Generating Web Monitors

Nader Mohamed, Ahlam Romaithi, Eiman Samahi, Maitha D. Kendi, and Ebtisam Jabrie
The College of Information Technology
United Arab Emirates University
Al Ain, P.O. Box 17551, UAE, nader.m@uaeu.ac.ae

Abstract - A web monitor is a program used to monitor some public HTML-based Internet information available from dynamic HTML documents on the Internet based on a user defined set of criteria. It is able to deal with public information available on the Internet on a single or multiple websites as a set of variables which the user uses to define some required notification conditions. The monitored information may be stock information, current and expected weather status, currency exchange rates, etc. Based on the defined criteria the monitor generates SMS or email messages and sends them to the user notifying him/her whenever the criteria are met. In this paper, we describe a development environment where web monitors can be automatically developed and generated. This development environment facilitates creating web monitors.

Keywords: Notification Service, Code Generation, Internet Information, HTML

1 Introduction

A notification system is a combination of network services and an infrastructure that provides a mechanism of delivering notification messages to a single or multiple recipients based on the occurrence of an event specified by the recipients. Notification systems usually consist of two types of parties, the publisher/producer and the subscriber/consumer. There are different notification systems used for different domains and applications such as telemedicine applications [2], accrual failure detectors [3], e-commerce applications [4], emergency services [5], and finance/banking.

Over the Internet, some websites provide notification services that allow users to define some notification criteria for specific events. The user is notified whenever his/her criteria are met. The notification can be delivered by SMS, email, or online messages such as Messenger. The user can state for example, on the CNN website, that he/she wants to be notified whenever there is news about UAE. In another example, a user can set a notification condition on Yahoo Finance that causes him/her to be notified whenever the Wal-Mart stock price drops to a specific value or whenever the Wal-Mart price increases by a certain percentage.

Internet notification services are very useful; however, current available notification services 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 they 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 service. For example, a user can set a notification condition on Yahoo Finance such that whenever the Wal-Mart stock price drops to a specific price 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 a specific price. Although, the trading volume is available in real-time over the Internet, Yahoo Finance does not support that type of notification. Therefore, an investor can not define any advanced or compound criteria.

Recently a framework was developed for a generic notification service based on public information available over the Internet [8]. In this framework, users can use any public information available over the Internet to define the notification criteria. The core component of this framework is a web monitor that may be implemented based on user defined criteria for notifications. This monitor sends notification messages to the user whenever the criteria are met. In this paper, we discuss a development environment that can be used to generate different web monitors, which are basically Java programs. This development environment helps users to define their notification criteria based on the Internet information. Then based on the user’s definitions, a web monitor program is automatically generated.

In the rest of this paper, Section 2 provides background information about the developed framework for a generic Internet-based notification. Section 3 discusses the development environment for the web monitors and Section 4 describes how web monitors are automatically generated. Section 5 discusses some potential approaches where the web monitor development environment can be enhanced. Section 6 concludes the paper.
2 Generic Notification Service

Different live information is published on the Internet such as currency exchange rates, interest rates, stock prices, expected weather status, and other online information. This information is updated frequently. Recently, a new framework was developed to reuse these types of information to enable building flexible notification services that depend on the public information available in dynamic HTML documents available over the Internet [8]. This framework depends on defining a set of variables called Internet variables. The values of these Internet variables represent some dynamic values available on HTML pages over the Internet. For example, a user can define one Internet variable for the current Wal-Mart stock price which is displayed on one of the web pages. The user can also define another Internet variable for the current Dollar-Euro exchange rate which is also displayed on another HTML page on a different site. The user can define multiple Internet variables each of which reflects a piece of the dynamic information available over the Internet. These variables together can then be used to define the notification criteria.

A program called a web monitor is responsible for obtaining the most up-to-date values for these variables from the Internet. Obtaining Internet information can be achieved using an approach developed recently for retrieving live HTML-based Internet information [1]. 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 reference points to know the position of the required dynamic information. The proposed approach is developed as a Java class. This class is called `urlINFO`. 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, Figure 1 contains online stock information for stocks listed in Dubai Financial Market. This page is updated dynamically in real-time. The user can use this HTML document to find the value of 52 week price High for Scandinavian Insurance. This information can be defined as an Internet variable. To get the 52-week high, the user needs to call the `get("A. Scandinavian Insurance")` method. The get method will search for the title provided, "A. Scandinavian Insurance," and return the next data field after this title field, "4.25."

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)`. 

