The Computer Journal Advance Access published March 13, 2007

doi:10.1093/comjnl/bxm005

Context-aware Timely Information Delivery in Mobile Environments

Amit Thawani[1,*], Srividya Gopalan[1], V. Sridhar[1] and Krithi Ramamritham[2]
[1]Satyam Computer Services Ltd., Applied Research Group, Third Floor, SID Block, IISc Campus, Bangalore 560 012, India
[2]Indian Institute of Technology Bombay, Powai, Mumbai 400 076, India
**Corresponding author: Amit_Thawani@satyam.com*

In mobile environments, transmitting information relevant to an event along with notification of the event has been proven to be an effective means of providing revenue enhancing services. For example, a relevant advertisement can be displayed just before event notification; for instance, a product promotion by Beckham can be shown just before the notification of a goal scored by him. Challenges in achieving real-time search and delivery of information relevant to events as they occur include: (i) predicting the next event so that the appropriate information can be kept ready; (ii) finding information relevant to the context and content of the event; (iii) searching for and bringing the potentially needed information closer to the user location; and (iv) disseminating and displaying relevant information just before the actual event is notified to the user. In this paper, we propose a real-time information delivery system based on an event's context and content. The key features of our system are: (i) representing domains with event-based scenarios using statecharts; (ii) using a novel combination of history information and state transitions to predict events; and (iii) real-time delivery of information relevant to an event by prefetching and caching information based on the events predicted to occur in the near future. To illustrate our approach, we focus on the delivery of advertisements relevant to notified events as a specific case of delivery of information relevant to the events. Using an experimental setup, we measured the effectiveness of our approach as compared to a context-unaware approach to event prediction. Experiments demonstrate that our approach has a superior performance, resulting in: (i) better prediction accuracy; (ii) the delivery of the most relevant information most of the time; and (iii) an effective cache management in mobile devices.

Keywords: mobility, caching, context-awareness, performance

Received 20 March 2006; revised 18 December 2006

1. INTRODUCTION

Event notification and delivery services are usually based on user needs and preferences. An event notification service provides an opportunity to service providers to enhance their revenue by airing information relevant to events along with event notifications. In any media, for better Return on Investment (*RoI*), the information is usually shown before an event to make sure that the information delivered is viewed by users. Domain-specific event notifications (such as score update events in the domain of sports) are sent to users on a need and interest basis. Some information delivery systems do consider the relevance of the information, but relevance is limited to the overall domain and the entities involved and not the current event. Consideration is also not given to the current state of the domain; this can sometimes result in a negative impact of the information transmitted. For example, consider the impact of airing an advertisement endorsed by a popular player, after he fails to score a goal. Also, information becomes out of context if delivered along with an event notification that is delayed. Context-aware services in a mobile environment have been proposed while considering a user's contextual information or other context attributes such as location. In this paper, we propose a real-time information delivery system based on an event's context and content. The key features of our system are: (i) representing domains with event-based scenarios using statecharts; (ii) using a novel combination of history information and state transitions to predict events; and (iii) real-time delivery of information relevant to an event by prefetching and caching information based on the events predicted to occur in the near future. To illustrate our approach, we focus on the delivery of advertisements relevant to notified events as a specific case of delivery of information relevant to the events.

2. RELATED WORK

An event architecture for mobile computing that addresses two key requirements of terminal mobility and user mobility has been proposed in [1]. This approach, however, does not consider the context of the event or the limitations of the access device. An architecture for mobile content delivery systems by investigating representative usage scenarios has been proposed in [2]. This architecture is based on the publish/subscribe paradigm, which supports many-to-many interaction of loosely coupled entities. But context is limited to the subscription rules defined by the user. Moreover, this approach does not consider event context or device limitations. Some information delivery systems [3] that consider possibilities of pervasive advertising do consider relevance of the information but relevance is limited to the overall domain and the entities involved and not the current event.

Researchers have also addressed issues such as context-aware services while considering the context attributes of subscriber intention, location, user intent and emotion or device capabilities. A study of strategic parameters to determine service subscribers' intention to switch to a new service provider who offers personalized service is presented in [4]. A new method of personalizing mobile advertisement by considering not only the user location but also interest is proposed in [5]. Limitations of mobile devices, unreliability of the network and the different user's interest for the selection and delivery of multimedia information to mobile devices is addressed in [6]. A personalized information delivery system that reduces storage, bandwidth and processing power requirements and simplifies user interaction by using media conversion to generate presentations that fit the device and/or user capabilities is proposed in [7]. A novel system for delivering permission-based location-aware mobile advertisements to mobile phones is introduced in [8]. The design and implementation of a system that can be used for advertising by the mobile e-commerce user is described in [9]; the authors also reflect upon the possibility of using user emotion to increase effectiveness of the advertising. However, no explicit attempt has been made for timely delivery of such personalized information in a way to ensure the validity as well as usefulness of the content being delivered.

A survey of research on context-aware systems and applications, looking at the types of contexts used and models of context information, at systems that support the collection and dissemination of contexts, and at applications that adapt to the changing context is presented in [10]. A report on the challenges and opportunities of wireless advertising is presented in [11]. A study to identify factors that influence the effectiveness of a mobile messaging advertising campaign is presented in [12].

2.1. Motivating scenario

A sport enthusiast has subscribed to sports-related events with his mobile service provider. He is interested in soccer and is a fan of David Beckham. The European league soccer tournament is being played and sponsors for the tournament are Pepsi and Gillette. Event updates of the matches being played are being disseminated by the mobile service provider to interested users. As a revenue-enhancing service, service providers also disseminate information relevant to events and, as a specific case, advertisements of the sponsors are sent along with specific events within the game. Consider a match currently being played between Manchester United and Spain. The match is in its last few minutes and both the teams are at equal score. David Beckham of Manchester United playing at forward position gets a penalty corner and event sponsor Gillette and Pepsi would like their advertisement to be disseminated if the penalty corner results in a *scoring a goal* event or a *goal save* event.

An event is characterized by the entities involved in the event and actions performed by these entities. For example, *scoring a goal* event involves the entities: the player, the team, the place where the match is being held, etc., and the actions: scoring a goal, saving a goal, etc. *An entity is defined as a person, place or object of interest for a domain and its events. Since not all events provide an opportunity to air relevant information, we call certain events major events of the domain, which have a substantial significance value and provide opportunity to air relevant information.*

To have a better impact on the user and for a better RoI, if David Beckham were to score a goal at this point of time, the Gillette advertisement endorsed by David Beckham would be most relevant. Similarly, if the penalty corner results in a goal save, the Pepsi advertisement endorsed by the goal keeper would be most relevant. The current state of the game being a penalty corner state, the possible next major event can result in a *goal scored* state or a *goal saved* state.

