Airport Context Analytics

Eli Katsiri[1 2], George Papastefanatos[2], Manolis Terrovitis[2], and Timos Sellis[3]

[1] Department of Electrical and Computer Engineering,
Democritus University of Thrace, Xanthi, 67100, Greece,
[2] Institute for the Management of Information Systems,
Research and Innovation Centre in Information, Communication and Knowledge Technologies - "Athena",
Artemidos 6 & Epidaurou, Marousi, 15125, Athens, Greece,
{eli,gpapas,mter}@imis.athena-innovation.gr,
[3] School of Computer Science and Information Technology, RMIT University, Australia,
timos@rmit.edu.au

Abstract. Airports today can constitute a perfect environment for developing novel digital marketplaces offering location-specific and semantically rich context-aware services, such as personalized marketing campaigns, last minute, discounted airline tickets while helping users access the airport and speed through the airport process.
Underpinning the above vision is the ability to target service content to users' current context, e.g., their location, intent, environment, in real time. The contribution of this work is that it uses a *pervasive computing* system with three key ingredients: (a) a data model, comprising user and service content entities, (b) a user context model and (c) rules for simple *pattern matching* on service content and user context *event streams*. This modus operandi logic is encapsulated inside a SOA architecture, the *Common Airport Portal - CAP* and it is illustrated through the description of a real application, Offers and Coupons Services that was deployed recently at Athens International Airport (AIA).

Keywords: airport information systems, context-awareness, real-time analytics, personalisation, rule-based reasoning, system implementation

1 Introduction

Airports nowadays form an integral part of urban spaces, and have become more than just a place where you fly from. They provide the technology substrate for commercial stakeholders (e.g., airport companies, shipping companies, retail shops, etc.) to offer added-value services to potential consumers of the airport community (visitors, passengers and airport employees) and for airport stakeholders (e.g., airport authorities) to improve airport infrastructure, redesign existing services and take strategic decisions about the future.

An airport can collect and provide valuable information in the form of event streams (feeds), to location-constrained stakeholders about passenger flows, user travel status, alerts, flight data and time-schedules; this data is instrumental in developing spatial, temporal and context-aware personalized services, seamlessly operating over a variety of distribution channels such as web, sms, email, IPTV, info-kiosks. Such added-value services include but are not restricted to, *personalised route calculation* from a user's current position to their departure gate, *personalised marketing campains* involving offers and discount coupons, *last minute air-tickets* and *personalised alerts*. The above services are personalised and context-aware, i.e., relevant to user role, device, location, departure gate etc. Personalised marketing campaigns provide offers and *dynamic recommendations* based on what other users with similar profile are currently purchasing. These are relevant to the above parameters while abiding by constraints that are inherent in the airport domain: for example, *passengers flying to middle Eastern destinations should not be targeted for offers on alcoholic beverages*, while, *airport visitors escorting passengers should not be targeted for Duty Free discounts*.

Three successful paradigms exist in the literature that can be leveraged in order to achieve these goals: *Data Analytics* [1], *context-awareness and personalisation* [2] and *Service Oriented Architecture (SOA) [3]*: Big data helps businesses understand the data and extract patterns in order to become smarter. Context-awareness advocates that applications should be aware of user context in order to best serve them. Service Oriented Architecture (SOA), provides structured collections of discrete software modules, known as services that collectively provide the complete functionality of an application, with the ability of being reusable and composable into complex applications.

Combining the above paradigms leads to the definition of **airport context analytics**: User generated data such as position and device usage can be collected, analyzed and correlated with other airport sources in order to make service content provisioning aware of user context. The contribution of this work is that it uses the *Rete algorithm* [20] as the analytics engine. Rete is a very fast algorithm for matching data tuples (”facts”) against productions (”rules”). By reducing certain types of redundancy through the use of node sharing and by storing partial matches when performing joins between fact types, Rete avoids the complete re-evaluation of all facts each time changes are made to working memory, thus increasing performance and scalability.