Figure 1. Dubai Financial Market Online Stock Information.

Figure 2. RAKPROP Quote from Abu Dhabi Securities Market.
A web monitor can monitor one or more Dynamic HTML documents from one or several web servers on the Internet as shown in Figure 3. The user defines the needed Internet variables, which are updated by the system periodically by downloading the HTML documents defined by the user and extracting the required fields. The user can also 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 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 values of the variables (extracted and/or calculated). Furthermore, the user needs to define the emails or phone numbers where the notification message should be sent.

The process of defining the required Internet variables is one of the most critical steps in defining the notifications. Mistakes in defining the Internet variables will cause the notifications to function incorrectly. WMDE allows users to test their defined Internet variables before completing the configuration of the rest of the notification parameters. The user can select the option "Test," to get the development environment to display the current values of the defined Internet variables. The user can compare these values to the actual wanted information by browsing the defined URLs and match the displayed information with the values printed by selecting the "Test" option.

WMDE also allows the user to define new calculated variables using the defined Internet variables or other defined calculated variables. With each calculated variable the user needs to define the variable name and a mathematical equation which will be used to calculate the value of the variable. Calculated variables are useful to define new parameters for notification conditions. For example, the Wal-Mart Stores stock is listed in USA stock markets in dollar. If an investor needs to build a notification condition for this stock in Euro, he/she needs to define a calculated variable to convert the current price in dollar to Euro. This process can be done by defining two Internet variables. The first one is to obtain the current price of Wal-Mart Stores in dollar and the second one is to obtain the current exchange rate for dollar-to-Euro. These variables can be obtained form different HTML pages on the Internet. Then, the user needs to define a calculated variable that multiplies both defined Internet variables to get the current Wal-Mart price in Euro.

WMDE allows users to define their own notification messages. These messages can contain some constant text as well as some of the defined Internet variables or/and the defined calculated variables that are embedded in the constant text. For example, the user can define a message to inform him/her about an increase in specific price. In this message, there is a constant text as well as a figure which represents the increase in the price. For each defined message, there is a unique message ID of associated with it. This message ID can be used later to define notification conditions.

A number of notification conditions can be defined by users using WMDE based on their defined Internet variables or/and calculated variables. For example, the user can use that option to define when the current Wal-Mart stock price reaches a certain level, a message should be sent to him/her. Both basic and complex conditions can be defined. The syntax

![Figure 3. The Internet Information Notification System Monitors the Internet and Sending Notification Messages for End Users.](image-url)
used for defining conditions is the same as that in Java programs. The user can use relations operators (<, <=, >, >=, !=, ==) as well as combination operators (||, &&, !) to combine two basic conditions in a complex condition. Parenthesis may be used to combine multiple conditions and set operators priority (like in math. equations). The user also needs to define with each condition, a message ID for the message that will be sent as a notification message if the defined condition is fulfilled. In addition, the user defines a mobile phone number and/or an email address for the notification message to be sent. The user can also optionally define a subject for the email message.

WMDE provides the users with a set of fields which the user fills to define all the above mentioned options. The development environment was implemented using Java. The information defined by users is stored on an Oracle database. WMDE allows users to modify the previously defined notification applications. After defining all options, the user can generate from this environment a web monitor based on his/her specifications. This web monitor is a Java program which can be compiled and executed to monitor the Internet information based on the user’s specifications. In the next section, we will discuss in more details the process of generating web monitors.

4 Generating Web Monitors

The user uses the “Generate” function in WMDE to automatically generate web monitors. In this generating process, the configuration defined by the user (as described above) is used to generate the web monitor. The monitor needs to be executed on a computer that is connected to the Internet. This is to retrieve the information required by the user and to send the notification emails. In addition, in case SMS messages will be used in the system, a mobile phone or GSM modem is required to be connected to that computer to allow the web monitors to send SMS notification messages. The process of generating a web monitor and operating it to monitor some required Internet information is illustrated in Figure 4. When the monitor is executed, it will periodically retrieve the current values of the defined Internet variables, calculate the values of the defined calculated variables, check the defined conditions, and send notification messages if the corresponding conditions are fulfilled.

In order to simplify the process of generating Java based web monitors, a Java class was developed to generate a web monitor. This class has a number of methods that can be used to generate the code of the web monitor Java program. Executing each of these methods will generate a part of the code. These methods are the Initiate, URL, Internet_Variable, Calculate_Variable, Condition, Email, SMS and End methods.

