# Architecture of a Context-aware and Adaptive Learning Schedule for Learning Java

Jane Yau and Mike Joy
*Department of Computer Science, University of Warwick, UK*
*{j.y-k.yau; m.s.joy@warwick.ac.uk}*

## Abstract

*Novice programmers require large amounts of time and motivation to learn an object-oriented programming language such as Java. In this paper, the architecture of our Context-aware and Adaptive Learning Schedule (CALS) tool is described. The tool has been designed to focus initially on supporting first year computer science undergraduate students to become more proficient Java programmers, and makes use of a learning schedule, where the learner inputs their daily activities. Based on this information, the tool is able to automatically determine the contextual features such as the location and available time. The appropriate learning materials are selected for the students according to, firstly, the learner preferences (such as learning styles), and secondly the contextual features (such as the level of concentration).*

## 1. Introduction

It has been noted that the unique nature and characterization of mobile learning has become very difficult to define [1]. From the learner's perspective, mobile learning is *"any sort of learning that happens when the learner is not at a fixed, pre-determined location, or learning that happens when the learner takes advantage of learning opportunities offered by mobile technologies"* [2]. Another interpretation of mobile learning which also focuses on the learner is that learning is completed across contexts and it involves interacting with either portable or fixed technology [3]. Learning was also described as an "*intrinsically mobile activity*" [4]. Given the potential use of mobile technologies for facilitating learning anywhere and anytime, learners are able to make use of idle time that they have, for example, when waiting for public transport or meetings, in between lectures, and most significantly, traveling or commuting to and from work or university; time can be used more productively in terms of learning in this way [5].

Two characteristics of *context* have been described by Laerhoven [6] as *activity* and *environment*. The task that the user is performing at the moment is described by the activity, and focuses on the user of the device and his/her habits. The physical and social surroundings of the user are described by the environment, such as the current location and movements in the environment etc. The increasing use of mobile applications in varying contexts, locations and surrounding environments means that if these applications were made context-aware, then contextually relevant information from the devices can be transferred to the user [7]. These context-aware applications are described as "*more attentive, responsive and aware of their user's identity and their user's environment*" [8] and hence can derive the user's needs implicitly in the context which surrounds him/her at any point in time. It was noted that the purpose of context-awareness is to facilitate learning on mobile devices [9]. Sharifi *et al.* [10] argued that mobile applications must have context-awareness and personalization as integral parts of the application, and adaptive user interfaces must be generated for the learner, firstly, to maximize learning potential at different locations, and secondly, to decrease the limitations of mobile devices such as the usability of the small screens [9,11].

In this paper, we describe the architecture of our Context-aware and Adaptive Learning Schedule (CALS) tool in detail. Our chosen application domain of learning is the Java programming language and learning materials in the form of learning objects will be used in our application. Our tool consists of the following five main components – (Note that the original high-level pedagogical framework of our model was first proposed in [12])

- A Learning Schedule, for capturing and supporting the learner's daily activities.

Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)
0-7695-2916-X/07 $25.00 © 2007 IEEE

- A Learner Profile, for storing the learning preferences of the learner.
- A Learning Objects repository where Java learning materials are stored.
- A Learning Profile Adaptation module, which consists of learning styles adaptation, learning priorities adaptation and knowledge level adaptation sub-modules.
- A Context-aware Adaptation module, which consists of location-specific adaptation and time-specific adaptation sub-modules.

The remainder of this paper is organized as follows. Firstly related work is discussed, then the architecture of our CALS tool for Learning Java is described, and, finally our Conclusions and Future Work are given.

## 2. Related Work

Our system is related to three different categories of applications - Context-awareness and Adaptive, Learning Styles adaptation and Learning Objects for Java. There does not appear to be any application which combines all of these three areas of research.

