PLIS+: A Rule-Based Personalized Location Information System

Iosif Viktoratos[1], Athanasios Tsadiras[1], Nick Bassiliades[2],

[1]Department of Economics, [2]Department of Informatics,
Aristotle University of Thessaloniki
GR-54124 Thessaloniki, Greece

{viktorat, tsadiras, nbassili}@auth.gr

Abstract. In this paper, the idea of providing personalized, location-based information services via rule-based policies is demonstrated. After a short introduction, an innovative Personalized Location Information System (PLIS+) is designed and implemented. PLIS+ delivers personalized and contextualized information to users according to rule-based policies. More specifically, many categories of points of interest (for example shops, restaurants) have rule-based policies to expose and deploy their marketing strategy on special offers, discounts, etc. PLIS+ evaluates these rules on-the-fly and delivers personalized information according to the user's context and the corresponding rules fired within this context. After discussing the design and the implementation of PLIS+, illustrative examples of PLIS+ functionality are presented. As a result, PLIS+ proves that combining contextual data and rules can lead to powerful personalized information services.

Keywords: RuleML, Rules, Location Based Services, Context, Points of Interest, Jess.

1 Introduction

1.1 Rule-based Information Services and related work

Latest information services adopt rule based approaches so as to enable higher quality context perception. Rule-based systems are more autonomous because they are capable of understanding context changes and responding accordingly without user intervention [1].

As a result, up-to-date Location Based Services (LBS) combine semantics (ontologies, rules) with smartphone's capabilities (GPS, sensors) to deliver contextualized up-to-date information [2-5] to users. Thus, LBS have become a popular sector of everyday life and they are used consistently by millions of people for navigation, tracking, information, even in emergency situations [6].

1.2 Motivation-Overview

The aim of the presented work is to combine semantics with location information services to deliver personalized and contextualized information services to users. A system called "Personalized Location Information System" or PLIS+ was implemented for this purpose. PLIS+ is an extended version of the PLIS system that is presented in [7]. It can be accessed at http://tinyurl.com/ca42fwj

A rule-based approach was followed for PLIS+ implementation, based on discussion in section 1.1 Core component of PLIS+ is RuleML, a powerful markup language (XML with a predefined Schema) which supports various types of rules such as deductive, reactive and normative. As an XML-based language, RuleML addresses the issues of interoperability and flexibility among different systems on the web, by allowing rules to be encoded in a standard way [8]. Last but not least, because of the fact that PLIS+ users are capable of adding rules at run-time, an xml-based user friendly language is desirable.

PLIS+ could easily be combined with most of existing approaches and differs by enabling a dynamic rule base that offers users the option to add rules at run time. A detailed demonstration of the system is included in the following sections.

2 Design, Implementation and technical details

In everyday life, in order to deploy their specific business strategy, many points of interest (as businesses) adopt a rule-based policy (e.g. special offers). The general idea is to combine POI's policies with user's context to deliver 'qualitative' information. A general interface for connection between POI owners and potential customers-users is provided. Every time a user logs into the system to search for a point of interest, PLIS+ gets user's context (profile, location, day, time), evaluates the rules associated with nearby POIs and delivers personalized information to user, depending on the rules fired. Users can also become owners of various POIs and after that they are capable of inserting their own rule base policy for those POIs. The general idea is illustrated in Figure 1.
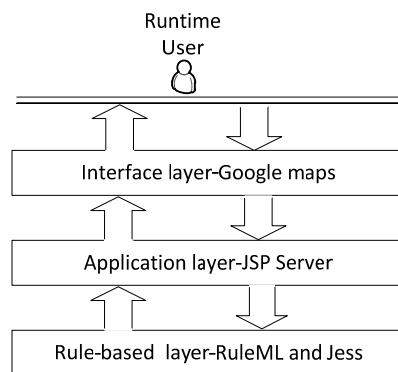

Fig. 1 General design of system

To be more specific, PLIS+ offers the following functionalities illustrate in Figure 2:

A: User's Registration:
- A1. User registers to the system by completing a registration form so as PLIS+ to build a profile (registration time user).
- A2. User profile data such as first name, last name, occupation, gender, age, city, state, etc are stored in the database.