The rest of this paper is structured as follows: Section 2 presents a context model for the airport domain. Section 3 discusses the ACA service its implementation using web services. Section 4 discusses literature and concludes.

2 Context Model

Due to restrictions very specific to the airport community culture, e.g. the sensitive nature of passenger flight data, context that is not available can be *inferred*. For this reason the context model contains both *static* (user profile and preferences) and *dynamic* predicates. Dynamic predicates can be either low-level, directly available from the sources, or high-level, indirectly derived from low-level attributes, by mining or inference. Examples of low-level dynamic predicates are *location* (user and device locations) and *user activity* (user clicks, user requests, coupon redeems). Examples of high-level predicates are *user colocation*. Dynamic predicates that are *infrequently changing*, (frequent flyer) are stored in generic placeholders in the database, as opposed to *frequently changing* ones (location) that is stored in memory. Next, a formal definition using Hoare logic is given.

Data Model comprises *User* and *Generic Service Content*, representing generic content that is eligible for processing and provisioning within the airport environment (e.g., Offers, Coupons, LM tickets, etc.). An *Offer* is a type of advertisement of discounted products on sale while a *Discount Coupon* needs to be *redeemed* at the time of purchase. *LM Tickets* are specialized Offers for airline tickets departing within the next 48 hours.

Static context entities include $Role(rid, description)$ and $Target\_Group$. An AIA User is an employee of the airport authority, Airport User is an employee of the associated companies and a Company Admin is a privileged user that can authorize service content updates. $Target\_Group$ represents a combination of roles. Also included are: $Product\_Category\_Prefs$, $Company\_Prefs$, $Distribution\_Channel\_Prefs$.

Dynamic, frequently changing, context entities include $Location(zone\_id, description, < x, y, z >)$, where $< x, y, z >$ are positions encoded in the AIA GIS coordinate system and $zone\_id$ are locations at airport zone granularity. $Intent(uid, description)$ refers to airport uses: an AIA User (Role) may be traveling as a regular passenger (Intent) and therefore be eligible for Duty Free offers. $Trip\_Status$ models the nature of the trip, e.g., $\{business, economy, traveling\_with\_family\}$. $Trip\_Phase$ results from associating airport zones with the airport processes: $\{before\_check\_in, after\_security, at\_lounge, at\_departure\_gate\}$. $User\_Activity$ can be inferred from logged information and user location (e.g., $has\_requested\_offer$).

Dynamic, infrequently changing, context entities represent features that are mined from historic context instances and include $Frequent\_Traveler$ $(uid, description)$, $Frequent\_Shopper$ $(uid, description)$, $Technology\_Savviness$ $(uid, description, level)$. The latter is determined by *device type* (conventional or smart phone), *frequency of service use* and *method of coupon redemption* (printed or electronic coupons.) Such knowledge can be used, for example, for *creating discounts campaigns targeted only to frequent shoppers.*

Data Model and contextualization. User and Service content, when associated with context entities, becomes *contextualized*. For example, the contextualized $Generic\_Service\_Content$ predicate is specified as follows:

$$Generic\_Service\_Content(gscid, args, tgid, rid, dcid, oid)$$

where $\{tgid, rid, pcid, compid, dcid, oid\}$ represent the context predicates: $\{Target\_Group, Role, Product\_Category\_Prefs, Distribution\_Channel\_Prefs, Opt\_ins\}$.

Matching contextualized service content to eligible users is implemented by the following simple rule:

$$\begin{aligned}
&User(uid, profile, context_id[])\\
&\wedge Offer(gsid, oid, prod_cat)\\
&\wedge Generic_Service_Content(gsid, context_id[])
\end{aligned}$$

