A Framework for Generic Internet-Based Information Notification

Nader Mohamed, Jameela Al-Jaroodi, and Imad Jawhar
The College of Information Technology
United Arab Emirates University
Al Ain, P.O. Box 17551, UAE, {nader.m,j.aljaroodi,ijawhar}@uaeu.ac.ae

Abstract

This paper introduces a framework to monitor information available dynamically on the Internet through web services, RSS feeds or dynamic HTML documents based on user defined criteria. The paper also discusses some of the open issues associated with the design and application of the framework. This framework is designed to deal with public information available on the Internet on a single or multiple websites and from a single or multiple web services as the information source of which the user can build his/her notification criteria. The Internet provides a huge amount of dynamic information such as news, stock information, current and expected weather status, currency exchange rates, etc. In the framework, all this public information is viewed as a set of Internet variables. These variables can be used to define the user monitoring and notification criteria. Based on the defined criteria, SMS and/or email notification messages can be generated and sent to the user notifying him/her whenever the criteria are met. A number of applications can benefit from this framework to provide real-time useful information to the user.

1. Introduction

Recently, the Internet has become one of the main sources of live up-to-date information and there are three main approaches to get this information. The first approach, which started with the concept of the Internet, is browsing. A user can move from one page to another to obtain the required information. The second approach is searching, where a user provides keywords to be searched for in the available documents. The third approach is notification, where a user defines some notification criteria for specific events on some websites. The user is notified whenever his/her criteria are met. The notification can be delivered by SMS or email messages. The user can state for example, on the CNN website, that he wants to be notified whenever there is news about Greece or news about new technology news. User notifications are very useful; however, current available notification systems are very restricted. The user is restricted to specific types of information to use and limited types of notifications. Users have no means to define advanced criteria he/she may be interested in. In addition, the data used for notification is usually limited to what is available on the website of the organization managing the notification system.

For example, an investor can set a notification condition on Yahoo Finance such that whenever the Wal-Mart stock price drops to a specific value in US dollars a notification is sent. However, the investor can not define advanced notification criteria such as when the trading volume reaches a specific quantity and when the Wal-Mart stock price reaches to a specific price in Euros. Although, both the trading volume and the USD to Euro exchange rates are available over the Internet, Yahoo Finance does not support that type of notification. Therefore, an investor can not define any advanced or compound criteria.

This paper introduces a framework for a generic Internet information notification system. This framework uses public information available on the Internet on one or more websites or from one or more web services as the information source of which the user can build his/her notification criteria. The Internet provides a huge amount of dynamic information such as news, stock information, current and expected weather status, currency exchange rates, etc. All this public information is viewed as a set of Internet variables, which can be used to define the user notification criteria. The main aim of the paper is to introduce the main concepts of the framework, to
provide a basic feasible design, and to discusses the associated design and research issues. As a result, we will not provide a detailed design.

In this paper, Section 2 discusses related work. Section 3 discusses how the Internet variables are defined and Section 4 describes the framework for the generic Internet information notification. Section 5 discusses some potential applications of the proposed framework and open issues are discussed in Section 6. Section 7 concludes the paper.

2. Related Work

A lot of literature is available on notification systems representing different projects, protocols, and architectures. Huang et al. [1] provide a comparative study of web services (WS) based event notification systems. The authors focus on the evolution of major specifications of notification systems. Several organizations proposed various event notification systems such as CORBA notification service specification [2], the notification specification in Open Grid Services Infrastructure (OGSI) [3], Web Services Eventing (WS-Eventing) [4], and Web Services Notification [5][6][7][8]. Carzaniga et al. [9] present SIENA (Scalable Internet Event Notification Architecture), which is an Internet-scale event notification middleware service for distributed event-based applications deployed over wide-area networks. Additionally, the authors in [10] discuss and classify the various models of the publish/subscribe communication paradigm. It presents the common denominator underlying these models with respect to time, space and synchronization. It also identifies the benefits and shortcomings of the various systems in terms of interfaces and implementations. In these notification systems, both the publishers and subscribers must follow specific communication protocols to enable the notification process. In some projects, the intermediate network among publishers and subscribers also must provide some support to enable the notification process. In all web-based notification systems, the subscribers are restricted with the information available within the publishers’ domain. In addition, subscribers are restricted with the rules of the publishers regarding available events and how to subscribe for events.