Further, suppose Beckham has scored 99 goals so far in his career and if he scores a goal at this point of time, he will achieve the milestone of scoring 100 goals. The possible next event of the game resulting in the *goal scored* state also results in the major event *Beckham scoring 100 goals*. This involves two *major* events: *scoring a goal* and *Beckham scoring 100 goals*. Since *Beckham scoring 100 goals* is the result of the event *scoring a goal* over a period of time, we call *scoring a goal* event *point* event and scoring 100 goals an *aggregated* event. *A point event is defined as the occurrence of a particular event at a given point of time while aggregated event is defined as an event that is based on the occurrence of point events over a period of time.*

Also, it is possible that the Beckham scores a goal in the last minutes of the match but it does not result in a win for Beckham's team. In this case, the possible next states of the game are *goal scored* state and *opponent team winning the match* state.

2.2. Challenges

To determine and deliver relevant information along with each of the aforementioned events, we need to address the challenges of determining information relevant for the possible next events and making the information available closer to the user so as to be delivered before event notification. Determining information relevant to possible next events demands a successful modelling of the domain, predicting events given the current state of the domain, representing event and information content such that the relevance of the information can be easily established, and determining *aggregated events*, if any, and information relevant to them. Handheld devices have limited storage capabilities and do not allow storing of all relevant information for all events likely to occur. Making the information available closer to users coupled with limited storage capabilities of devices requires that information be prefetched and cached closer to the user.

2.3. Our contributions

In this paper, we propose a real-time context-aware information delivery system for mobile applications. The key features of our system are: (i) modelling domains with event-based scenarios; (ii) using a novel approach of considering state transitions and history information to predict events; and (iii) delivering and displaying information in real time with content relevant to the event. Domain knowledge has to be captured effectively to deal with events in that domain. *A domain typically consists of states, entities, events and actions. A state in the domain is characterized by the current context of the involved entities, events that lead to transition from another state to this state, and events that lead to transition from this state to another state. Actions between entities trigger events and these events either lead to transition to another state or an update to the current context of the entities involved.*

Our system retrieves, delivers and renders information with content relevant to the event in real time, by: (i) representing information and event content using metadata encoding the semantics of the content; (ii) associating each state with a set of appropriate metadata based on specific scenario instantiation and state significance in the domain model; (iii) predicting the next event for the involved entities based on the current event and active state in the domain model; (iv) finding event significance for the involved entities based on domain-specific rules as well as the current state; (v) finding *aggregated events* and their significance resulting from predicted *point events* based on domain rules; (vi) finding the most appropriate events for the current state while considering event significance of the involved entities as well as of *aggregated events*; (vii) finding relevant information for the most appropriate events based on semantic similarity of information and event description; (viii) disseminating the identified information to prefetch and cache in the in-memory database of a user's handheld device; and (ix) rendering of information on the user's handheld device.

Among the modelling approaches such as statecharts, graphs, state machines, etc., a statechart-based representation [13] of domains with event-driven scenarios is most appropriate since it allows a domain and its events to be represented as states and events associated with the states, it allows transition from any state in the hierarchy to any other state in any other hierarchy, and it allows each state to be represented as a composite state representing a sub-statechart. We enhance the generic statechart model so that each state has attributes, which are inherited by the sub-states, and representation of history information is allowed at each state. We use the Weka data mining tool [14] to derive rules (for predicting events) from history information, which is represented in the form of state attributes in a statechart. We use a novel combination of state transitions on events as well as the rules associated with each state to predict events.

As a proof of concept of our techniques, we focus on the problem of delivering relevant advertisements along with salient events to a mobile user who is 'watching' an ongoing cricket game. Our system models the domain of cricket using statechart realization in concurrent hierarchical finite state machine (CHSM) language [15]. The generic statechart implementation provided by the CHSM language is enhanced to accommodate the representation of metadata and history associated with the corresponding states. The history information of the domain and semantic significance of the states is maintained along with the state information in CHSM implementation. The predicted events are used to determine relevant advertisements and selected advertisements are cached in the mobile device. The experimental setup is used to evaluate our approach for computing: (i) prediction accuracy of events – for all the entities involved; (ii) effectiveness of replacement of advertisements in the cache of a mobile device; (iii) effectiveness of replacement of advertisements in the cache of mobile device considering *predicted future* events (event prediction for predicting not just the next event but subsequent events); (iv) the number of instances in which the most relevant advertisement is displayed to the user; and (v) the number of instances in which the second most relevant advertisement is displayed to the user.

For a specific example, consider Fig. 1 depicting a system involving a Nokia 6600 mobile with Symbian OS and in-memory database. The server side components comprise a statechart representation of the domain, the relevant information search component, information delivery component,

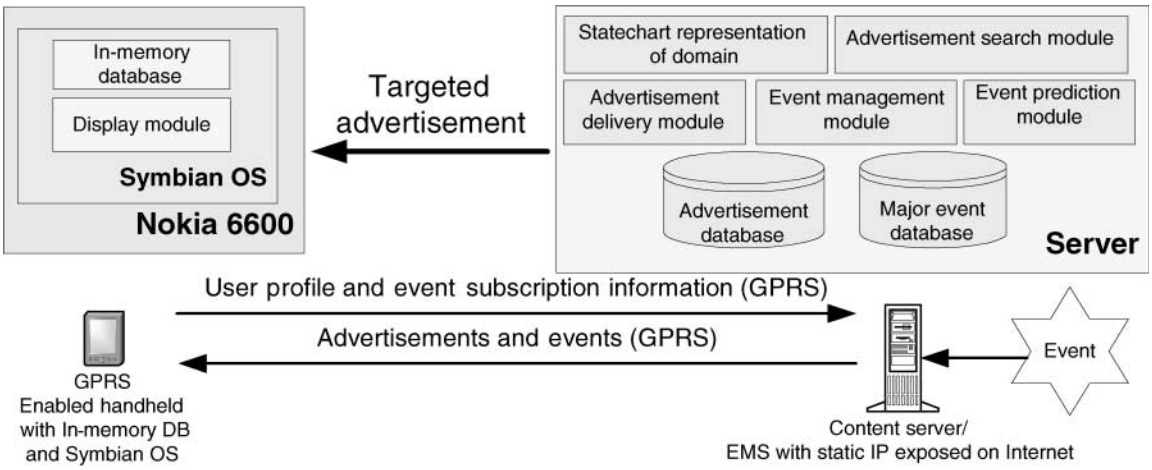

FIGURE 1. Advertisement delivery system.

event management component, event prediction component and information and event databases. The client side components comprises in-memory database to store relevant information and a display component to display the cached information.