An interactive context-aware mobile learning application has been developed for foreign students to learn English [13]. A schedule/calendar was not used in their system; rather the location and available time were inputted by the learner. The authors noted two reasons for this, firstly, students often do not conform to their schedule as observed by their absence from lectures, and secondly, their system was designed for use within short periods and mainly in between other activities which were not usually entered into the schedule. Given these two reasons, they noted that the location could not be detected automatically and accurately, and there was no way of inferring the available time that the learner has. Their system also includes two location-specific features which their activities adapt to accordingly, namely the concentration level of the learner and the frequency of interruption at a specific location.

Another context-aware mobile learning application which also combines the adaptation of learning styles was developed by Martín *et al.* [5]. Their system can be used individually or collaboratively. When the system is being used individually, each user's requirements are considered, for example, the system provides direct guidance for non-experienced students through a set of activities; whereas the authors considered that it would be more appropriate for more experienced students to navigate freely and select the activities that they prefer themselves. Novice learners are also recommended to perform extra basic activities and to acquire more practice before participating in collaborative activities with other students. When the system is being used collaboratively, four factors are taken into consideration for proposing activities to the user at a certain time. The first two are the user's and his/her partner's personal features, learning styles, preferences or previous actions. The latter two are the contextual features of the user and his/her partner, including their locations, idle time and the devices used.

Mungunsukh and Cheng [14] described some distinct psychologically-based learner characteristics specifically for programming; these are different types of degrees including *remembering*, *forgetting*, *understanding*, *knowledge*, and *interest*. A system was designed to increase the students' motivation and confidence of learning to program, as it was noted that novice learners may be unmotivated to self-learn programming. This system is automatically able to analyze and diagnose the error mistakes made by the students on their programs or programming techniques based on the learner's characteristics. Personalized feedback for students is then provided and the appropriate correct actions are suggested.

A learning object was defined by Adamchik and Gunawardena [15] as an "*integrated module containing the core text, code examples, review questions, supplementary material, and programming labs*" and a Learning Objects application for Java, known as Adaptive Book, was developed. This application is both an authoring environment for teachers allowing learning objects to be developed, distributed and managed, as well as a browser for learners to view the learning objects. Various functions can also be performed inside the browser such as adding bookmarks and hyperlinks, incorporating questions and answers, and highlighting and annotating text. A learner profile consisting of various learning objects can be created by an instructor, which the student can then download to work with. This approach was noted to be effective in allowing students to learn programming concepts quickly and for increasing the efficiency of communication between teachers and learners.

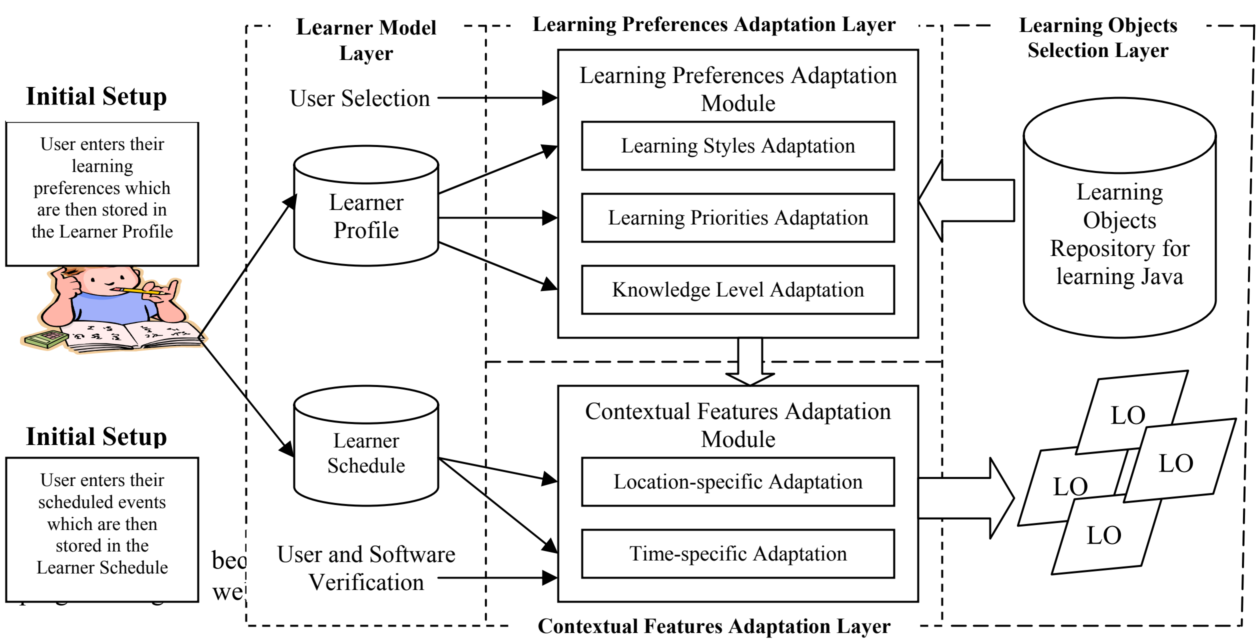
## 3. Architecture of our CALS tool for Learning Java