B: Insertion of Points of Interest:
- B0. After a user has logged in, the system obtains user position and retrieves nearest POI's from external sources such as Google Places API. If information of a POI is already in the system, PLIS+ updates its related data with the latest information.
- B1. User is capable (by becoming POI owner) of attaching to existing places attributes and explicit rules relevant to those attributes. In detail, user is able to assert a rule base which contains any attribute related to his POIs and then assert rules concerning these attributes. For example a restaurant owner asserts a policy which contains data (related to his business) such as Pizza 10€, Spaghetti 8€, Minimum order 5€ and along with above data, relevant rules such as "*if a person is a student and hour is after 18:00, then Pizza price is 8€*".Furthermore, a user is of course capable of inserting his/her own POIs accompanied by their own rule based policy. Alternatively, place owners can upload their rule base to their website and insert the relevant link. The editing of the corresponding rule base is authorized only to the POI's owner.
- B2. All data and rules concerning the POI are saved to the corresponding database. Except from this, files containing rule bases are kept to the server.

C: Presentation of Personalized information. To present the personalized information to the end user, the following steps are made:

- *Step C1*:
  a. After registration, user is able to log into the system by entering his/her username and password.
  b. System checks user profile database for authentication.
- *Step C2*: Java Server Pages (JSP) collects user context (profile, location, time, day) attribute values (run time user).
- *Step C3*:
  a. For every POI, rules (if any) are being fetched (by JSP), along with relevant attribute values (for example price, etc).
  b. Rules (after being transformed to a machine understandable language), POI data and user context attribute values are asserted to the Jess rule engine.
- *Step C4*: Jess rule engine evaluates rules using the asserted facts and updates POIs' data according to the rules fired depending on user's context. The new data are fetched by JSP.
- *Step C5*: Finally, data transfer to client is performed for visualization and presentation of personalized information. It's worth mentioning that a user-oriented interface has been implemented so as the run-time user to become capable of understanding the general idea of PLIS+. First of all, different markers are applied for better illustration. In detail, except from the standard red marker for POIs, a) a yellow marker

indicates that the place contains a rule base but no rule fired for current user, b) a green marker indicates that the place contains a rule base and rules were fired for the current user, c) a crown over the marker indicates that the current user is also the POI owner of this place. Moreover, when a person clicks on a place additional information appears in a message explaining which rules were fired and why or in the case that no rule was fired for the specific POI, which rules exist for the place (if any).

Basic component of PLIS+ is Reaction RuleML, a subcategory of RuleML. It is used for rule representation. This subcategory was chosen because such kinds of policies are usually represented by production rules and Reaction RuleML is suitable for that task [9]. Reaction RuleML also supports both deductive rules, i.e. rules which derive data at their RHSs, and event-based rules, i.e. rules which are activated by specific events, such as user actions or temporal events. Therefore, Reaction RuleML could be easily used to express a multitude of rule-based calculations and data processing for business strategies.

On the other hand, to implement a system like PLIS+, an inference engine is needed in order rules to be executed by a machine. Jess was chosen to implement the core of PLIS+, because of the fact that it is a lightweight rule engine and connects well with web technologies, which were needed for PLIS+ system implementation [10]. So for example, after the translation from RuleML to Jess, the rulebase for the POI with the following characteristics: a) pizza: 10€, b) spagheti: 8€, c) minimum order: 5€ and d) a rule considering "pizza price decreased to 8€ after 18:00 for students" is represented as shown below:

```
(bind ?fact (assert (place (pizza 10) (Spaghetti 8)
(minimum_order 5))))
(defrule decreased_pizza_price
  (declare (salience 10))
  (person (hour ?d) (occupation student))(test (> ?d 18))
=>(modify ?fact (pizza 8) )
(store EXPLANATION "Pizza price decreased to 8€ after 18:00 for students"))
```