Our implementation experience indicates the effectiveness of our modelling approach (using statechart), our event prediction approach (using mined rules) and use of metadata (attached to both events and information) for selecting relevant information. In our experiments, we measure the effectiveness of our context-aware approach as compared to an approach that does not exploit the context. From the results, we conclude that our approach performs better, resulting in better prediction accuracy, delivering the most relevant information most of the times, and effectively managing the cache in mobile devices.

The reminder of this paper is structured as follows: Section 2 introduces the issues real-time information delivery. Section 3 describes statechart-based domain modelling and Section 4 presents the realization of an information delivery system. Section 5 illustrates our approach using scenarios from the domain of cricket and explains the effectiveness of our approach described in results. We conclude in Section 6 and present future work.

3. ISSUES IN REAL-TIME INFORMATION DELIVERY

The infrastructure involved in providing event dissemination and delivery service by mobile service providers is shown in Fig. 2. For various domains such as gaming, entertainment, etc., in the real world, events are constantly generated and published. Events published can either be captured by edge servers in the network or by an event management system. Published events are communicated from edge servers to the event management system to be notified to subscribed users. Event notification to subscribers follows the path from the event management system to the gateway of the wireless

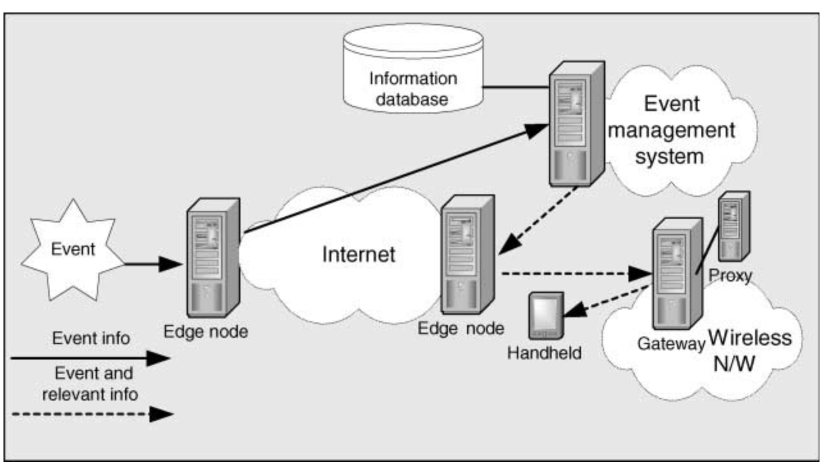

FIGURE 2. Information delivery infrastructure.

network either directly or using one of the edge servers on the Internet. Users are notified of published events on their handheld devices by disseminating events from the gateway to the handheld device.

Real-time information delivery relevant to events is a closely knit process with event notification service. Real-time information delivery uses the same infrastructure and follows the same dissemination path as event notification. Since event notification should be in real-time before the event turns *stale*, to disseminate information most relevant to events, the next events needs to be successfully predicted and relevant information determined in time.

Information to be disseminated to the users is stored in the information database connected with the event management system. On event prediction, relevant information is retrieved from the information database, processed to determine its destination based on event subscription information, cached at intermediate locations or on a user's mobile device and rendered on a user's mobile device. Storing relevant information on the mobile device prior to their need requires: (i) accounting for relevance and significance of information in terms of events as well as timeliness of delivery; (ii) replacing the currently cached information with more relevant information; (iii) replacement considering not only the next predicted event but also accounting for events that are likely to occur in the near future; and (iv) considering relevance ratings of the information to make the replacement decision.

4. STATECHART-BASED DOMAIN MODELING

Event-driven scenarios can involve multiple entities in a single event. Each of the involved entities can have significance for the event in consideration. Further, each of the entities is associated with a set of states, actions and events. To model states, actions and events specific to each of the entities, a statechart must be an instantiated. To find relevant information, our approach models the semantics of the current instance of state, the history of events associated with the current instance of the state, relevance attributes of the event within the state and event significance for the entity within the state.

We attempted modelling the domains of two games, namely cricket and soccer. Games, being highly event driven, are a good choice to evaluate statechart modelling of event-driven systems. We not only modelled the game domain but also modelled the typical scenarios of that domain to verify the appropriateness of using statecharts. Figure 3 describes the modelling of domains for cricket and soccer. The statechart model is based on representing the domain with composite states, start/end states and events representing transitions between these states. Each of the composite states is further decomposed to describe the sub-states, events and transitions to/from those states. For the domain of cricket we have described only the batting.

The cricket domain modelled using statechart considers: (i) two innings as composite states; (ii) each innings comprising composite states 'batting' and 'fielding'; (iii) 'batting' comprising 'batsman batting' and 'batsman waiting'; (iv) 'batsman batting' comprising 'batsman 1' and 'batsman 2'; (v) each batsman state comprising concurrent states for 'scores' and 'boundaries'; and (vi) event transitions of 'run', 'out' and 'innings completed' with respective states.

Similarly, the soccer domain considers: (i) a match consisting of two composite states for two halves of the match; (ii) each half-time state consisting of composite states 'teams'; (iii) each 'team' state consisting of concurrent states 'goal keeper' and '10 players'; (iv) 'goal keeper' consisting of 'goals' and 'saves'; (v) 'player' consisting of 'goals' and 'kicks'; and (vi) events such as 'red card', 'yellow card', 'foul', 'goal', 'save', etc., associated with the respective states.

Modelling games of cricket and soccer using statecharts provided proof of concept that domains with event-driven scenarios can be modelled using statecharts. This also provides an opportunity to associate attributes such as significance of events as well as history associated with the state.

5. DETAILS OF INFORMATION DELIVERY SYSTEM

Steps involved in realization of an information delivery system are described in detail next:

5.1. Semantic representation of event and information content

The semantics of events associated with the domain as well as relevant information for all the events of the domain are described by metadata. This metadata is associated with the respective event and information content and are used to identify the degree of relevance between two contents. There are various standards for metadata representation of content, such as Dublin Core, RDF, PRISM, CC/PP, etc. For experimental purposes, the event and information content is manually tagged with sample metadata to match for determining relevance. The events and information for the experimental setup are described manually. For the typical case of delivering advertisements relevant to events, the aforementioned information is stored in the advertisement and major event databases as shown in Fig. 1.

Example: In a game of cricket Sachin is the batsman. Sachin's current score is 10.

Event: Sachin scores 4 runs.