The Initiate method generates the beginning of the web monitor java program and creates two objects, one for an SMS object and the other is for an email object. These objects will be used to send SMS and email notifications messages. The URL method is used to generate the code to create an object for an HTML document that is defined by the user. This method will use the urlINFO class explained in Section 2 to create an object for the defined HTML page and to use this object to download an update information. This object will also be used to retrieve the required information. The Variable method is to generate the code for defining Internet variables and obtaining the values of these variables using the defined urlINFO object. This method is also used to generate code to convert retrieved String-type numbers to double type format. The Calculate_Variable method is used to generate the code for each defined calculated variable. The Condition method is used to generate the code for the defined conditions. The Email method is used to generate the code to send an email message to the address that was defined by the user in WMDE. In this method JavaMail APIs [6] are used for sending email messages. The SMS method is used to generate the code to send SMS notification messages. This method uses SMSLib [7] for sending SMS messages. Finally, The End method is to generate the code to suspend the generated monitor for a few seconds and other code to end the generated monitor Java program.

Figure 4. Generating and Executing a Web Monitor.

When the user selects the “Generate” option, a process starts to create a new file for the web monitor java program. Based on what the user defined, this generate process will call the appropriate methods to create the web monitor Java program. An example of a generated web monitor is shown in Figure 5. In this example, the user defined two Internet variables: the first is for the current price of EMAAR stock listed in Dubai Financial Market and the second is for today’s
low price, the minimum price that *EMAAR* was sold today. The user is interested in receiving notification messages if the difference between the current price the lowest price is one Dirham.

```java
import java.io.*;
pubic class test
{
pubic static void main(String[] args)
throws Exception
{
SendMailDemo s = new SendMailDemo();
SendMessage ms = new SendMessage();

urlINFO stock = new urlINFO
("http://www.dfm.ae/marketwatch/default.aspx");

while(true)
{
stock.download();
String _Low Price = stock.get(1, "EMAAR Properties", 5);
double LowPrice = Double.parseDouble(_LowPrice);
String _Current Price = stock.getWI(4);
double CurrentPrice = Double.parseDouble(_CurrentPrice);

double DIFF = CurrentPrice - LowPrice;
if( DIFF >= 1.0 ){
s.mail( "THE DIFF = +Diff, "About EMAAR", "2003999@uaeu.ac.ae");
ms.sms( "THE DIFF = +Diff, "97150441111" );
}
Thread.sleep(30000);
}
}
```

Figure 5. A Generated Web Monitor Java Program.

5 Discussion

WMDE saves a lot of time and effort in creating web monitors that are created based on user needs. The current generation process in WMDE generates basic implementation of web monitors. However, more code can be added to the generated web monitor program to solve some issues raised with having a generic Internet-based information notification service [8]. One example of these issues is that 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 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 make the “Generate” process in WMDE to automatically capture and store the format patterns of the HTML documents. These patterns can be used by the monitor to discover any future changes in the downloaded documents. In case there are some changes, the generated monitor notifies the system administrator by email to configure new parameters for the Internet variables. The validation process can be added to the generated web monitor to automatically discover if there are changes in the defined HTML pages.

Some Internet information can be very dynamic. For example, the current price of a stock, where the values change very frequently (e.g. every second). The issue here is how to develop web monitors that efficiently capture all value changes. One possible approach is to use multithreads to download and process all changes. It is very difficult for a regular user to develop a multithreaded web monitor. However, it is possible to enhance the “Generate” process to automatically generate multithreaded web monitors that capture all the values of the required Internet variable in an efficient manner.

Notification conditions can also be temporal. An example is when a notification message is needed to be sent if a stock price drops 10 dollar within 3 days, which requires keeping track of changes over an extended period of time. It is very difficult for users to develop a web monitor that deals with this type of notification. It is better to have an environment such as WMDE to support developing this type of conditions.

6 Conclusions

This paper introduced a development environment, WMDE, that generates web monitors. The generated web monitors monitor and obtain information available on dynamic HTML documents on the Internet. This environment allows a user to define a set of Internet variables, calculated variables, notification conditions, notification messages, emails, and mobile phone numbers. The development environment generates web monitor java programs based on user definitions. These generated programs are executed to monitor the required information over the Internet. Whenever the required condition is satisfied notification messages are sent through email and/or SMS messages.

Several notification applications can be designed using this environment to provide sophisticated notification services for users in various application domains. Examples of these domains are applications related to stock markets, weather information, and currency exchange information.

Acknowledgments

This work was supported in part by UAEU Research grant # 01-03-9-11/08.
References