Our Context-aware and Adaptive Learning Schedule (CALS) system is fundamentally grounded on a number of theoretical foundations, namely the theories of contexts of mobile learning and computing and the theories of two different learning styles models – Dunn and Dunn [16], and Felder and Silverman [17]. These were established in [18]. Using a learning schedule to record the learner's learning events and activities can be motivating, in terms of learning, for the student as it helps them to fit or appropriate learning more naturally into their daily routines [19]. The term *appropriation* was defined by Waycott [20] to describe the process of integrating technology into a user's activities. Our learning schedule can also support students in making full use of their spare available times, regardless of where they are, when it is and the duration of their available times. These learning opportunities are created by the unique attributes of mobile devices [3,9].

Java programming aimed at first year computer science undergraduate students was selected as our learning domain because an object-oriented programming language such as Java is an essential part of a computer science degree course and Java is a popular language for study at university level. The programming modules also account for a significant percentage of the degree – students are required and expected to spend a large amount of time mastering it. they can proceed with Java programming, for example computer hardware basics, problem solving skills and fundamental concepts of programming. Many accounts of novice programmers experiencing difficulties associated with learning Java have been documented [14]. By integrating the learning of Java into the student's daily routine, our goal is to help them to become more proficient Java programmers.

Our system architecture, illustrated in Figure 1, is logically divided into four layers – Learner Model layer, Learning Preferences Adaptation layer, Contextual Features Adaptation layer and Learning Objects Selection layer. During the initial setup stage, a small amount of work is required for the learner to input their scheduled events and their learning preferences, which are then stored into the Learner Schedule and Learner Profile respectively. The learners are also responsible for keeping this information accurate and up-to-date. Login details are generated. By providing this information initially, the system will be able to automatically determine the learning preferences and contextual features of the learner at a given place and time; this saves time and effort for the learner to have to input this information whilst he/she is 'on the move'. For example, if the student would like to do some extra revision fifteen minutes prior to an exam, the tool will be able to select the appropriate learning objects for the learner for that duration of time, taking into consideration their learning priorities, learning styles and location.

**Figure 1 - System Architecture of the Context-aware and Adaptive Learning Schedule**



### 3.1 Learner Model Layer

This layer consists of two databases - the Learner Profile and the Learner Schedule, which are responsible for storing the students' learning preferences and their scheduled events. A User Selection option is included, so that the learner has the ability to override a particular learning preference at any time and/or location. For example, if the student's preferred learning style is verbal, however, he/she feels that he/she would prefer another type of learning material at a certain location, this can be overridden. This component is based on Kolari *et al.*'s work [7], which they noted, firstly, it is important that the adaptations do not over-determine the user's intention and that the learners should be able to override the automatically chosen contexts as sometimes it is not desirable to have an automatic recognition of context. The authors argued that there must be a balance between an automatic selection by the system and an active selection of context by the user and that users themselves should be allowed to select and activate contexts when necessary.