Metadata Associated with Event: <Sachin, India, Batting, Four>.

Information: Sponsor advertisement endorsed by Sachin (Advertisement shows Sachin hits a shot and India wins the match).

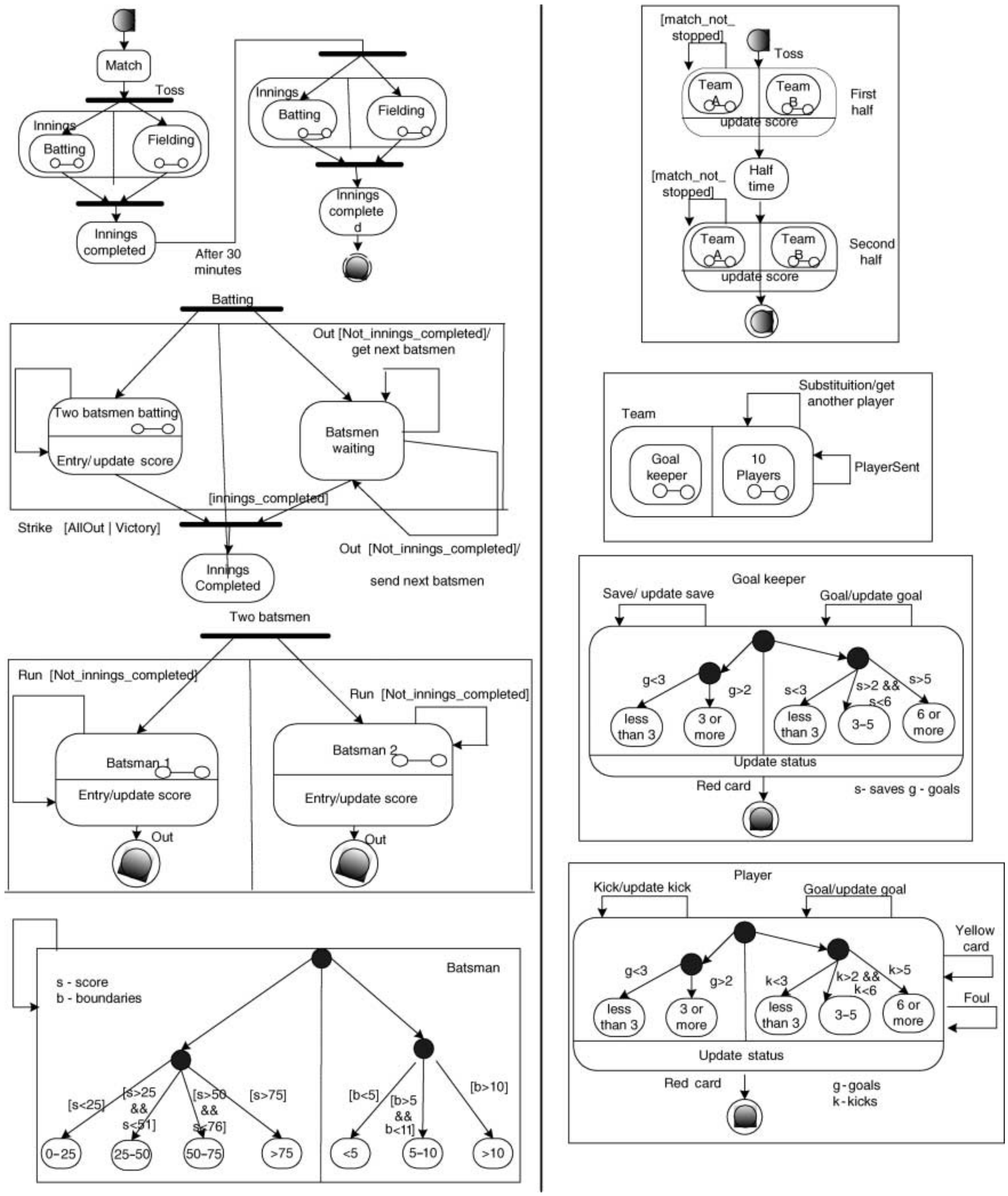

FIGURE 3. Modelling of cricket (left) and soccer (right) game domains.

Metadata Associated with Information: <Sachin, Cricket, Batting, Win, Cheer>.

5.2. Domain representation

The domain and the events along with the particular entities are modelled using statecharts and implemented using CHSM. For each of the entities involved, a statechart is instantiated representing the possible states the entity can transition to on the occurrence of events associated with it. Further, based on the instantiations, for each of the states, metadata describing the state and its significance are associated based on domain knowledge. For the typical case of delivering advertisements relevant to events, as shown in Fig. 1, the statechart representation of domain module is responsible for representing the domain using statecharts.

Example: Fig. 3 describes domain representation using statecharts for games of cricket and soccer. As depicted, the cricket domain consists of two innings with each innings consisting of a batting and a fielding side. The batting side has the two batsmen batting as well as the next batsman waiting. Each batsman scores some runs and hits some boundaries. Events such as innings complete, run scored, boundary scored, etc., govern the flow of the game.

5.3. Event prediction

Historical data for a domain is gathered, pruned and mined to be used for event prediction. We specifically consider the history of the entities for event prediction, using historical data to derive rules as well as to test data to predict events based on these rules. A database is created describing *major events*, and metadata associated with these events, which describe the relevance attributes including the semantics of the events, are also described. The events are mined using the Weka machine learning and data mining tool, and rules are derived for predicting events given the current event occurrence as input. The predicted events and their significance is further associated with the states of the instantiated statecharts. For delivering advertisements relevant to events, as shown in Fig. 1, the Event Prediction module is responsible for predicting events.

Example: Data from matches played by Sachin are mined to determine probability of a shot at each score when Sachin is batting (say probability of a shot when Sachin is at the score of 10). Further, for each of the scores, probabilities of shots resulting in major events are determined. To predict the next event in the game being currently played, based on the probabilities of major event occurrence as determined earlier, the event that has the maximum probability of occurrence is predicted (say Sachin's current score in the game being played is 10. If the probability of hitting four runs is greater as compared to other shots, a 4 is predicted).

5.4. Finding relevant information

A database of relevant information, along with the semantics and other relevance attributes described by metadata associated with the information is created. Given metadata associated with the predicted event and with the currently active state, and considering event significance of predicted events (both *point* and *aggregated*) for the entities involved, the most relevant information is determined by comparing the associated metadata. The information used is from sponsors who typically sponsor events in that domain to deliver advertisements relevant to events, as shown in Fig. 1, the advertisement search module is responsible for finding information relevant to events.

