Proactive Mobile Learning on the Semantic Web

Rachid Benlamri[1], Jawad Berri[2] and Xiaoyun Zhang[1]

[1]Department of Software Engineering, Lakehead University
955 Oliver Rd, Thunder Bay, Ontario, P7B 5E1, Canada

[2] Department of Computer Engineering, Etisalat University College
PO Box 573 Sharjah, United Arab Emirates

Abstract. Flexible and personalized instruction is one of the most important requirements to next generation intelligent educational systems. The intelligence of any e-learning system is thus measured by its ability to sense, aggregate and use, the various contextual elements to characterize the learner, and to react accordingly by providing a set of customized learning services. In this paper we propose a proactive context aware mobile learning system on the Semantic Web. The contribution of this work is a combined model using both a probabilistic learning technique and an ontology-based approach to enable intelligent context processing and management. The system uses a Naïve Bayesian classifier to recognize high level contexts in terms of their constituent atomic context elements. Recognized contexts are then interpreted as triggers of actions yielding a Web service composition. This is achieved by reasoning on the ontological description of atomic context elements participating in the high level context.

1 Introduction

Research work in the field of mobile learning [1][2][3][4][5] has shown that the educational potential of mobile technologies is driven by the continuing expansion of broadband wireless networks and the capacity of the new generation of cellular phones. However, the utilization of these technologies for educational purposes has been sparsely explored and many problems related to: context acquisition and management, conceptual knowledge modeling for personalized instruction, and adaptive information discovery remain unresolved. This paper contributes towards this direction, aiming at using the evolving semantic web and mobile computing to enable context-aware learning which delivers adaptive instructional resources on a learner's schedule. Context-aware learning is a critical support mechanism for educational institutions and organizations to compete in the new economy. Today's global market requires adaptive, fast, just-in time, and relevant learning processes that can be initiated by user profiles and business demands [6].

In this paper we propose an integrated approach to context modeling and reasoning based on Naïve Bayesian classifiers and ontological structures. First, higher-level contexts are recognized using a Naïve Bayesian classifier. Then, ontology-based

Benlamri R., Berri J. and Zhang X. (2007).
Proactive Mobile Learning on the Semantic Web.
In *Proceedings of the 1st International Joint Workshop on Wireless Ubiquitous Computing*, pages 63-73
DOI: 10.5220/0002434400630073

reasoning with the recognized contexts triggers actions yielding Web service composition that are customized to learner's context, needs, and preferences. The contextual information used in the personalization process encompasses all elements that characterize the learner's interaction, task at hand, the resources on which the Web services are to be performed, and surrounding environment.

The remaining of the paper is organized as follows. Section 2 describes background knowledge and related work. Section 3 describes context representation and modeling schemes. In section 4, we describe the higher-level context recognition process. Section 5 presents the framework for ontology reasoning and Web services composition to generate adaptive learning services. Finally, conclusions are drawn and further research work is suggested.

2 Background and Related Work

A considerable amount of research in knowledge-based and intelligent e-learning systems is now moving towards ontology-based context acquisition and management for personalized learning [7][8][9][10]. The main issues and challenges are however related to the ability of such systems to model and consistently reason with high level contexts at the semantic level. Although, some research attempts were made to solve some of these problems [9][10][11][12], the shortcoming of most of these efforts is their limitations to specific context elements and specific learning scenarios. General-purpose modeling and reasoning with context is a complex problem, and much research work is needed before achieving any real progress in this field. Most developed learning systems restrict the use of ontology relations and rules to describing and adapting content and sequencing of learning material according to some sensed context. However, little contextual semantics has been embedded in the ontology itself.