The main difference between our framework and other web-based systems is that our system does not have specific publishers of information or events; instead we expand the source of information to every public-domain website. Therefore, all public internet sites can be considered as publishers for our notification system and all the information available on these sites can be used to build the notification rules. In other words, the users will be able to define the notification rules based on information available and updated on any website on the Internet or based on information provided by any web service.

3. Internet Information Sources

The generic notification system for internet information depends on a set of variables defined by the users. These variables represent dynamic information available on the Internet. A user can for example define a variable for Wal-Mart current stock price and another variable for US Dollar to Euro exchange rates. The Internet variable content's sources are from dynamic HTML documents [11], XML documents [12], web services, or RSS feeds [13] available on the Internet. This information may be for stock market information, currency exchange rates, temperatures, oil prices, gold prices, interest rates, etc. The proposed framework depends on the Internet for obtaining the required and updated information to build the notifications. This Section discusses how to get the required information available over the Internet.

If the information needed by a notification application is available in web services, then user can easily define an Internet variable and link it with the corresponding web service which provides the required information. Web services can provide a structured and simplified way to obtain services or specific information from the Internet. Web services provide web APIs that can be accessed over a network, such as the Internet, and executed on a remote system hosting the requested services. The main problem with web services is that not all types of information available over the Internet are provided by web services. Most of the useful information on the Internet is still only available in semi-structured HTML documents.

Unlike XML documents, HTML documents do not have any semantics for their data. HTML documents usually contain tags, scripts, links, and user defined data. Obtaining specific data from a dynamic HTML document for reuse in other applications can be a complex task. It is very difficult to identify the required data components and dynamically use them in other applications. However, we have recently developed a simple and efficient approach for retrieving live HTML-based Internet information [14]. This approach can be used to define the Internet variables that will have their data updated from the Internet. The main idea of this
approach is based on finding fixed titles or headers that appear in browsers for HTML documents directly or semi-directly before the needed dynamic information. These fixed titles or headers are used as references to know the position of the required dynamic information. The proposed approach is developed as a Java class. Multiple objects can be created from this class for different Internet HTML documents that contain some of the required information. A number of techniques were developed to find this information in any HTML document. These techniques are implemented in a set of Java methods.

The first technique is to find information directly after a specific text header. For example, in Figure 1, the Wal-Mart Stores, Inc. Stock information is displayed from the Yahoo Finance site. The user can find the last trade price, the previous closing price, and the trade volume from this page. All this information can be defined as notification variables in the proposed system. To get the last trade price, the user needs to call the `get("Last Trade:")` method. The `get` method will search for the title provided, "Last Trade:", and return the next data field after this title field, "48.10."

The second technique is to find certain information after the appearance of a specific header for the $n$th time. For example, Figure 2 is web information for RAK Properties with symbol RAKPROP that is listed in Abu Dhabi Securities Market. We can see that, the bid volume amount appears after the "RAKPROP" header while that header appeared twice in the page. To get the bid volume, this technique will search for two "RAKPROP" text headers before returning the bid volume amount. This technique is implemented in another method with interface `get(n, header)`.

The third technique is to find information semi-directly after a specific fixed title or header. This finds the $i$th information field after the appearance of a specific header $n$ times. For example in the RAKPROP example, the user wants to get the current price of the stock. This price is listed 7 fields after the second appearance of "RAKPROP" header. The interface for this method is `get(n, header, i)`. Users can use any HTML document on the Internet to define their variables. More information about the methods, optimization techniques, implementation, and performance of this approach can be found in [14].