### 3.2 Learning Preferences Adaptation Layer

This layer consists of the Learning Preferences Adaptation Module, which contains sub-modules - Learning Styles Adaptation, Learning Priorities Adaptation and Knowledge Level Adaptation. The learning preferences of a learner are retrieved from the Learner Profile database and incorporated into the relevant sub-module for the appropriate learning objects to be chosen at a later stage. These cognitive user contexts are included in our system because many authors have acknowledged that these internal dimensions of context such as the user's goals, tasks, work content and personal events form an important area of context. However, this context area has often been neglected when designing and developing context-aware applications [8,9]. Matching the correct level of information according to the learner's most appropriate learning style can also create a more enjoyable and effective learning experience for the learner [9].

### 3.3 Contextual Features Adaptation Layer

This layer consists of the Contextual Features Adaptation Module, which contains sub-modules – Location-specific Adaptation and Time-specific Adaptation. Each of the contextual features are retrieved from the Learner Schedule database and incorporated into the relevant sub-module for the appropriate learning objects to be chosen at a later stage. As discussed earlier, there may be a possibility that the learner does not conform to his/her schedule, and that it may also be possible that the available time that the student has is no longer accurate because the student has a new appointment, for example. The first method in place to rectify this problem is to have a User Verification option. This prompts the user at the beginning of the learning session to indicate whether the location and the available time that the tool has retrieved is accurate. If this is not verified, the system then prompts the user to enter the current location and/or available time.

Another method which can also be used to detect discrepancies between the learner's stated location and his/her current location is to have an option for Software Verification. An appropriate positioning method such as GPS, network-based positioning and/or WLAN-based positioning can be used to verify the geographic location. The importance of obtaining the actual location of the user derives from the fact that the contextual features surrounding the location are different in various different places, and can affect the learner's ability to study such as their concentration level, which can be affected by the level of noise in the location or environment.

A number of methods for obtaining the noise level to determine the possible level of concentration that the learner has at a location have been considered. Firstly, a microphone sensor can be used to detect the noise level which can approximately indicate the level of concentration that the learner has in such an environment with that level of noise. Secondly, results obtained by Cui and Bull [19] can be used to map the concentration level of a learner to a certain type of location. In the authors' study, they had recorded the students' location of study and their corresponding chosen level of concentration, and the authors discovered that the chosen concentration levels in various types of location by different students were found consistent even though noise levels may have been different. For example, they had found that at a station or in a vehicle, students selected that their concentration levels were low; in a restaurant or in an outing, they chose the level to be medium; and at home or on campus, they chose the level to be high. These results indicate that students shared similar levels of concentration in the same location despite the varying levels of noise. This also suggests that there are other elements in the environment which could affect a student's concentration level such as movement.

### 3.4 Learning Objects Selection Layer

This layer consists of a Learning Objects Repository for learning Java and contains many learning objects which are provided to the Learning Preferences and the Contextual Features Adaptation Modules for the appropriate learning objects to be selected for the learner. The user's view can be simplified into the following list of actions to be performed by the user –

1. Student logs in and indicates that he/she would like to perform some learning.
2. *User Selection* - user indicates whether his/her learning styles should be considered.
3. *User Selection* - user indicates whether his/her learning priorities should be considered.
4. *User Selection* - user indicates whether his/her knowledge levels should be considered.
5. User Verification - user indicates whether the stated location is accurate.
6. User Verification - user indicates whether the stated available time is accurate.
7. Appropriate learning objects are selected.

## 4. Conclusions and Future Work

In this paper, we have described the system architecture of our CALS tool which is fundamentally grounded on a number of theoretical foundations and has four logical layers – learner model, learning preferences adaptation, contextual features adaptation, and learning objects selection. It is a generic tool for selecting the appropriate learning objects for learners based on their learner preferences and contextual features. Our initial prototype of the system will include learning objects for supporting first year computer science undergraduate students to learn Java (on mobile devices such as Pocket PCs). The implementation of the system is currently in progress and the effectiveness of the system will be evaluated both quantitatively and qualitatively, using a series of simulations and a small number of human users to work with our system. We will also extend our tool for incorporating other types of learning objects and/or materials.

