Web Application for Intelligent Management of Entities of Remote Web Servers

Petr Pospěch
Department of Measurement and Control,
Faculty of Electrical Engineering and Computer Science
VSB Technical University of Ostrava,
Ostrava-Poruba, Czech Republic
p.pospech@seznam.cz

Abstract— On some web sites (shops, auctions, data storage) are constantly created new items, and in some cases, it is necessary to watch these items. This project deals a solution to this problem via a custom application to track and monitor items on the web servers. For each query to search, the web server responds with a few words that match the specified search term. All items matching expressions are saved and sent to the user selected communication channel common by email. This problem is solved in the created application for Windows that scans on the server and information on new items such as email. The application contains a definition for the user interface and the possibility of sending information by email.

Keywords— web server; C# application; .NET framework; items search, web application, DLL library

I. INTRODUCTION

On the remote web servers are often new items, which the user would like to known. Due to the large number of items and server load the user would like to be familiar with the selected type each time, when application starts, or in defined time intervals [1].

The user therefore has a program that searches its own pre-defined servers and looking at it defined items. There is an application, which send by communication channel (email) new information corresponding to the search. While the application running, defined web server is tasked in defined time intervals to known, which items has been changed. In a match with the defined term user is notified via email. Selected information will be sent to the specified e-mail (email) and prepared for processing by the user. [2].

The application is developed in .Net Framework development environment in Microsoft Visual Studio 2010 in C#, with using DLL libraries, for faster application running. To access the web servers are used HTML tags of web pages, which are intended to find a match with the search term. The application server passes the whole page and after finding all the items are acquired data is processed and the results sent. In still running application servers are queried in a defined time. If a match is found the information items is automatically processed and sent. It is important to ensure that the servers were not too often been questioned [3]. Asking the web server in each second is totally unnecessary, the computer time used by the queried server, and a large amount of information would lead to overloading the server and there would not be in such short intervals no new information. Searching on servers starts about 1 time per hour [4], which is set as the default value.

By the long run the application immediately after finding the items, informations are sent by mail, to monitor the important section. The application allows monitoring of multiple servers. Each Web page is defined differently. This means that every page is necessary to define custom algorithm, which converts source code page to code applications so recognizable to any page in the application to be processed.

The aim of the algorithm code is read correctly, to recognize tags and web pages to identify signs that precede the searched information. Each website has its own style, which makes it impossible to use the same algorithm on different pages. The algorithm must be adapted to each site separately. It gives a lot of algorithms for many pages. User may not use all of the algorithms, but chooses what he needs. This application will not grow too much.

The application is designed for 3 different servers. Proposal applications will address issues related to creating web applications. Exist several possible solutions. The final solution will include a DLL, an application that facilitates the work with text, that is associated with parsing the html page. The application will have an interface to configure and is selected email as a communication channel or with a message informing the user if it does not want to set anything. The application is able to run independently at the first start by user. The application requires no additional modules and does not need its installation, so you can easily remove.
Applications as PC programs runs on a PC, when user start it, while the server is running continuously, which means that data coming into the server has no response to our system. The project aims to find ways for communicating server with our computer and sent him the selected data in a defined format. Searched data is placed on a web server and are presented with Web pages as plain text. Applications must download this information in an appropriate format, which will be suitable for further processing. Downloaded applications must be processed and then classified according to predefined criteria and process them in a convenient format.

The problem remained how to download data from the Web to your computer. One possibility would be an implemented algorithm on the server. It would be the simplest and best solution, if we had access to the server. Usually we don’t have a access to server and we have no possibility to connecting to server programs, you can only view. This solution would be best because the server itself must export the data into an HTML page. Possibility is add an algorithm that would send data to other places. In that case, we immediately receive the required data when the server is received. This option failed, and the server provides information only on his defined website. For this reason, it was necessary to obtain information from an external server pages.