**4. The Notification Framework**

The system can be developed as a client for a single user or as a web server for multiple users. Since the main aim of this paper is to introduce the main concept of generic notification system for Internet information, only a basic Internet monitor is discussed. The notification system monitors one or more Dynamic HTML documents from one or several web servers on the Internet as shown in Figure 3. In addition, the system can deal with one or more web services to get some other information. The user can define the needed Internet variables as discussed in the previous section. These variables are updated by the system periodically by downloading the HTML documents defined by the user and extracting the required fields or calling the corresponding web services.

The user can add some calculated variables from the defined Internet variables. These calculated variables are usually functions of the defined Internet variables. The system periodically checks the set of notification criteria defined by the user. Whenever one of the criteria is met, a predefined notification message that
contains some of the defined Internet variables or the defined calculated variables will be sent to the user. The user needs to define first all Internet variables and all calculated variables he/she needs for preparing the notification criteria. With each notification criteria, a notification message should be defined. The message consists of some static text defined by the user and the relevant variables (extracted and/or calculated). Furthermore, the user needs to define the emails or phone numbers where the notification message should be sent.

To explain the main operation of the system, we consider a simple case needed by a European investor (using the Euro as trading currency) who would like to sell EMAAR stock listed in Dubai Stock Market (DFM) in Dirham if the price increases above 2 Euros. Two types of information are needed, the stock price in DFM offered by the DFM website and the Dirham-Euro exchange rate offered on a different currency exchange site such as the Yahoo Currency Exchange site. Both values are dynamic and finding the correct price normally requires the investor to open both sites and continuously perform currency conversions on the listed prices until the desired value is reached. However, using our framework the investor will need to specify two variables for the current stock price in Dirham, *EmaarAED*, and the current exchange rate between Dirham and Euro, *AEDEUR*. He/she can use these two Internet variables to generate a calculated variable, *EmaarEUR*, for the current stock price in Euro.

The value of *EmaarEUR* variable is calculated based on an equation, *EmaarAED x AEDEUR*, defined by the user. In addition, the user needs to specify a condition at which a notification message will be sent as soon as the price reaches the desired value in Euros.

The generic internet notification system can be implemented in different ways. However, here in this paper, we will explain a basic implementation to clarify the main concepts of the framework. Basic pseudo code for the notification system is shown in Figure 4. There are three main sections executed in an infinite loop. The sections are the Internet variables section, the calculated variables section, and the notification condition section. In the Internet variable section, the user can use a web service interface or the HTML interface mentioned in Section 3 to get Internet information. In this section for example, we can define the two Internet variables mentioned in the European investor's case, *EmaarAED* and *AEDEUR*. In the calculated variables section, all new variables that are functions of the defined Internet variables needed by the notification conditions are calculated. One example is calculating the *EmaarEUR* variable. In the conditions section, all notification conditions, notification messages, and notification types (SMS/email) and destination are defined.

```plaintext
// An Internet Notification Program
While( infinity )
begin
  // The Internet variables section
  // all Internet variables are defined.
  // Their values either from live HTML
  // documents or from web services
  Get EmaarAED from DFM website
  Get AEDEUR from Yahoo website

  // The calculated variables section
  // the calculated variables are functions of
  // internet variables
  EmaarEUR = EmaarAED x AEDEUR

  // The notification conditions section
  // defined conditions are checked and
  // messages are sent
  If( EmaarEUR > 2.0 )
    SendSMS(message, phone)
  // Wait some time.
EndWhile
```

Figure 4. Basic Internet Notification Program.
5. Applications

Using the proposed framework, new real-time notification applications can be easily developed. These notification applications can be based on some information from the Internet and the users’ needs. Examples of these applications are:

