AndroCon: An Android-Based Context-Aware Middleware Framework for Data Provisioning

Jian Yu
Department of Computer Science, Auckland University of Technology
Auckland, New Zealand
jian.yu@aut.ac.nz

Quan Z. Sheng
Department of Computing, Macquarie University
Sydney, Australia
michael.sheng@mq.edu.au

Wei Qi Yan
Department of Computer Science, Auckland University of Technology
Auckland, New Zealand
wyan@aut.ac.nz

Chris Wang
Department of Computer Science, Auckland University of Technology
Auckland, New Zealand
chris.wang@aut.ac.nz

Olayinka Adeleye
Department of Computer Science, Auckland University of Technology
Auckland, New Zealand
olayinka.adeleye@aut.ac.nz

ABSTRACT

Mobile devices have become major sources of context-aware data due to their ubiquity and sensing capabilities. However, deploying mobile devices as dynamic, unabridged context data provider either locally or remotely is still challenging due to their limited computing capability. Furthermore, integrating physical sensor data with social context data from online social networks is necessary for rich context data provisioning. In this paper, we present AndroCon, an Android-based context-aware middleware framework that enables mobile devices to acquire, integrate, manage, and provision context data. We have applied AndroCon to manage social and physical context data from various sources and have evaluated its performance in terms of power consumption and CPU utilization.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; *Ubiquitous computing*; • **Information systems** → RESTful web services;

KEYWORDS

Data Provisioning, Context-Awareness, Integration, Middleware, Android

ACM Reference format:

Jian Yu, Quan Z. Sheng, Wei Qi Yan, Chris Wang, and Olayinka Adeleye. 2017. AndroCon: An Android-Based Context-Aware Middleware Framework for Data Provisioning. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Melbourne, Australia, November 7–10, 2017 (Mobiquitous 2017),* 2 pages. https://doi.org/10.1145/3144457.3144509

Mobiquitous 2017, November 7–10, 2017, Melbourne, Australia

ACM ISBN 978-1-4503-5368-7/17/11…$15.00
https://doi.org/10.1145/3144457.3144509

1 INTRODUCTION

Over the years, the computing and sensing capacity of mobile devices has been constantly improving to support smart context-aware applications. While the volume of context data provided through various mobile applications and sensors are the rapid increase, collecting these data from various physical and virtual sources, integrating them for efficient processing, and provisioning them both locally and remotely are still challenging and have become the major bottlenecks in developing mobile context-aware systems [4].

We present in this paper AndroCon, an Android-based, context-aware middleware framework for acquiring, integrating, managing, and provisioning both physical and social context data. AndroCon integrates AWARE [1] and SCIMS [2] frameworks. While AWARE acquires raw physical context data from mobile sensors, SCIMS collects raw social context data from multiple social media sources. AndroCon uses a general context ontology to represent contextual knowledge and to organize raw context data. It also supports the provisioning of context data with a cross-platform web service interface.

2 SYSTEM OVERVIEW

The AndroCon system comprises of four main components as shown in Figure 1: the context acquisition is the bottom layer, which incorporates a series of adapters from AWARE and SCIMS frameworks. Above is the context integration layer which provides a common interface for accessing the raw context data. The integration layer passes data to the context storage layer, which organizes context data as RDF triples based on a general context ontology. The top is the context provision layer, which provisions context data to applications through RESTful API.

2.1 Context Acquisition

AndroCon acquires both physical and social context data by reusing AWARE and SCIMS frameworks.

The AWARE client runs on Android mobile devices and collects raw data from sensors such as barometer, gyro, temperature, accelerometer, proximity, compass, gesture, and heart rate. In order to

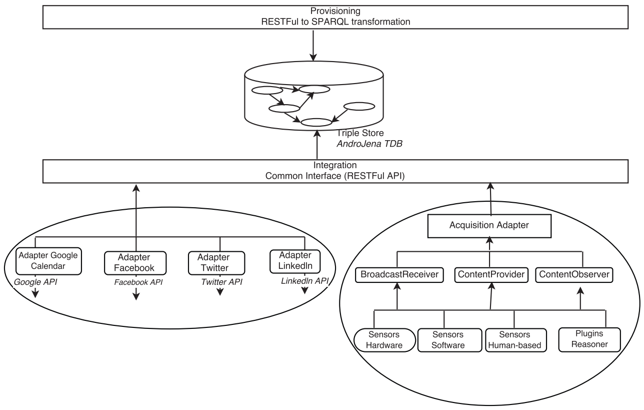

Figure 1: AndroCon System Architecture

allow other Android applications to have access to context data retrieved, it stores the data in a local relational database and encapsulates it with Android's *ContentProviderAPI*. AWARE also supports the *ContentObserver* approach, which allows messages to be sent only when context changes. AndroCon adopts *ContentObserver* to ensure that only modified context be acquired and stored.