Choice for some users could be RSS feeds. RSS feeds are in XML-format, which allows users to read news on the web. RSS technology allows users to subscribe to news, but only from a site that supports RSS feeds. This source is usually found on the pages where the content changes and is added very often. Originally, this format was used only for the actual transmission of news among servers so that they can easily refer to the actual articles on other sites. [5] RSS format provides the content of the article or its part, reference to the original article and metadata. This information is sent as an XML file called RSS feed, web feed or RSS feeds. Currently the most widely used version of RSS 2.0 is available under a Creative Commons license, which is managed by the Berkman Center for Internet & Society [6]. Why RRS? If the author uses the web feeds, the website visitors will appreciate the ability to obtain information without having to visit. Compared with other methods of promotion site visitor registration is not required and there are no such problems with their unwillingness to disclose personal information. This Web site will ultimately increase traffic because more people will come back. The concept of RSS allows site visitors to keep in constant contact [7].

Example RSS Feed:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<rss version="2.0">
  <channel>
    <title>RSS Example</title>
    <description>This is an example of an RSS feed</description>
    <link>http://www.domain.com/link.htm</link>
    <pubDate>Tue, 29 Aug 2006 09:00:00 -0400</pubDate>
    <lastBuildDate>Mon, 28 Aug 2006 11:12:55 -0400</lastBuildDate>
    <item>
      <title>Item Example</title>
      <description>This is an example of an Item</description>
      <link>http://www.domain.com/link.htm</link>
      <guid isPermaLink="false">1102345</guid>
      <pubDate>Tue, 29 Aug 2006 09:00:00 -0400</pubDate>
    </item>
  </channel>
</rss>
```

**Chyba! Nenalezen zdroj odkazů.**

However, many sites like ebay.com, aukro.cz not support RSS. For this reason, this technology is rejected, even though it seems very appropriate. If you need access to other servers, must be found other, universal method, which is valid for all sites. Given the complexity and diversity of sites, applications must process defined term in any style needed. One method is reasonably complex code browsing the site. Method assigning application to a server would be advantageous, the server would itself provided the data immediately after their gaining, but many servers not allow this option, because it would be overload by many applications, most of them in a short time leave function before the user has lost interest. The most promising method of RSS would solve many of the shortcomings of previous methods under consideration. It should not be big demands on disk size, would provide additional information about items and allowing them to introduce grading system in the items. The reason for not using this search method is that many sites are not supported by this technology.

A possible solution is to download the web code from the server. In this case, we get all the necessary information, but this information is hidden in the complex structure of the website. The problem with such data collection is the diversity of the structure of the web site server. This means that each page would be needed other algorithm which it can handle. The advantage would be to immediately obtain information from the server, but there remains the problem of the complexity of Web sites.
### New Solution

**Parsing**

In computer science and linguistics, parsing, or more formally, syntactic analysis, is the process of analyzing a text, made of a sequence of tokens (for example, words), to determine its grammatical structure with respect to a given (more or less) formal grammar. Parsing can also be used as a linguistic term, especially in reference to how phrases are divided up in garden path sentences.

Context-free grammars are limited in the extent to which they can express all of the requirements of a language. Informally, the reason is that the memory of such a language is limited. The grammar cannot remember the presence of a construct over an arbitrarily long input; this is necessary for a language in which, for example, a name must be declared before it may be referenced. More powerful grammars that can express this constraint, however, cannot be parsed efficiently. Thus, it is a common strategy to create a relaxed parser for a context-free grammar which accepts a superset of the desired language constructs (that is, it accepts some invalid constructs); later, the unwanted constructs can be filtered out. [9]

The solution of this problem is parsing. Parsing a website is able to process web content regardless of whether using a service such as RSS or not. In our application, code is loaded and algorithm located specific the type of items that appear in it. When you load the html code into applications, tokens are finded. Application function is to inform about items on the server do not need any change the sort. Important is only the content that was loaded. The application looks at the content page and try to find elements of the corresponding prescription items. It is important to select the appropriate page encoding. When a match is found with a search term, application passes to search for information following a search item, into the ending token. Thus, it searched the entire site.

In computing, a parser is one of the components in an interpreter or compiler, which checks for correct syntax and builds a data structure (often some kind of parse tree, abstract syntax tree or other hierarchical structure) implicit in the input tokens. The parser often uses a separate lexical analyzer to create tokens from the sequence of input characters. Parsers may be programmed by hand or may be (semi-)automatically generated (in some programming languages) by a tool.