## 5. References

[1] M. Sharples (Ed.), "Big Issues in Mobile Learning: Report of a workshop". In *Kaleidoscope Network of Excellence Mobile Learning Initiative*, 2006.

[2] C. O'Malley , G. Vavoula, J. Glew, J. Taylor, M. Sharples and P. Lefrere, "Guidelines for learning/teaching/tutoring in a mobile environment", *Mobilearn project deliverable*, 2003.

[3] I. Arnedillo-Sánchez, M. Sharples and G. Vavoula, "Beyond Mobile Learning Workshop", *Kaleidoscope, Trinity College Dublin Press*, 2007.

[4] G. Dettori, T. Giannetti, A. Paiva and A Vaz (eds.) "Technology-mediated narrative environments for learning", *Sense publishers*, 2006.

[5] E. Martín, R. Carro and P. Rodríguez, "A Mechanism to Support Context-Based Adaptation in M-Learning", *EC-TEL,* LNCS 4227, pp. 302-315, 2006.

[6] K. Laerhoven, "Online adaptive context- awareness starting from low-level sensors", *PhD Thesis*, 1999.

[7] J. Kolari, T. Laakko, T. Hiltunen, V. Ikonen, M. Kulju, R. Suihkonen, S. Toivonen and T. Virtanen, "Context-aware services for mobile users – technology and experiences", *VTT publications 539*, 2004.

[8] P. Prekop and M. Burnett, "Activities, Context and Ubiquitous Computing", *Computer Communications* 26*,* pp. 1168-1176, 2003.

[9] R. Beale and P. Lonsdale, "Mobile Context Aware Systems: the intelligence to support tasks and effectively utilize resources", *Mobile HCI*, 2004.

[10] G. Sharifi, R. Deters, J. Vassileva, S. Bull and H. Röbig, "Location-aware Adaptive Interfaces for Information Access with Handheld Computers", *Adaptive Hypermedia, Springer-Verlag, Berlin Heidelberg*, pp. 328-331, 2004.

[11] T. Rodden, K. Chervest, N. Davies, "Exploiting Context in HCI Design for Mobile Systems", *GIST Technical Report G98-1*, 1998.

[12] J. Yau and M. Joy, "A Context-aware and Adaptive Learning Schedule framework for supporting learner's daily routines", *International Workshop on Mobile Communications and Learning*, 2007.

[13] S. Bull, Y. Cui, H. Röbig and M. Sharples, "Adapting to Different Needs in Different Locations: Handheld Computers in University Education", *WMTE*, pp. 48-52, 2005.

[14] H. Mungunsukh and Z. Cheng, "An Agent Based Programming Language Learning Support System", *Conference on Computers in Education*, pp. 148-52, 2002.

[15] V. Adamchik and A. Gunawardena, "A Learning Objects Approach to Teaching Programming", *Information Technology: Computers and Communications*, 2003.

[16] R. Dunn and K. Dunn (1978) Teaching students through their individual learning styles: A practical approach. Reston, VA: Prentice-Hall.

[17] R. Felder and L. Silverman (1998) Learning Styles and Teaching Styles in Engineering Education, Vol. 78, No. 7, 674-681.

[18] J. Yau and M. Joy, "Context-aware and Adaptive Learning Schedule for Mobile Learning", *MULE Workshop, ICCE Conference,* pp. 17-24, 2006.

[19] Y. Cui and S. Bull, "Context and learner modeling for the mobile foreign language learner", *Science Direct, System 33*, pp. 353-367, 2005.

[20] J. Waycott, "The appropriation of PDAs as larning and workplace tools: an activity theory perspective", *PhD thesis, Open University*, 2004.


0-7695-2916-X/07 $25.00 © 2007 IEEE