Other approaches to context modeling have also been considered. McCalla [13] has introduced an approach to learning design where learners' models are attached to Learning Objects (LOs) they interact with, and useful learning patterns are then derived by mining those models. The problem with this approach is its limitation to context that can be inferred from the learner's profile only, ignoring other type of context. Stojanovic et al. [6] however, have extended ontology usage to describe content, context, and sequence of learning material. Content-ontology was used for checking consistency as well as searching and navigating repositories of LOs. Context-ontology was used to present learning material in various learning contexts. However, learning style ontology was used to describe the way knowledge can be dynamically connected to adapt to learners' cognitive needs and preferences. Sets of relations, rules and axioms have been separately defined for each type of adaptation. The shortcoming of this approach is that efficient modeling of mobile learning scenarios would require the definition of atomic context elements at the semantic level and the use of the various ontologies in an orthogonal way. This is due to the fact that context, content, and learning styles are semantically inter-related aspects of cognitive learning [14]. This paper explores such a new dimension. The emphasis is on context discovery and its semantic modeling and management. Mobile users equipped with wireless devices go through several contextual changes as they move around in physical and social surroundings. These contextual changes could be used to drive ontology navigation and reasoning for better modeling of mobile learning scenarios.

Another challenging aspect addressed in this paper is automation of metadata generation for mobile learning. Metadata provides a common set of tags for describing, indexing, searching, and reusing learning materials on the Web in an interoperable way [14]. However, it is really difficult to create and maintain metadata rich enough to meet the diverse and ever changing needs of potential mobile learners. Mobile learning requires additional metadata to capture context. In this study, an attempt is made to solve this problem by defining contextual information at three hierarchical levels – atomic context – composite context – and higher-level context. Atomic context elements are sensed from the learner's interaction, task at hand, the used mobile-device, and the surrounding environment. These are then grouped into four composite context classes – learner context – activity context – device context and environment context. Composite contexts are further aggregated to build meaningful time-stamped higher-level contexts which are matched against context classes describing typical learning scenarios. Context classes are simply built from previously sensed similar higher-level contexts that have exhibited high degree of confidence. Matching higher-level contexts against these context classes is performed using a Naïve Bayesian classifier. The Naïve Bayesian classifier technique is used to cope with the uncertainty embedded in most sensed atomic contextual elements. Recognized contexts are then interpreted as triggers of actions that are translated into Web service compositions. This is achieved through ontological descriptions and reasoning with higher-level context.

Fig. 1 describes the overall system architecture which consists of four main components – context acquisition and aggregation – context recognizer – ontology reasoning engine – and Web-service composer. The context acquisition and aggregation component controls the user's interaction with the system and senses atomic context information from different sources. These are then aggregated into domain related contexts. Mobile learners go through continuous contextual changes as they move in their environments. It is the context acquisition and aggregator's job to communicate and update such changes yielding new contexts. The context recognizer identifies the aggregated contexts by matching them against well defined context classes stored in a context repository. The recognition process is performed using a Naïve Bayesian classifier. The context recognizer also allows for newly formed context classes to be added to the context repository.

The third component of the system is an ontology reasoning engine which uses the recognized higher-level contexts to customize learning services. Two ontologies are used to perform such a task – device/environment ontology – and domain ontology. The former is used to generate metadata that is used to discover Web-services that can run in the learner's device/network environment. However, the later is used to customize the learning content and the learning sequence according to the learner's current activity, background and preferences. This requires an ontological description and interpretation of higher-level contexts in terms of their constituent atomic context elements. Finally, the Web-service composer uses the generated device/environment metadata and the inferred learning concepts' sequence to compose Web-services accordingly.