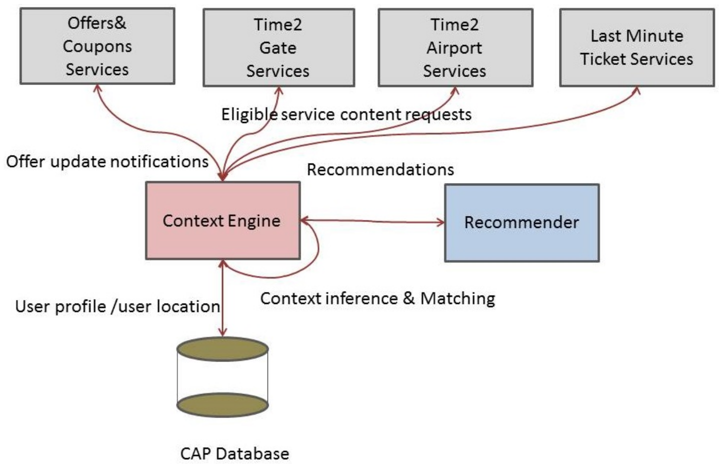

Fig. 1. ACA Service Component Diagram

$$\wedge user.context\_id[] = offer.context\_id[]$$

where $context\_id[]$ represent matching context predicates between users and service content (e.g., offers for AIA Users).

3 Airport Context Analytics Service (ACA)

The ACA Service modus operandi is the following:

1. **Offer Matching:** On reception of a request from any of the services shown in Figure 1 (top), containing (a) a description of the offer or coupon or related content to be dispatched, (b) the unique id of the current user and (c) a set of optional eligibility parameters, respond by returning a list of requested content that is eligible for that user, based on the supplied parameters. This is the primary operation.
2. **Recommendation Matching:** On reception of a request from one of the CAP components shown in Figure 3, containing (a) an ordered list of recommendations to be dispatched, (b) the unique id of the current user and (c) a set of optional eligibility parameters, respond by returning a subset of the ordered list of recommendations that is eligible for that user, based on the supplied parameters.
3. **Offer Notification:** On insertion, deletion, update, cancellation of an Offer, Coupon, LMT Offer or similar content entity, respond by repeating the matching process and generating a new set of eligible content. This mechanism is based on publish/subscribe.

Implementation (Figure 2) includes three basic web services: *CAP Drools Server, CAP Context Service, CAP Context Notify Service*. CAP Drools Server

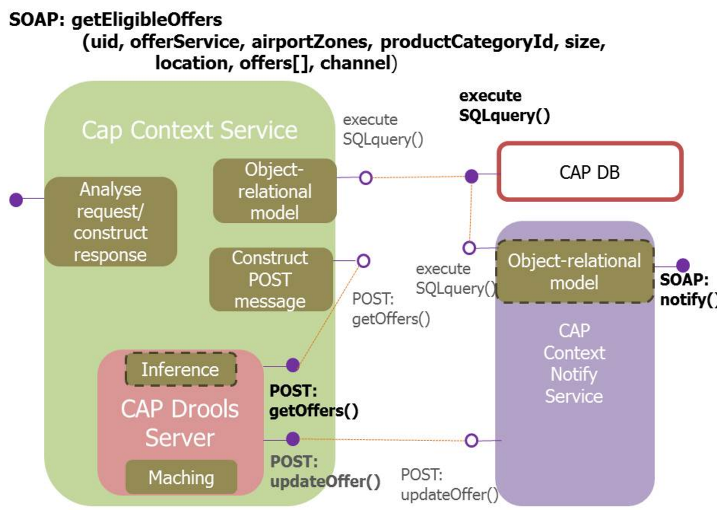

Fig. 2. ACA Implementation

integrates *Drools* [13], a business rule management system based on the*Rete algorithm*. CAP Context Service receives the service requests, extracts service parameters and queries the CAP database for the most up-to-date data. Next it constructs contextualized facts, encodes them in HTTP POST messages and inserts them in the Drools Server, triggering a matching cycle. The CAP Context Notify Service implements publish/subscribe. First, it registers with the CAP Event Server, subscribing to databases changes (inserts, updates, deletes, cancellations, rejects) to the tables Users, Offers and Coupons. Next, it listens asynchronously for any such events, in which case it invokes the CAP Drools Server, making all relevant updates to the affected facts and re-triggering a matching cycle. In this way, when queried by the service layer, it is up-to-date.

