource-constrained environments, Tech. rep., Carnegie Mellon Software Engineering Institute (2013).

[41] S. Mohapatra, N. Venkatasubramanian, Optimizing power using a reconfigurable middleware, Tech. rep., UC Irvine (2003).

[42] M. Ok, J.-W. Seo, M.-s. Park, A distributed resource furnishing to offload resource-constrained devices in cyber foraging toward pervasive computing, in: Network-Based Information Systems, Springer, 2007, pp. 416–425.

[43] M. J. O'Sullivan, D. Grigoras, The cloud personal assistant for providing services to mobile clients, in: Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on, 2013, pp. 478–485.

[44] S. Park, Y. Choi, Q. Chen, H. Yeom, SOME: Selective offloading for a mobile computing environment, in: Cluster Computing (CLUSTER), 2012 IEEE International Conference on, 2012, pp. 588–591.

[45] T. Phokas, H. Efstathiades, G. Pallis, M. Dikaiakos, Feel the world: A mobile framework for participatory sensing, in: F. Daniel, G. Papadopoulos, P. Thiran (Eds.), Mobile Web Information Systems, Vol. 8093 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2013, pp. 143–156.

[46] L. Pu, J. Xu, X. Jin, J. Zhang, SmartVirtCloud: virtual cloud assisted application offloading execution at mobile devices' discretion, in: 2013 IEEE Wireless Communications and Networking Conference (WCNC): SERVICES and APPLICATIONS, 2013.

[47] M.-R. Ra, A. Sheth, L. Mummert, P. Pillai, D. Wetherall, R. Govindan, Odessa: enabling interactive perception applications on mobile devices, in: Proceedings of the 9th international conference on Mobile systems, applications, and services, MobiSys '11, ACM, New York, NY, USA, 2011, pp. 43–56.

[48] K. K. Rachuri, Smartphones based social sensing: Adaptive sampling, sensing and computation offloading, Ph.D. thesis, University of Cambridge (2012).

[49] M. R. Rahimi, N. Venkatasubramanian, S. Mehrotra, A. V. Vasilakos, MAPCloud: mobile applications on an elastic and scalable 2-tier cloud architecture, in: Proceedings of the 2012 IEEE/ACM Fifth International Conference on Utility and Cloud Computing, IEEE Computer Society, 2012, pp. 83–90.

[50] M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies, The case for VM-based cloudlets in mobile computing, Pervasive Computing, IEEE 8 (4) (2009) 14–23.

[51] C. Shi, P. Pandurangan, K. Ni, J. Yang, M. Ammar, M. Naik, E. Zegura, IC-Cloud: Computation offloading to an intermittently-connected cloud, Tech. rep., Georgia Institute of Technology (2013).

[52] J. N. Silva, L. Veiga, P. Ferreira, SPADE: scheduler for parallel and distributed execution from mobile devices, in: Proceedings of the 6th international workshop on Middleware for pervasive and ad-hoc computing, ACM, 2008, pp. 25–30.

[53] Y.-Y. Su, J. Flinn, Slingshot: deploying stateful services in wireless hotspots, in: Proceedings of the 3rd international conference on Mobile systems, applications, and services, MobiSys '05, ACM, New York, NY, USA, 2005, pp. 79–92.

[54] T. Verbelen, P. Simoens, F. De Turck, B. Dhoedt, Aiolos: Middleware for improving mobile application performance through cyber foraging, Journal of Systems and Software 85 (11) (2012) 2629–2639.

[55] Y. Xiao, P. Simoens, P. Pillai, K. Ha, M. Satyanarayanan, Lowering the barriers to large-scale mobile crowdsensing, in: Mobile Computing Systems and Applications, 2013.

[56] K. Yang, S. Ou, H.-H. Chen, On effective offloading services for resource-constrained mobile devices running heavier mobile internet applications, Communications Magazine, IEEE 46 (1) (2008) 56–63.

[57] F. Yang, Z. Qian, X. Chen, I. Beschastnikh, L. Zhuang, L. Zhou, J. Shen, Sonora: A platform for continuous mobile-cloud computing, Tech. rep., Technical Report. Microsoft Research Asia (2012).

[58] Y. Zhang, X.-t. Guan, T. Huang, X. Cheng, A heterogeneous auto-offloading framework based on web browser for resource-constrained devices, in: Internet and Web Applications and Services, 2009. ICIW'09. Fourth International Conference on, IEEE, 2009, pp. 193–199.

[59] X. Zhang, A. Kunjithapatham, S. Jeong, S. Gibbs, Towards an elastic application model for augmenting the computing capabilities of mobile devices with cloud computing, Mobile Networks and Applications 16 (3) (2011) 270–284.

[60] Y. Zhang, G. Huang, W. Zhang, X. Liu, H. Mei, Towards module-based automatic partitioning of java applications, Frontiers of Computer Science 6 (6) (2012) 725–740.

[61] X. Zhang, W. Jeon, S. Gibbs, A. Kunjithapatham, Elastic HTML5: Workload offloading using cloud-based web workers and storages for mobile devices, in: Mobile Computing, Applications, and Services, Springer, 2012, pp. 373–381.

[62] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, A. Patti, Clonecloud: Elastic execution between mobile device and cloud, in: Proceedings of the Sixth Conference on Computer Systems, EuroSys '11, ACM, New York, NY, USA, 2011, pp. 301–314.

[63] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, R. Buyya, Cloud-based augmentation for mobile devices: Motivation, taxonomies, and open challenges, IEEE Communications Surveys Tutorials 16 (1) (2014) 337–368.

[64] H. T. Dinh, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: architecture, applications, and approaches, Wireless Communications and Mobile Computing 13 (2011) 1587–1611.

[65] N. Fernando, S. W. Loke, W. Rahayu, Mobile cloud computing: A survey, Future Generation Computer Systems 29 (2012) 84106.

[66] R. K. Lomotey, R. Deters, Architectural designs from mobile cloud computing to ubiquitous cloud computing-survey, in: Services (SERVICES), 2014 IEEE World Congress on, IEEE, 2014, pp. 418–425.

[67] K. Kumar, J. Liu, Y.-H. Lu, B. Bhargava, A survey of computation offloading for mobile systems, Mobile Networks and Applications 18 (1) (2013) 129–140.

[68] P. Yu, X. Ma, J. Cao, J. Lu, Application mobility in pervasive computing: A survey, Pervasive and Mobile Computing 9 (2012) 2–17.

[69] J. Flinn, Cyber foraging: Bridging mobile and cloud computing, in: M. Satyanarayanan (Ed.), Synthesis Lectures on Mobile and Pervasive Computing, Morgan & Claypool Publishers, 2012.