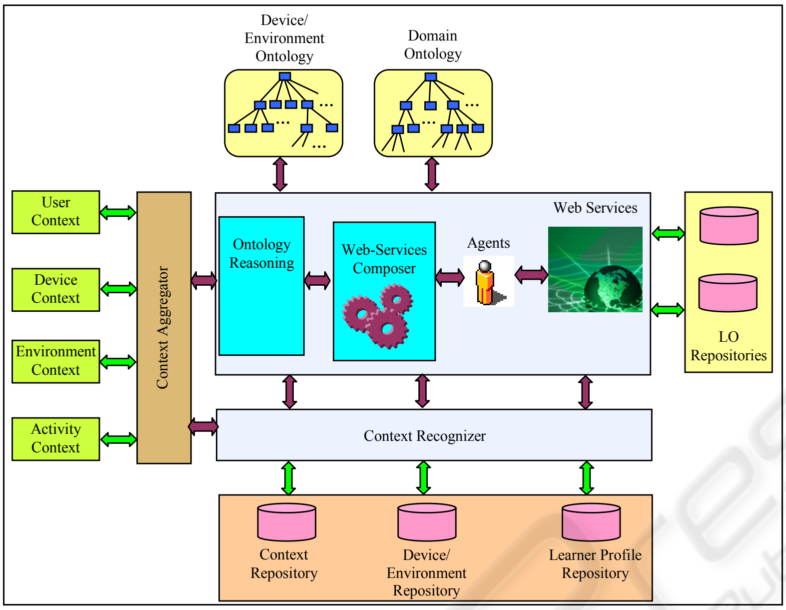

Fig. 1. System Architecture.

3 Context Acquisition and Aggregation

Contextual information used in this study is defined at three hierarchical levels – atomic context – composite context – and higher-level context. At the lower level, atomic contextual elements consist of the basic information describing the learner's profile, the current learner's activity, the used mobile device, and the surrounding environment. These can be either direct or indirect atomic contextual elements. Direct atomic contextual elements are those that can be directly sensed from the user interaction with the system and may originate from different sources such as the used device (i.e. device type, communication protocol), the task at hand (i.e. current learner's activity), and the surrounding environment (i.e. location, time, wireless network, network security). Indirect atomic contextual elements are however those elements that can be indirectly inferred from the direct atomic context elements. Inference of indirect atomic context elements is performed by the context aggregator relying on the device/environment repository and the learner profile repository. For instance, information such as device's operating system, device memory, and screen resolution of a specific mobile device which is previously stored in a device repository can be inferred using the atomic context element *device-type*. Similarly, other information related to the learner's pre-requisite knowledge, previously accessed services, and learner's preferences can be inferred from the learner profile repository. The use of indirect contextual elements aims at reducing the amount of contextual information that has to be sensed from the learner's interaction, device, and surrounding environment, which significantly speedup the context recognition process.

An atomic context element c_i is defined by:

$$c_i = (c_{iv}, c_{ip}) \tag{1}$$

where c_{iv} is the context value, and c_{ip} is the probability of context c_i of value c_{iv} being part of a higher-level context. The context value c_{iv} , as shown below, can be either a specific value (i.e. device type, learner identifier), a binary value (i.e. whether the used device is browser-enabled or not, secured/non-secured wireless network), or a value within a predefined range (i.e. network bandwidth, screen resolution).

$$c_v = \begin{cases} specific\_value \\ binary\_value \\ value \in [v1..v2] \end{cases} \tag{2}$$

Composite contextual elements are aggregates of atomic context elements describing a specific context type. There are four context types – c_L learner context – c_D device context – c_E environment context – and c_A activity context. Each of which is defined by:

$$c_{composite} = \left\{ \sum_{i=1}^{p} c_{i-direct} \cup \sum_{j=1}^{q} c_{j-indirect} \right\} \tag{3}$$

Finally, higher-level contexts consist of four-tuples $C_t = (c_L, c_D, c_E, c_A)_t$ which are built out of configurations of composite context elements sensed at time *t* and which characterize typical learning scenarios in a specific domain. Classes of higher-level contexts are defined at the ontological level in that they can be interpreted directly as triggers of learning actions implemented as Web service compositions.

4 Context Recognition

While ontologies have the ability to communicate context information by naming different concepts in machine readable fashion and allowing for the use of everyday words and concepts when interacting with the technology, they are unable to efficiently recognize learners' context. This is because the mapping between the defined concepts and the sensed real world atomic context elements is not so straightforward due to the *uncertainty* embedded in some atomic context elements. The mapping fails because ontologies do not handle *uncertainty*. They rather rely on well defined logic which assumes all information required to make a logical decision is available and produces either true, false or undeterminable statements. Uncertainty on the other hand produces similar statements but with degrees of truth or falseness [15]. To cope with uncertainty, higher-level contexts are recognized using a Naïve Bayesian Classifier. Bayesian Classification is a probabilistic learning technique where prior knowledge can be combined with observed data. The aim is to recognize a currently observed context state against a set of learned context classes. The input to the classifier is thus a set of sensed/observed atomic context elements which describe the user's context at a given instant of time, while the output is a learned context class. The classification process is thus performed with no user intervention or understanding required. The Bayesian classification also makes the implicit assumption that the data being handled is noisy and can tolerate any missing pieces of information. One difference between the Bayesian classification and the ontology approach is that once the ontology is defined then it can be available immediately whereas in the Bayesian classification approach each context has to be experienced at least once before being recognized again [15].