Rules in RuleML format transformed to Jess rules by using XSLT [11]. XSL technology is used massively to transform XML documents. Another core technology of PLIS+ is Java Server Pages (JSP). The vast majority of rule engines such as Jess, Drools, Prova [12] are not only server-oriented (for security issues) but also java-based. JSP implementation fulfills both criteria. Moreover, JSP can be easily embedded into html documents and is heavily used along with client-oriented technologies such as JavaScript for visualization.
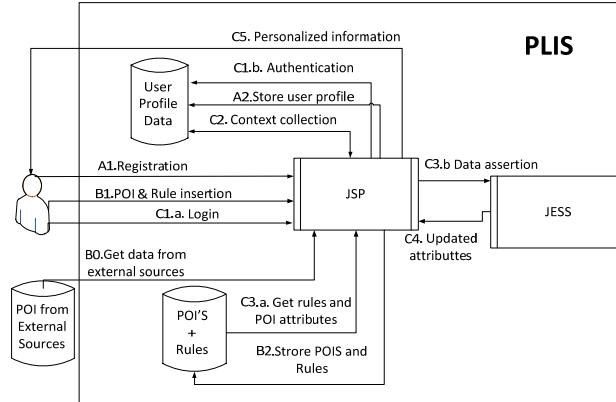
Fig. 2 PLIS+ functionalities

3   Demonstration of PLIS+

This section includes a demonstration of the PLIS+ system. A random user profile snapshot is used as an example (Table 1). On the other hand two random places from the database selected for testing (**Table 2**). Table 2 shows the attributes and also the rules that were attached to these places.

Table 1. A random user profile.

Profile					Environment		
	Name	*Occupation*	*Gender*	*Age*	*Time*	*Day*	*Location*
User A	Susan	Student	Female	26	20:12	Friday	LocationA

Table 2. Two random places.

	Name	**Asserted Data**	**Rule 1**	**Rule 2**
Place A	Cafe Delmundo	Coffee:3€ Ice-cream: 5€	Coffee price reduced to 2€ for students which are closer than 200m after 22:00	Half prices for coffee and ice-cream (1.5 and 2.5€) for women on Fridays after 18:00 o'clock
Place B	Verona pizza	pizza:10€ fish:12€ minimum order:5€	40% discount (6€) into pizza price for students	special prices for pizza (8€) and a decreased minimum order (4€) for women under 35 years old which are closer than 500m

As it was previously referred, PLIS+ gets user profile, evaluates rules and displays personalized information. When user A ("Susan") logs into the system she is able to click on places displayed by green markers (places containing rules that fired according to current user profile), so as to understand which rules fired for her and why

(Figure 3). Taking Place A (Table 2) as an example, rule 2 is fired (because she is a woman, the day is Friday and current time is after 18:00). PLIS+ updates attribute values according rule 2 and delivers contextualized information to Susan (Figure 4). Coffee and ice-cream price for Susan are 1.5 and 2.5€ after rule 2 execution. She is also capable of understanding why a rule fired with the rule explanation field.

Fig. 3. PLIS+ Starting screen

Similarly, both rules were fired for place B. Susan is a student (Rule 1 criterion) and she is a woman under 35, closer than 500m from place B (Rule 2 criteria, assuming that location A is closer than 500m). According to Rule 2, minimum order for current user is 4€, but there is a confliction for pizza price (pizza price is 6€ according to rule 1 and 8€ according to the second). PLIS+ handles rule confliction problems by applying priorities according to assertion turn. A Rule which was inserted first has a higher priority. Taking those under consideration pizza price for Susan is 6€ and related information displayed as in Figure 5.

This example illustrates how the delivered information is displayed to the end user and the capabilities of PLIS+ concerning a) gender, b) day, c) location, d) occupation, e) time, f) a non-applicable rule case and g) rule confliction.

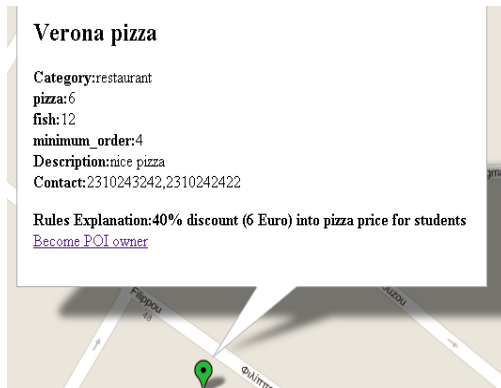**Fig. 4.** Information for Susan concerning Place A.

Fig. 5. Information for Susan concerning place B

4   Limitations and comparison with related works

A limitation of the proposed system is the fact that users can add data and rules concerning these data only in textual form. Furthermore, no rule sharing between place owners or a rule recommendation operation is supported. Moreover, there are some security issues concerning place owners operations. For example, an authentication process validating that the user who is becoming POI owner is the actual place owner would be desirable. In addition to this, a rule validation operation would be useful.