SCIMS retrieves social context via Online Social Network APIs. Sources of social context include Facebook, Twitter, LinkedIn, etc. SCIMS is a server-based framework only and AndroCon uses an adapter to invoke its RESTful API to acquire the raw social context data.

2.2 Context Integration

AndroCon uses ontology based approaches to organize and integrates context data. It uses the *NeOn* methodology [5] to merge and re-engineer ontological resources from the *mIO*! Ontology [3] and the SCIMS ontology.

AndroCon physical ontology is divided into two layers: The upper level captures basic concepts abstracted from the real-world scenarios including classes such as *Activity*, *Location*, *Environmental Condition*, and *Physicalentities*. The lower layer comprises of subclasses of the upper layer.

AndroCon social ontology defines five upper-class entities, which comprises of three FOAF ontology used to define *Person*, *Organization* and *group*. The other components are Social class and Instant entities. The resulting integrated ontology is shown in Figure 2.

2.3 Context Provisioning

One of the features that separates AndroCon from other context-aware frameworks is its capability to enable mobile devices function as a Web service provider. AndroCon Web services consists of two layers: the web server layer and the database layer. The web server upon receiving the request generates different types of triple store transactions based on the type of HTTP method invoked. The database layer is implemented as a RDF triple store that consists of a SPARQL engine and an underlying TDB instance. The *AndroConAddress* object will transfer RESTful requests to SPARQL statement and forward the statement to SPARQL engine. The engine then interacts with TDB and return the query results.

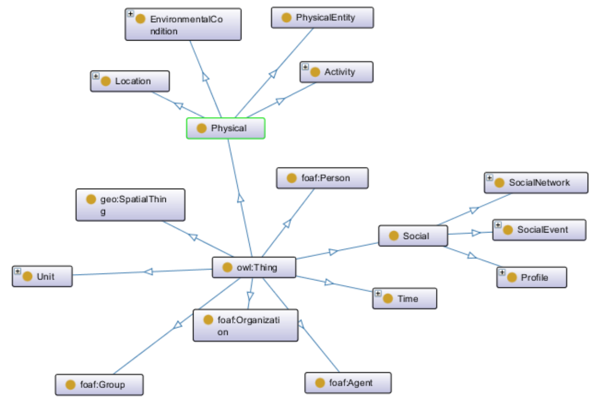

Figure 2: AndroCon Integrated Context Ontology

Table 1: AndroCon testing Results

Run Time (mins)	Battery Consumption %	CPU Load %
5	4	4.04
10	7	4.10
15	9	4.12
20	12	5.17
30	13	6.27
60	19	5.85

3 EVALUATION

We have evaluated AndroCon's battery consumption and CPU load on Nexus 6.0 Android phone. Qualcomm's Trepn Profiler (*QTP*) is used to get the readings. The results are shown in Table I, which contains records for time taken to execute the entire AndroCon processes, Battery Consumption in percentage, and CPU load in percentage.

As we can see in Table I, running AndroCon for one hour consumes about 19% of the battery energy, which means AndroCon can run continuously for about five hours on a mobile phone. During the testing timeframe, AndroCon starts by using about 4% of CPU loads and then increase to about 6% at the end of the testing. But In general, the CPU load is stable between 4-6%.

4 CONCLUSION

In this paper, we have presented AndroCon, an Android-based context-aware middleware framework for acquiring, integrating, and provisioning both physical and social context. Our future direction is on its application in the Internet of Things area.

REFERENCES

[1] Denzil Ferreira, Vassilis Kostakos, and Anind K Dey. 2015. AWARE: mobile context instrumentation framework. Frontiers in ICT 2, 6: 1–9. (2015).

[2] Muhammad Kabir, Jun Han, Jian Yu, and Alan Colman. 2012. SCIMS: a social context information management system for socially-aware applications. In *Advanced Information Systems Engineering*. Springer, 301–317.

[3] Maria Poveda Villalon, Mari Carmen Suárez-Figueroa, Raúl García-Castro, and Asunción Gómez-Pérez. 2010. A context ontology for mobile environments. (2010).

[4] Vaskar Raychoudhury, Jiannong Cao, Mohan Kumar, and Daqiang Zhang. 2013. Middleware for pervasive computing: A survey. *Pervasive and Mobile Computing* 9, 2 (2013), 177–200.

[5] Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, and Mariano Fernández-López. 2012. The NeOn methodology for ontology engineering. In *Ontology engineering in a networked world*. Springer, 9–34.