Let *X* be a current context whose class label is unknown, and let *H* be a hypothesis that *X* belongs to context class *C*, the classification problem consists of determining *P(H/X)* that is the probability that the hypothesis holds given the observed context *X*. This is defined by:

$$P(H|X)=\frac{P(X|H)P(H)}{P(X)} \tag{4}$$

where :

- *P(H)* is the prior probability of hypothesis *H* (i.e. the initial probability before we sense the current context and reflects the background knowledge).
- *P(X)* is the probability associated to the current context.
- *P(X|H)* is the probability of observing the context *X*, given that the hypothesis holds.

The above Bayesian model assumes that the observed context elements are related and depend on each other, and therefore, requires initial knowledge of many probabilities, as well as, significant computational cost. However, since most sensed atomic context elements are independent, the above model can be further simplified by applying the Naïve Bayesian classifier which is defined by:

$$P(X|C_i)=\prod_{k=1}^{n} P(x_k|C_i) \tag{5}$$

Where: C_i is a context class, and the set of x_ks are the atomic context elements forming the higher-level context *X* as defined in section 3.1.

The Naïve Bayesian classifier greatly reduces the complexity of the model, as well as its computational requirements. The context recognition problem is solved by assigning the current context *X* to the class C_k that satisfies the following condition:

$$X \in C_k \left| P(X|C_k)=\underset{i=1..m}{Max}\{P(X|C_i)\cdot P(C_i)\}\right. \tag{6}$$

where *m* is the number of recognized context classes.

Fig. 2 describes the context acquisition and recognition cycle. First, direct-atomic context elements are sensed, these are then used to infer related indirect-context elements. Next, the Naïve Bayesian classifier is applied to recognize the associated higher-level context-class, and finally, changes to the learner's context are sensed and a new context recognition cycle is performed. It should be noted here that the contextchange detection process significantly speedup the recognition time of successive high level contexts. This is because we just infer the indirect-atomic contexts of those context elements that have undergone some changes. The subset of newly observed context elements designated by $C_{changes}$ is defined by:

$$C_{changes} = (c_L, c_D, c_E, c_A)_{t_{i+1}} \setminus (c_L, c_D, c_E, c_A)_{t_i} \qquad (7)$$

where "\" means set subtraction.

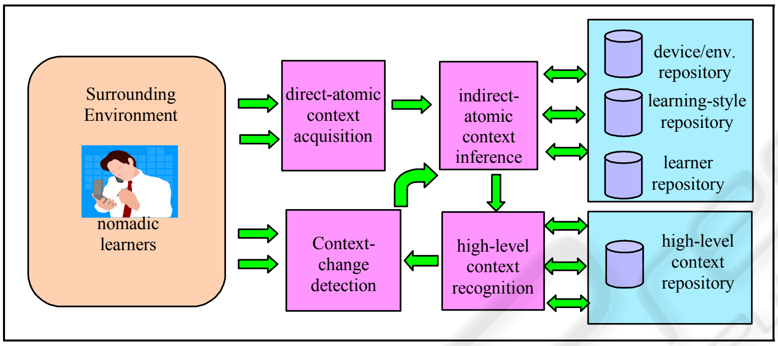

Fig. 2. Context Acquisition and Recognition Cycle.

5 Ontology Reasoning and Web Service Composition

Recognized higher-level contexts are fed to the ontology reasoning engine in order to customize learning services based on the learner's context, preferences and background. Reasoning with recognized higher-level contexts is performed using the two ontologies – device/environment ontology – and domain ontology. A set of ontological rules are applied to the device/environment ontology to infer the computing resources and the operational environment features compatible with the used mobile device and its surrounding environment. We call this process, *context-driven resources adaptation.* The output of this reasoning process is a set of metadata that will help discovering the Web services that can run into such an operational environment. The inference rules that are built around the domain ontology however are used to provide the learner with a learning sequence and content tailored to his/her current activity, previous background and preferences.