Example: For the example presented in Section 5.1, the information is considered relevant to the event as they have similar semantics as represented in associated metadata.

The Complete flow of the information delivery system-involving statechart representation of domains implemented using CHSM, semantic representation of event and information content using metadata, representation of attributes associated with the states using metadata, associating rules for event prediction derived from mining history data of events, and determination of information relevant to event content-is depicted in Fig. 4.

5.5. Information dissemination

The relevant information is prefetched and cached at the users' device. The cache contains information relevant for

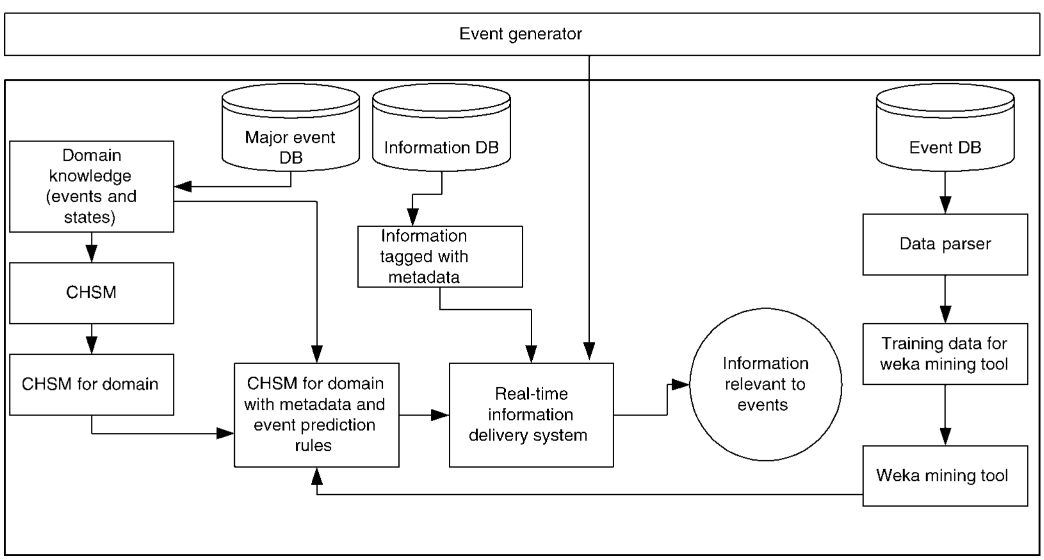

FIGURE 4. Flow of information delivery system.

not only the events predicted next but also for events likely to occur in the near future. The cache also contains information specific to the entities and to both point and aggregated events. Information is replaced in the cache on the basis of predicted events as well as of the significance associated with the events. Further, replacement also considers delivery deadline of point and aggregated events and the information relevant to them. For the typical case of delivering advertisements relevant to events, as shown in Fig. 1, the advertisement delivery module at the server is responsible for information dissemination, and the in-memory database and display module in the users' handheld device is responsible for prefetching, caching and rendering of information.

Example: Sachin is batting at a score of 10. The information relevant (advertisements) to the events predicted at the score of 10 and also for near future scores, such as 11–15, is determined as described. (Say, the predicted event at the score of 10 is 4 runs and at 14, 6 runs. The relevant information for events 4 and 6 runs is determined). This information is cached in the mobile device and shown to the user if an actual event matches the predicted event.

6. SCENARIO FROM CRICKET

We consider a game of cricket to implement the proposed approach. We model a player's game from the time he starts his play till his play ends. To evaluate our proposed system, we consider the play of two players 'Sachin' and 'Shoaib'. Historical data for one-day tournaments played by India and Pakistan (from 1999 to 2004) against other teams is used for event prediction. Specifically, we base our scenario on the history of Sachin's and Shoaib's play in these games. The data for training as well as test data for games played by India are downloaded from archives of one-day international matches from *www.cricinfo.com*. A ball-by-ball commentary of matches is downloaded and parsed to generate event data.

6.1. Typical scenario

Let us consider a cricket match being played between India and Pakistan and a user who has subscribed to sports-related events and is specifically interested in events related to cricket. In the last few minutes of a match India is chasing a score of 320. India's current score is 302 and Sachin is batting at 95. It is the last over of the match and Shoaib is the bowler. Considering the current state, sponsors' advertisements can achieve maximum impact if, on the occurrence of a particular next event resulting in a particular next state, an advertisement from a particular sponsor and endorsed by a particular player is displayed to the user, as depicted in Table 1.

For the above possible events of the match, at the current state, an advertisement endorsed by Sachin will be relevant if Sachin hits a four or a six. Similarly, an advertisement endorsed by Shoaib will be relevant if Shoaib bowls a dot ball (i.e. a ball is bowled but the batsman does not score) or takes Sachin's wicket. Consider the possible event Sachin hitting a six and scoring a century, since it results in the occurrence of two major events, i.e. 'hitting a six' and 'scoring a century'. The relevant advertisement for both the events should be cached and delivered.

Table 1. Event occurrence and resultant state

Next event	Next state	Sponsor	Advertisement endorsed by
Sachin hits a four	Indian team nearing victory	Airtel	Sachin
Sachin hits a six	Indian team nearing victory	Airtel	Sachin
Sachin scores a century	Indian team nearing victory	Airtel	Sachin
Shoaib bowls a dot ball	Pakistan team nearing victory	Pepsi	Shoaib
Shoaib takes Sachin's wicket	Pakistan team nearing victory	Pepsi	Shoaib

In the rest of this section, we first describe the off-line data preparation for experimentally evaluating our approach for the domain of cricket and then we present how this data is used in the on-line advertisement dissemination process to achieve dissemination of information relevant to events. We conclude this section by presenting the results from our experiments.

6.2. Off-line data preparation

The match summary for each match in the training describes events related to each ball of the play including the entities involved and the related actions. To help in the prediction of the next event, the history of events and transitions related to both batsman and bowler is extracted and represented as shown in Table 2.

The advertisement database consists of video clips downloaded from various sources as shown in Table 3.