   Concerning comparison with related approaches described in section 1 PLIS+ differs by letting users adding and editing rules at run time. It is a fully dynamical service where users can add an unlimited number of data and rules, and consequently it becomes more and more intelligent and autonomous as soon as new data and rules are asserted to the system. On the other hand, many rule based approaches adopt more complex rule bases in relation to PLIS+. Last but not least, by using RDF and OWL such kind of services can be more flexible.

5   Conclusions and Future Work

Location-Based Services are currently popular, enjoying both commercial and scientific interest. An important part of LBS is the points of interest, and as mentioned, most of them can have their explicit rule policies. Taking these facts under consideration, embedding rules dynamically to location-based information systems can offer a boost to the quality of delivered information. PLIS+ combines location-based technologies with rule-based technologies to demonstrate the viability of this idea.

   PLIS+ implementation can evolve in various ways. First of all, because of the fact that POI owners are unfamiliar with RuleML, a user-friendlier environment has to be implemented. Either a form-based interface will be implemented, or otherwise a ruleml editor [13] will be embedded, so as place owners to become capable of adding and editing rules without much effort. Additionally, system should propose rules that

other owners have added previously and run time owners-users should be capable of choosing between existing rule sets.

Furthermore, in our future plans is to use OWL and/or RDF data (as in linked data) to represent user profiles and POI related information, for greater flexibility. Moreover, recommendation algorithms depending on the retrieved information (after the rules fired) would be useful. Additionally, future work will also focus on security and interoperability issues. Besides these, a mobile application e.g. for a smartphone, can be implemented and integrated with the native context sensing devices (e.g. GPS, compass, etc.). In addition, experimental testing of the system is in progress by making PLIS+ public.

References

1. Lassila, O. Applying Semantic Web in Mobile and Ubiquitous Computing: Will Policy-Awareness Help? Semantic Web and Policy Workshop (2005)
2. D.Serrano, R. Hervás, J. Bravo. Telemaco: Context-aware System for Tourism Guiding based on Web 3.0 Technology. International Workshop On Contextual Computing and Ambient Intelligence in Tourism (2011)
3. N.Tryfona, D. Pfoser: Data Semantics in Location-based Services. Journal on Data Semantics III, vol. 3534, pp. 168–195 (2005)
4. lliam V. Woensel, S. Casteleyn and O. Troyer.Applying semantic web technology in a mobile setting: the person matcher," Web Engineering, pp. 506- 509(2010)
5. Van Woensel, W., Casteleyn, S., and Troyer, O. A framework for decentralized, context aware mobile applications using semantic web technology. In Proc. of the Confederated International Workshops and Posters On the Move to Meaningful Internet Systems. Springer, 88-97. (2009)
6. S. Steiniger, Moritz N., Alistair E.: Foundation of Location Based Services. Lecture Notes on LBS (2006)
7. Viktoratos I., Tsadiras A. and Bassiliades N. *Personalizing Location Information through Rule-Based Policies* accepted for presentation to RuleML 2012
8. Yuh-Jong Hu, Ching-Long Y., Wolfgang L.: Challenges for Rule Systems on the Web. RuleML '09 Proc. of the 2009 International Symposium on Rule Interchange and Applications (2009)
9. Paschke A., Kozlenkov A., Boley H.: A Homogenous Reaction Rule Language for Complex Event Processing, 2nd International Workshop on Event Drive Architecture and Event Processing Systems (EDA-PS 2007), Vienna, Austria, (2007)
10. Ernest Friedman-Hill: Jess in Action.Rule-Based Systems in Java. Manning Publications. ISBN-10: 1930110898, pp. 32-33(2003)
11. Gregory Sherman: A Critical Analysis of XSLT Technology for XML Transformation. Senior Technical Report. (2007)
12. S. Liang, P. Fodor, H. Wan, M. Kifer: OpenRuleBench:An Analysis of the Performance of Rule Engines. WWW 2009 Madrid(2009)
13. E. Kontopoulos, N. Bassiliades, G. Antoniou, A. Seridou: Visual Modeling of Defeasible Logic Rules with Dr-VisMo. Intern. Journal on Artificial Intelligence Tools, 17(5), pp. 903–924, 2008.