The two ontologies are coded in the Web Language Ontology – OWL; and the inference engine is implemented in Rule Markup Language – RuleML. Metadata derived from the ontology reasoning process is compliant with the IEEE-LTSC Learning Object Metadata (LOM) specification which is coded in XML. In particular, the XML description of both the inferred learning concepts and the device-related operational environment are used for Web services discovery. However, the inferred learning sequence which we call in this paper *domain-context* (i.e. the order of learning concepts inferred using the properties and relations between the domain-ontology's classes [16]) is used for Web service composition. This is described in OWL-S. A domain context represents a control structure that makes it possible to adapt the domain knowledge to a particular higher-level context. This adaptation is facilitated by the ontology O_M for a given domain M. O_M is defined by $O_M = (Co_M, R_M)$, where $Co_M = \{co_1,...,co_n\}$ is a set of concepts and $R_M = \{r_1,...,r_q\}$ is an ordered set of rules defined as follows: $p(co_1...co_j) \rightarrow_{r_s} q(co_k...co_l)$, where p and q are predicates reflecting respectively the factual information and the resulting one based on the inferential rule r_s.

The semantic of the ontological links is obtained by the rules in R_M. These rules are prioritized to reflect their importance or abstraction levels in a given knowledge taxonomy. For example, if the sensed higher-level context reflects a time-constrained learning scenario, one would like to focus only on say "the necessary-part-of" rules of the ontology to get a quick abstraction on the general structure of the requested knowledge. In a less time-stringent learning scenario however, this abstraction could further include the "part-of", and/or "case-study" rules, etc. These knowledge-supporting rules generate additional concepts of the ontology in multi-level clusters which are used to infer a progressive knowledge based on the learners' context denoted by C_L and the activity context denoted by C_A as described in section 3.

A software agent as shown in Fig. 1 is spawned at the server side to supervise a learning session for each learner. The agent typically represents the learner on the Semantic Web. The agent successively invokes the inference engine to get the current learner's focus, then discovers, composes, and invokes the chosen Web services accordingly.

To illustrate the main functions provided by our framework, we provide the following example ontologies describing a C++ programming course as a domain ontology, and a device/environment ontology. These are shown in Fig. 3 and Fig. 4 respectively.

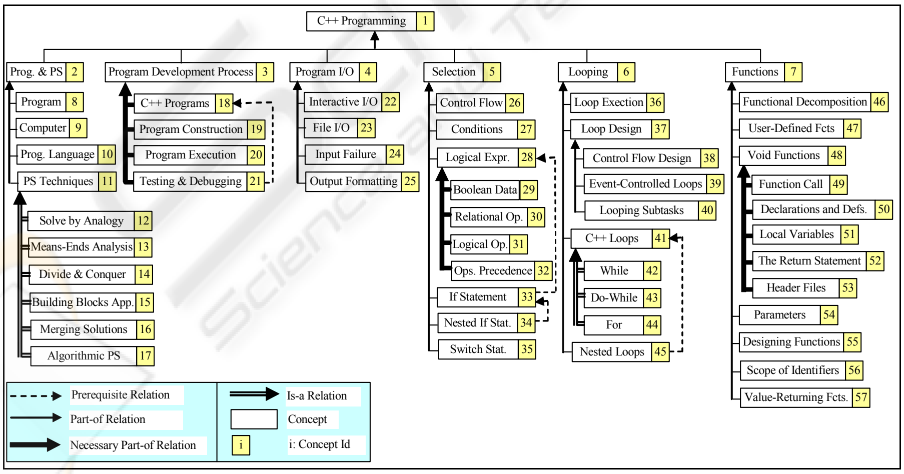

Fig. 3. Ontology for C++ Programming Course.