Table 2. History of events

Bowler	Batsman	Present score	Next shot
Bowl: Shoaib	Sachin	0	0
Bowl: Shoaib	Sachin	0	0
Bowl: Shoaib	Sachin	0	1
Bowl: Shoaib	Sachin	1	4
•	•	•	•
•	•	•	•

Table 3. Advertisement database

Sponsor name	Metadata of the ad	Ad clip
Pepsi	Indian cricket team:lion:Harbhajan:six	a1.wmv
Pepsi	Sachin:Shane:Hooper:memory loss:plane:Honululu	almondjoy.wmv
•	•	•
•	•	•
Coke	Aamir:Shopkeeper:eightrupees:verbal:duel	britecereal.wmv

To represent and describe the semantics of the event content, the *major events* for both the players in a cricket game are considered. For Sachin, *major events* are events resulting in a boundary, quarter century, half century, century or out condition and for Shoaib *major events* are events resulting in a dot ball, ball resulting in less runs in crucial overs of match and wicket taken. The major event database consists of description of the *major events* and semantic tags associated with them based on their significance in the game. For example, a half century event can be a result of Sachin scoring a six at the score of 44, a six at the score of 45, a four at the score of 46, etc. For the scenarios in consideration, we have tagged these major events and advertisements with sample metadata suitable for the simulation. For a realistic implementation, these should reflect the actual semantics. The events before which the advertisements should be aired are specified by the sponsors. The format of the database is as shown in Table 4.

With the database created, training data needs to be mined for event prediction. The Weka data mining tool is used to determine the probabilities to predict the next score, the possible shot at each score for the batsman (Sachin) and the next shot and wicket taken at each ball in the over for the bowler (Shoaib). The probabilities of occurrence of major events are used to determine state transitions at each score of the batsman and each ball bowled by the bowler.

To find the probability of the next event being a *major event*, at each score, the probabilities of possible events that result in a *major event* or a state transition are of interest. At each score, since the event with the maximum probability might not result in a *major event* or a state transition, the output from the Weka data mining tool is filtered to compute probabilities for instances that might result in a *major event*. For example, at a score of 47, the maximum probability of next event can be for 1 run which does not result in a state transition or a *major event*. Hence, the probability of scoring a boundary or 3 runs at the score of 47 is determined and used, since scoring a half century results in a *major event*.

For another example, consider the events at scores 21 and 24. The maximum probability of the next event is for runs 1 and 0, respectively. Since 1 and 0 runs do not account for a *major event* or a state transition, we consider the probabilities of boundaries, i.e. fours and six. Now consider the probabilities of runs at the score of 21. For a state transition at the score of 21, a four or a six is required. However, the probability for a four or a six is zero at this score. So, no *major event* is predicted at the score 21. At the score of 25, the probability of both four and a six being non-zero, a four can be predicted since the probability of four is more than the probability of a six. A subset of the data depicting the probabilities of various events on a given score is depicted in Table 5.

6.3. On-line advertisement dissemination

The experimental evaluation of our approach starts when Sachin comes to bat. Shoaib is the bowler. The batting and bowler representation of statechart is instantiated for Sachin and Shoaib, respectively, with each state appropriately tagged the basis of on history and state significance. Further, event predictions and events leading to state transitions along with event significance are populated for each state of these instantiations. For each event prediction, *aggregated events*, if any, are also determined and populated for each of the states.

We use events from one match as test data for finding the accuracy of our event prediction approach. Events in the test data on each score of Sachin's play are used to identify the currently active state to which this event belongs, and further event predictions for current score and ball bowled checked. Events are considered for each run that Sachin scores in the match till he gets out. Events are also considered for each ball bowled by Shoiab when Sachin is batting.

Table 4. Major event database

Event name	⟨Present Score, Next Shot⟩	Event semantics	Relevant advertisements
Quarter century	⟨19,6⟩:⟨20,6⟩:⟨21,6⟩:⟨22,6⟩...	fishing:beach:listen:boost:strength:exercising	dinosaurs.wmv:levis.wmv:joebridge.wmv
Half century	⟨44,6⟩:⟨45,6⟩:⟨46,6⟩:⟨47,6⟩...	Arshad:leak:nepali:trekking:vinod:highinair	armstrong.wmv:bonkers.wmv:cosmick.wmv
•	•	•	•
Out near century	⟨76,out⟩:⟨77,out⟩:⟨78,out⟩...	youth:traditions:ancient:Dravid:enjoying: dream:marriage:memories:emotional	certs.wmv:glamor.wmv:hersyrup.wmv

Table 5. Probabilities of events at a given score

	Next event					
Current score	0	1	2	3	4	6
12	0.28	0.12	0.07	0.2	0.19	0.14
21	0.54	0.28	0.10	0.08	0	0
25	0.31	0.18	0.20	0.16	0.21	0.15
29	0.50	0.20	0.10	0.08	0.07	0.05
•	•	•	•	•	•	•
•	•	•	•	•	•	•
37	0.29	0.13	0.09	0.25	0.06	0.18

Having predicted the events at each score and ball bowled, relevant advertisements are determined and cached on the mobile device. If the predicted event occurs, the advertisement is displayed before notifying the event. To manage the cache on the mobile device effectively, apart from the *predicted next* event, *predicted future* events are considered to determine relevant advertisements. Further, prediction is also used to replace the data in the cache.

6.4. Results

Our implementation is based on two different approaches for predicting events viz. (i) *Context Un-aware Approach* (CUA), and (ii) *Context-Aware Approach* (CAA) as described in Section 4. We evaluate and compare results obtained from each one of these approaches for event prediction. If we consider history data, the probability of predicting *major events* at different scores is not the same, as we predict a *major event* only if the prediction probability is above threshold. The threshold value is considered as the minimum probability (non-zero value) of the occurrence of any event at a particular score. Table 6 describes the comparison of both the approaches used for event prediction.

For CAA, we use five different sets of training data whereas for CUA we collect results for five different runs, for better representation and for clarity. To measure the impact of training data on event prediction, the training data are divided into sets consisting of 5, 10, 15, 20 and 28 matches. The matches in each set are inclusive of the matches of the previous set. A 29th match is used to evaluate the whole approach. For each set and each run, we evaluate effectiveness of our techniques using the following metrics:

(i) *Prediction Accuracy*: Prediction accuracy = total number of successful predictions/total no of events in test data.
(ii) *Replacement measure* of advertisements in the cache of the mobile device as well as replacement measure considering *predicted future* events is computed as: *Replacement measure* = number of advertisement replacements in cache/number of predictions.
(iii) Instances in which the most relevant advertisement displayed to user is computed as: *Number of best advertisements delivered* = Number of instances for which the most relevant advertisement is displayed.
(iv) Instances in which the second most relevant advertisement displayed to the user is computed as: *Number of second best advertisements delivered* = Number of instances for which the second most relevant advertisement is displayed.

Considering multiple relevant advertisements for each of the major events and a fixed cache size, the cache management approach used is depicted in Fig. 5. For example, at a particular score and instance of time, cache management approach as described in Fig. 5 are applied in a scenario as depicted in Table 7.