4 Related work

Several definitions for context have been proposed in the literature [4–8], including the author's previous work [21, 22]. Certain mobile applications [15–18] use context for selecting the best communication channel per device and application. [11] proposes a platform for executing web services that adapt to application QoS, under changing conditions. [19] discusses an airport knowledge-base system designed with the CommonKADS methodology. None of these works are directly applicable here. CAP (http://airpoint.gr/en) goes far beyond previous efforts bringing an integrated solution, a new paradigm for service management in indoors spaces.

References

1. Jagadish, H. V.: Big Data: It's not just the analytics. http://wp.sigmod.org/?p=430
2. Weiser, M.: Ubiquitous Computing, IEE Computer, Hot Topics, (1993)
3. Arsanjani, A.: Service-oriented Modeling and Architecture. How to identify, specify and realize services for your SOA. http://www.ibm.com/developerworks/library/ws-soa-design1
4. Schilit, B., Adams, N., Want, R.: Context-aware computing applications. First International workshop on Mobile Computing Systems and Applications, 85–90 (1994)
5. Schilit, B.: Disseminating active map information to mobile hosts. IEEE Network, 8(5):22–23 (1994)
6. Dey, K., Abowd, G. D., Wood, A.: CyberDesk: A framework for providing Self-Integrating context-aware services. Knowledge-base Systems, 11:3–13 (1999)
7. Brown, P. J.: The Stick-e Document: A framework for creating context-aware applications. Electronic Publishing 96 259-272 (1996)
8. Dey, A. K., Abowd, G.: Towards a better understanding of context and context-awareness. HUC '99 Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing (1999)
9. Mahout: Scalable machine learning and Data Mining. *http://mahout.apache.org/*
10. Welcome to Apache Hadoop. *http://hadoop.apache.org/*
11. PAWS: Processes with adaptive web services. *http://www.paws.elet.polimi.it/*
12. Truong, H. L., Dustdar, H.: A survey on context-aware Web Service Systems. International Journal of Web Information Systems of context and context-awareness. 5(1):5-31, Emerald, (2009)
13. Drools: The business logic integration platform. *http://www.jboss.org/drools*
14. Sanchez-Pi, N., Carbo, J., Molina J.M.: Building a Knowledge Base System for an Airport Case of Use. International Symposium on Distributed Computing and Artificial Intelligence 2008 (DCAI2008), Advances in Intelligent and Soft Computing, 50:739-747 (2009)
15. Capra, L., Emmerich, W., Mascolo C.: Carisma: Context-aware reflective middleware system for mobile applications, IEEE Transactions on Software Engineering. 29:929–944 (2003)
16. Chakraborty, D., Lei, H.: Pervasive enablement of business processes. Second IEEE Annual Conference on Pervasive Computing and Communications (PERCOM04), Orlando, Florida, USA (2004)
17. Team, T., Mais, M.: Multichannel adaptive information systems. International Conference on Web Information Systems Engineering, Rome, Italy (2003)
18. Capiello, C., Comuzzi, M., Mussi E., Pernici B.: Context Management for Adaptive Information Systems.
19. Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W., Wielinga, B.: Knowledge Engineering and Management, The CommonKADS Methodology. MIT Press, Cambridge (1999)
20. Forgy, C.: Rete: a fast algorithm for the many pattern/many object pattern match problem. Expert Systems, 324–341, IEEE Computer Society Press (1990)
21. Katsiri, E.: Knowledge-base representation and reasoning for the autonomic management of pervasive healthcare. Eighth Join Conference on Knowledge-Based Engineering (JCKBSE 2008), Aug. 25-28, University of Piraeus, Greece (2008)
22. Katsiri, E., Bacon, J., Mycroft, A.: Linking sensor data to context-aware applications using abstract events. International Journal of Pervasive Computing and Communications. 3(4):347–377 (2007)