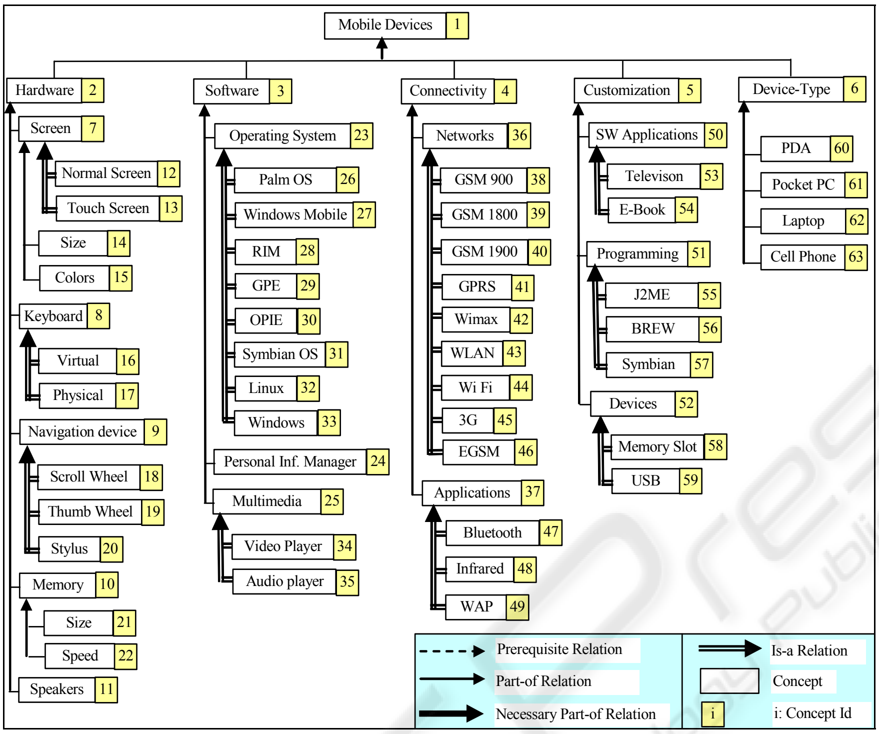

Fig. 4. Device/Environment Ontology.

A fragment of the ontology shown in Fig. 3, describing concept 3 "*Program Development Process*", is described in OWL in Fig. 5. The OWL definition of the semantics of the different relationships used in the C++ programming ontology is also given in Fig. 5.

Details about the rules used by the ontology reasoning engine to customize the learning sequence can be found in our previous work [17].

```
<owl:ObjectProperty rdf:ID="NecessaryPartOf">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <owl:inverseOf rdf:resource="#hasNecessaryPart"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isPartOf">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <owl:inverseOf rdf:resource="#hasPart"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="is-a">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <owl:inverseOf rdf:resource="#has"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isPrerequisiteOf">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
     <owl:inverseOf rdf:resource="#hasPrerequisite"/>

<owl:Class rdf:ID="Program Development Precess_3">
  <rdfs:subClassOf rdf:resource="#C++ Programming_1"/>
  <owl:disjointWith rdf:resource="#Prog and PS_2"/>
  <owl:disjointWith rdf:resource="#Program I/O_4"/>
  <owl:disjointWith rdf:resource="#Selection_5"/>
  <owl:disjointWith rdf:resource="#Looping_6"/>
  <owl:disjointWith rdf:resource="#Functions_7"/>
</owl:Class>

<owl:Class rdf:ID="C++ Programs_18"/>
  <rdfs:subClassOf rdf:resource="#Program Development Process_3"/>
  <rdfs:subClassOf>
    <owl:Restriction>
          <owl:onProperty rdf:resource="#isNecessaryPartOf"/>
          <owl:allValuesFrom rdf:resource="#program Development Process_3"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
          <owl:onProperty rdf:resource="#isPrerequisiteOf"/>
          <owl:allValuesFrom rdf:resource="#Testing and Debugging_21"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <owl:disjointWith rdf:resource="#Program Construction_19"/>
  <owl:disjointWith rdf:resource="#Program Execution_20"/>
  <owl:disjointWith rdf:resource="#Testing and Debugging_21"/>
   </owl:Class>
```

Fig. 5. Fragments of OWL description of the C++ Programming ontology.