To compute the metrics of the number of instances when the second best advertisement is delivered, we consider that for an event occurrence, no time is available to prefetch and cache the most relevant advertisement. In such a case, we deliver the second best advertisement available in cache. For the events of 4 (or 6) runs, we consider a relevant advertisement for 6 (or 4) runs, respectively, as the second best advertisement. Similarly, for events of 0 (or 1) runs and batsman out event, we consider any relevant advertisement for 1 (or 0) runs, respectively. For example, if a relevant advertisement for a 4-runs event is not available in the cache, then we consider an advertisement for 6 runs as second best advertisement.

Results for various metrics are as shown in Table 8. From the results we can observe the folllowing.

Table 6. Comparison of CUA and CAA Approach

Feature	CUA	CAA
Event prediction	Random prediction of major events	Probabilistic prediction of major events based on history data
'No Prediction'[a] condition	Random	Probabilistic prediction based on history data
Prediction probability for the major events and 'No Prediction' condition	Equal	Only if prediction probability is above a determined threshold

[a]For a particular score, no major event is predicted.

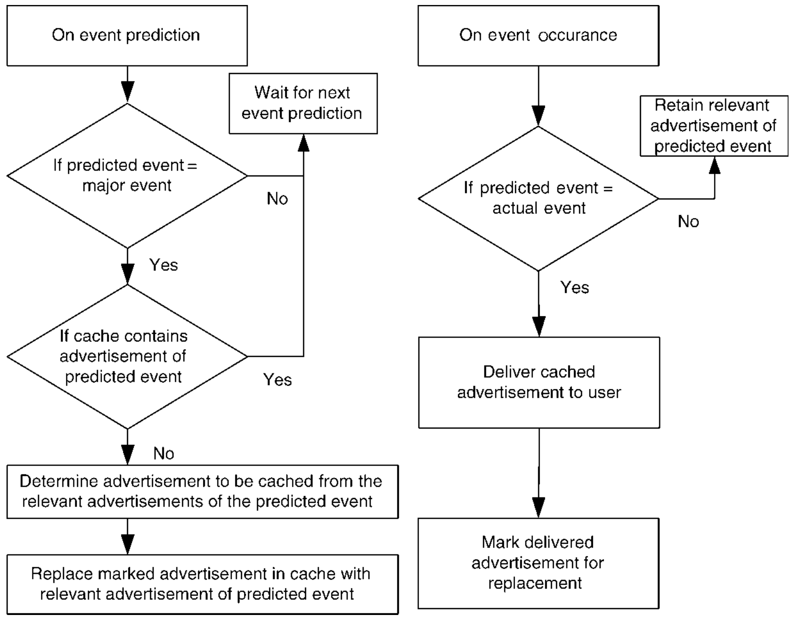

FIGURE 5. Cache management.

(i) Though prediction accuracy of CUA as well as CAA is in the same range (0.2–0.48 for CUA and 0.27–0.51 for CAA), the number of predictions for CUA are more, as CUA predicts all *major events* with equal probability whereas CAA predicts *major events* only if their prediction probability is more than the threshold. Example: the number of predictions for sets 1 and 2 is much less for CAA as compared to CUA.

Table 7. Scenario depicting application of cache management rules

Serial no	Attribute	Nominal value	Situation
1	Relevant advertisements for a four event	'a', 'b', 'c', 'd' and 'e'	Initial condition
	Relevant advertisements for a six event	'f', 'i', 'k', 'l' and 'm'	
	Cache size	5	
	Cache contains advertisements	'f', 'g', 'h', 'i' and 'j'	
2	Next prediction	Six runs	Prediction and event occurrence after the initial condition
	Actual event occurrence	Six runs	
	Prediction successful	True	
	Advertisement delivered to user	'f'	
	Advertisement marked for replacement	'f'	
3	Next prediction	Four runs	Prediction and event occurrence after situation 2
	Advertisement brought to cache	'a'	
	Advertisement replaced in cache	'f'	
	Actual event occurrence	Two runs	
	Prediction successful	False	
4	Next prediction	Four runs	Prediction and event occurrence after situation 3
	Actual event occurrence	Four runs	
	Prediction successful	True	
	Advertisement delivered to user	'a'	
	Advertisement marked for replacement	'a'	
5	Next prediction	Four runs	Prediction after situation 4
	Advertisement brought to cache	'b'	
	Advertisement replaced in cache	'a'	

Table 8. Results

	CUA						CAA					
Set no	Prediction accuracy (no of predictions)	No of ads delivered to cache	No of replacements	No of ads delivered to User	No of best ads delivered	No of second best ads delivered	Prediction accuracy (no of predictions)	No of ads delivered to cache	No of replacements	No of ads delivered to user	No of best ads delivered	No of second best ads delivered
With consideration of predicted future events												
1	0.44 (63)	16	11	12	11	1	0.27 (35)	11	6	8	8	0
2	0.37 (65)	14	9	10	10	0	0.48 (51)	13	8	14	14	0
3	0.20 (65)	10	5	6	6	0	0.51 (64)	19	14	15	15	0
4	0.48 (63)	16	11	13	12	1	0.48 (74)	14	9	14	14	0
5	0.34 (62)	14	9	10	8	2	0.44 (76)	13	8	13	13	0
With consideration of predicted future events												
1	0.44 (63)	13	8	11	10	1	0.27 (35)	10	5	8	8	0
2	0.37 (65)	11	6	9	9	0	0.48 (51)	10	5	14	14	0
3	0.20 (65)	8	3	6	6	0	0.51 (64)	14	9	15	15	0
4	0.48 (63)	14	9	12	12	0	0.48 (74)	8	3	14	14	0
5	0.34 (62)	11	6	10	10	0	0.44 (76)	11	6	13	13	0

(ii) The number of advertisements delivered to the cache reduces while considering *predicted future* events for both CUA as well as CAA. Example: for set 1, the number of advertisements delivered to the cache reduces from 16 to 13 for CUA whereas it reduces from 11 to 10 for CAA.

(iii) In the case of CAA, with or without considering *predicted future* events, there is consistency in the number of best advertisements delivered to the user. Example: the number of best advertisements delivered reduces from 11 to 10 for set 1 and increases from 8 to 10 for set 5 when considering *predicted future* events for CUA, whereas for CAA, for all sets, the number of best advertisements delivered remains the same.

(iv) Only in the case of CAA does the number of advertisements delivered to user be more than the number of advertisements delivered to cache. Example: for set 2, the number of advertisements delivered to cache is 13 whereas the number of advertisements delivered to user is 14.