I. Monitoring weather changes by farmers: Many farmers rely on changes in weather conditions to decide on certain actions such as when to plow the land, when to put to the seeds, how many times to water the plants and when to harvest the crops. In addition, some crops like grapes and delicate fruits could face dangers due to sudden frost conditions. Farmers need to know weather forecasts for the coming few days to be able to make better decisions about their work. For example, the farmer could specify that if temperatures will reach a specific value within the next 6 days in the farm area and will reach another specific value in the surrounding towns then a notification should be sent. This will allow the farmer to decide on the next course of action. In some cases knowing of possible storm conditions could be helpful to protect the crops or harvest them if necessary. In addition farmers could also choose to be notified if rain forecasts reach a certain level. Knowing about rain forecasts help farmers decide on watering plans and when to water the plants and how much they will need in light of the notification information received. Weather expectations are available on the Internet. The user can set his notification conditions using this public information. The proposed system will monitor weather expectations and generate notification messages whenever the defined criteria are met.

II. Notification System for Stock Information Changes in Stock Markets: Some stock markets provide online information about stock prices; however, they do not provide a notification mechanism to allow users to get information about stocks when special criteria are met. The proposed system can be used to define notification variables and certain price change criteria. The system will notify the users by emails or by any other messaging mechanism whenever those criteria are met. Examples of the criteria may be as simple as notifications of prices reaching a specific value or could be complex to cover multiple variables and markets. For example requesting a notification when a stock reaches a certain price level and the selling volume increases by a given percentage or notification when a stock price increases/decreases by a certain percentage while the sell orders are above/below a specified level. The proposed system will monitor stock prices over the Internet and generate notification messages whenever these criteria are met.

6. Open Issues

The framework mentioned in this paper provides a generic Internet notification system which can be very useful for some applications. The proposed framework provides a mechanism to reuse information available on the Internet to implement notification applications. Although in this paper the main concepts of the generic Internet information notification is discussed and a basic implementation is demonstrated, a number of design and research issues are also raised. Some of these issues are currently under our investigation.

- Most HTML documents have a relatively fixed structure and are mostly updated within the original format. However, a problem may occur in some cases when the component formats of the HTML document are changed. By this we mean adding extra fields with the same labels we used. An automatic validation mechanism is needed to allow the system to make sure that the formats of the defined HTML documents were not changed before attempting to extract the information for the Internet variables. One possible solution for this problem is to automatically capture and store the format patterns of the HTML documents. These patterns can be used by the system to discover any future changes in the downloaded documents. In case there are some changes, the system notifies the system administrator to configure new parameters for the Internet variables.

- The generic notification system can be implemented as a single user system as discussed in this paper or as a multi-user web server. For the multi-user server, the main issue is how to optimize the server operations by avoiding any repetitive operations due to similar or semi-similar user-defined notifications.

- Some Internet information can be very dynamic. Example is the current price of a stock, where the values change very frequently (e.g. every second). The issue here is how a generic notification system can efficiently capture all value changes.

- Notification conditions can be temporal. An example is a notification message should be sent if a temperature drops 10 degrees within one hour, which requires keeping track of changes over an extended period of time. The issue here is how a generic notification system can efficiently support that type of notifications.
• Regular users should be able to use the notification system. The issue here is in how to provide a basic graphical interface to allow regular users to easily use the system.

7. Conclusion

This paper introduced a generic notification framework for Internet Information. This framework is based on information available from web services or dynamic HTML documents available on the Internet. The framework allows a user to define a set of Internet variables, calculated variables, notification conditions, notification messages, emails, and mobile phone numbers. Users are given the freedom to define and set their personal criteria of notification without being limited to one notification provider or to a single type of simple notifications. Several applications can be designed using this framework to provide sophisticated notification systems for users in various application domains. The methods designed in the framework allow the retrieval of data values from various locations in the HTML document or from web services. Examples of applications that can use this framework are the stock market, weather information, currency exchange information, and merchandise prices. The paper also discussed some design and research issues associate with that framework that need to be addressed. In our current work, we are investigating these issues and working on providing some practical solutions.

Acknowledgement

This work is supported by UAEU Research Grant # 01-03-9-11/08.

References