6 Conclusions

In this paper, we proposed a proactive mobile-learning system on the Semantic Web. We argued that a probabilistic learning model is more suitable that an ontology-based approach for context recognition. This is mainly due to uncertainty embedded in some atomic contextual information. Higher-level recognized contexts are however described at the semantic level using ontology rules and axioms. The ontology reasoning process allows the system to react to any observed contextual changes by interpreting the newly sensed contexts as triggers of actions yielding a Web service composition. We are currently implementing a prototype of our framework as part of our personalized-learning provision project.

Acknowledgements

This work is funded by Lakehead University and NSERC under a startup research grant.

References

1. L. Low and M. O'Connell, "Learner-centric design of digital mobile learning", Proc. of the On-line Learning and Teaching Conference, Brisbane, Australia, September, (2006) 71-82.
2. M. Milrad, "Mobile Learning: Challenges, Perspectives and Reality", Book chapter in Kristóf Nyíri (ed.), Mobile Learning: Essays on Philosophy, Psychology and Education, Vienna, (2003) 25-38.
3. B. Alexander, "Going Nomadic: Mobile Learning in Higher Education", EDUCAUSE Review, 39(5), (2004) 28–35.
4. L. Chittaro, "Visualizing Information on Mobile Devices", IEEE Computer, March (2006) 40-45.
5. G. Vavoula and M. Sharples, KLeOS: A personal Mobile Knowledge and Learning Organisation System", Proc. 2nd IEEE Int. Workshop on Wireless and Mobile Technologies in Education, Sweden, (2002) 152-157.
6. L. Stojanovic, S. Staab and R. Studer, Elearning based on the semantic web, Proc. of the World Conf. on the WWW and Internet (WebNet2001), Orlando, Florida, USA, (2001) 1174-1183.
7. R. Mizoguchi and J. Bourdeau, Using Ontological Engineering to Overcome Common AI-ED Problems", Int. Journal of Artificial Intelligence in Education, 11 (2), (2000) 107-121.
8. L. Aroyo and D. Dicheva, Courseware Authoring Tasks Ontology", Proc. Int. Conf. on Computers in Education, Los Alamitos, CA, (2002) 1319-1321.
9. J. Kay and A. Lum, Ontology based User Modeling for the Semantic Web, Proc. of the Int. Workshop on Personalisation on the Semantic Web, Edinburgh, UK, (2005) 14-23.
10. J. Wang et al., "Personalized Knowledge Service Framework for Mobile Learning", Proceedings of the Second Int. Conf. on Semantics, Knowledge, and Grid (SKG'06), November, (2006) 102-103.
11. I. Tirrellil et al., "Fractal Adaptive Web Service for Mobile Learning", Int. Journal, of Emerging Technologies in Learning, 1(2), (2006) 25-35.
12. P. Veith, "Conception and Development of Reusable and Modular Mobile Content", Proc. of Int. Conf. on Mobile Learning mLearn'2005, Cape Town, 25-28 October, (2005), Retrieved from <http://www.mlearn.org.za/CD/ > Dec. 2006.
13. G. McCalla, The Ecological Approach to the Design of E-learning, Environments: Purpose-based Capture and Use of Information about Learners, Journal of Interactive Media in Education, 7, (2004) 1-23.
14. L. Aroyo and D. Dicheva, The new challenges for e-Learning: The educational Sementic Web, Educational Technology and Society, 7(4), (2004) 59-69.
15. J. Flanagan, Unsupervised clustering of symbol strings, in Intl' Joint Conference on Neural Networks,IJCNN'03, Portland Oregon, (2003) 3250–3255.
16. R. Benlamri, J. Berri and Y. Atif, A Framework for Ontology-aware Instructional Design and Planning, Journal of E-Learning and Knowledge Society, 2(1), (2006) 83-96.
17 J. Berri, R. Benlamri and Y. Atif, Ontology-Based Framework for Context-aware Mobile Learning, Proc. Int. Conf. on Wireless Communication and Mobile Computing, Vancouver, Canada, (2006) 1307-1310.