(v) In the case of CAA and especially while considering *predicted future* events, the number of advertisements delivered to the user are substantially higher than the number of advertisements delivered to the cache. Example: the number of advertisements delivered to user for sets 2, 4 and 5 are much more than the number of advertisements delivered to cache while considering *predicted future* events.

(vi) Instances in which the second best advertisement is delivered is observed in CUA and not in CAA. Example: for sets 1, 4 and 5, the second best advertisements are delivered for CUA whereas in CAA, always the best advertisement is delivered.

From the observations, the following can be noted.

(i) Although the prediction accuracy for delivering relevant advertisements are identical, CUA requires more predictions than CAA [prediction accuracy of 0.20 (third row in table) for CUA as compared to prediction accuracy of 0.27 (first row in table) for CAA]. This results in not only an increase in computation to determine the relevant advertisement for the predicted event but also in an increase in bandwidth usage to disseminate and cache advertisements frequent cache updates are also required.

(ii) Reducing frequent cache updates by considering *predicted future* events has an impact only on CAA (for CAA, the number of advertisements delivered to user are more than the number of advertisements delivered to cache). Hence, consideration of *predicted future* events results in effective utilization of available bandwidth and less computation overhead to achieve delivery of relevant advertisement in CAA.

(iii) CAA results in delivery of the best advertisements (there is no instance of delivery of a second best advertisement). This results in achieving the maximum impact on the user by delivering the most relevant advertisement at the most appropriate time.

(iv) While considering *predicted future* events, CAA results in delivery of the best advertisements for most of the events with a minimum number of cache updates (the number of best advertisements delivered to the user as compared to the number of advertisements cached). The result is less computation overhead to determine relevant advertisements as well as effective bandwidth utilization for delivery and caching of relevant advertisements.

From the results we conclude that our approach has a superior performance resulting in: (i) better prediction accuracy; (ii) delivery of the most relevant information most of the time; and (iii) effective cache management in mobile devices.

7. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a real-time personalized information delivery system. The system disseminates and displays information relevant to events before event notification to mobile users. To achieve real-time information delivery, the system is based on: (i) the representation of domains with event-based scenarios using statecharts; (ii) the use of a novel combination of history information and state transitions to predict events; and (iii) the delivery and display of information in real time with content relevant to the event by pre-fetching and caching information based on the predicted events. We compare our context aware approach with a Context Un-aware Approach of predicting events. The results show that our approach requires less number of predictions for the same prediction accuracy. Also, consideration of *predicted future* events results in increased relevance of information delivered as well as lower information replacement in the cache. We also notice that consideration of *predicted future* events has a positive impact on our approach.

We would like to extend our approach to validate it in other domains having event-driven scenarios. Of the contextual attributes of entity, activity, time and location, our current approach is based on entity and activity contextual parameters. We would also like to extend our approach on spatial and temporal dimensions of context by considering contextual attributes such as location and time.

REFERENCES

[1] Tarkoma, S. (2004) Event dissemination service for pervasive computing. *Proc. Pervasive 2004 Doctoral Colloquium*, Linz/Vienna, Austria, April 18–23, pp. 155–160. Published in Advances in Pervasive Computing, Austrian Computer Society Press.

[2] Podnar, I., Hauswirth, M. and Jazayeri, M. (2002) Mobile push: delivering content to mobile users. *Proc. ICDCS Workshops 2002*, Vienna, Austria, July 2–5, pp. 563–568. IEEE Computer Society Press, Washington, USA.

[3] Ranganathan, A. and Campbell, R.H. (2002) Advertising in a pervasive computing environment. *Proc. 2nd ACM Int. Workshop on Mobile Commerce*, Atlanta, Georgia, USA, September 2002, pp. 10–14. ACM Press, New York, USA.

[4] Ho, S.Y. and Kwok, S.H. (2003) The attraction of personalized service for users in mobile commerce: an empirical study. *ACM SIGecom Exch.*, **3**, 10–18.

[5] De Castro, J.E. and Shimakawa, H. (2006) Mobile advertisement system utilizing user's contextual information. *Proc. MDM 2006*, Nara, Japan, May 9–13, pp. 91. IEEE Computer Society Press, Washington, DC, USA.

[6] Boll, S., Krösche, J. and Scherp, A. (2004) Personalized mobile multimedia meets location-based services. *Proc. Multimedia Inf. Syst. Workshop at 34th Annual Convention of the German Informatics Society (GI)*, Ulm, Germany, September 2004, pp. 64–69. Lecture Notes in Informatics 51 GI 2004.

[7] Gibbon, D., Begeja, L., Liu, Z., Renger, B. and Shahraray, B. (2004) Multimedia processing for enhanced information delivery on mobile devices. *Proc. Emerging Applications for Wireless and Mobile Access, MobEA II*, New York, USA, May 18.

[8] Aalto, L., Göthlin, N., Korhonen, J. and Ojala, T. (2004) Bluetooth and WAP push-based location-aware mobile advertising system. *Proc. 2nd Int. Conf. Mobile Systems, Applications and Services*, Boston, USA, June 6–9, pp. 49–58. ACM Press, New York, USA.

[9] Hristova, N. and O'Hare, G.M.P. (2004) Ad-me: wireless advertising adapted to the user location, device and emotions. *Proc. Softw. Track of the 37th Hawaii Int. Conf. System Sciences (HICSS-37), DC*, January 5–8. IEEE Computer Society Press, Washington, Island of Hawaii, USA.

[10] Chen, G. and Kotz, D. (2000) TR2000-381. *A survey of context-aware mobile computing research*. Dartmouth College, Hanover, NH .

[11] Mohamed Yunos, H., Zeyu Gao, J. and Shim, S. (2003) Wireless advertising's challenges and opportunities. *Computer*, **36**, 30–37.

[12] Drossos, D. and Giaglis, G. (2006) Mobile advertising effectiveness: an exploratory study. *Proc. 5th Int. Conf. Mobile Business*, Copenhagen, Denmark, June 26–27, p. 2. IEEE Computer Society Press, Washington, DC, USA.

[13] Glinz, M. (1995) An integrated formal model of scenarios based on statecharts. *Proc. 5th European Softw. Eng. Conf.*, Sitges, Spain, September 25–28, pp. 254–271. Springer-Verlag Press, London, UK.

[14] Witten, I.H. and Frank, E. (2001) *Data Mining: Practical Machine Learning Tools with Java Implementations*. Morgan Kaufmann Publishers, San Francisco, USA.

[15] Lucas, P.J. (1994) UIUCDCS-R-94-1868, *An object-oriented language system for implementing concurrent, hierarchical, finite state machines*. Department of Computer Science, University of Illinois, Urbana-Champaign.