The most common use of a parser is as a component of a compiler or interpreter. This parses the source code of a computer programming language to create some form of internal representation. Programming languages tend to be specified in terms of a context-free grammar because fast and efficient parsers can be written for them. Parsers are written by hand or generated by parser generators.
IV. Implementation

The new program solution is a to load HTML pages that allow him to work. The loading of the code is in the method btn_address_Click, which causes your click Go in the program window or timer. After you click "method retrieves the text box inpAddress address of the selected site and submit it as a parameter using the System.Net classes. Using the class StreamReader instance is loaded this page with the method ReadToEnd () from start to end. This code is then displayed in the text box text box1. For information, is displayed at the bottom the length of the loaded program code. Parsing the code run immediately after reading. Parsing is performed by DLL, allowing increased processing speed of the loaded code of your web page. The DLL exports the html page code and import main parsed text. The main reason to use DLL was the speed at which data is processed. Using DLL's also possible to use multiple libraries for multiple sites. The language used for writing DLL's is C language. To process the page content, you need to know, how the items are defined programmatically in the code. This is for every other page. In the code of your web pages need to be found how items are defined. For this reason, it is difficult to create a universal code applicable to all sites. This issue is dealing an HTML parsers that tags are specified for pages and parse the code page of the most powerful and modify them into XML, which is much more understandable. The solution is to create a custom parser in dll library, which passes through the code for the page as a string and the last record retrieved characters. These characteristics are compared with input tag. The agreement will enable the resulting string to which are offered all found items. This notation is to find the trailing tag. The finding that the resulting string to stop adding features loaded, but reading the code page goes on to reach the next item when the content is added to the resulting string. Reaching the end of the site is indicated by the program when the trailing length is the same code. Upon reaching the end of the resulting string is exported back to the application. The application then continues to work with string. The string to look for excess levels of pollutants in the air. Exceeding the values are displayed in tabs překročeni.Parsed string is modified into blocks and aligned in rows. Exceeding values are colored red. Then mail the application expects to be treated to send data. If you dont enter it, the application displays the excess value in a message. If the user entered an email, the data is sent to the selected address. For Mail, a new thread is created. To send e-mail is used for System.Net and System.Net.Mail library. MailMessage Class MailMessage and constructor () creates an instance for new email.

When start the application, is shown main window, see screen. pic.3. Next to Go button is a drop down menu with a choice of servers. Use button Go to start application. In the text box below the address found items are offered. Below the text box at the bottom of the window are listed additional information about the status of the program. On the File menu under settings find the settings for your application. In Settings are found email and the timer settings.

Application is developed in Microsoft Visual C# 2010 using the graphical – user interface.
V. TESTING OF DEVELOPED APPLICATION / TESTOVÁNÍ VYVINUTÉ APLIKACE-ŘEŠENÍ

The application was first tested using Visual Studio Tools, such as debug to check for errors arising during the application. Application is tested in several steps. The first step for proper functionality is load the HTML page and display the selected text box. If the code is displayed correctly, the code can be parsed. Parsing will using functions in the DLL library. Ongoing work in the parser DLL library is not required to only monitor the expected outcome of the operation. The result is a text written in lines. You may not find any elements, which is not shown in results of operations. Another part is to check on that text, the application is able to find and correctly identify the limits of air excess. There needs to check with the original web page from which the code is executing. Applications must recognize the measuring stations, its values and exceeded. Applications must also write the limits to Excess tabs and in a right format.

During a text format change also marks the limits with red. These limits are consistent with the limits in the tab Exceed. Now the application process the input text into tabular form and displays the results of the application or sent to selected email. The last control is the email that arrived at the destination address and whether it contains text that should be sent.

Application must be tested as a whole on a different computer than the computer where it was created. Application was transferred to another computer. For test was deliberately chosen computer with another operating system. The selected computer with installed Windows XP Professional was a good candidate for testing the application, XP is still widely used. The selected computer does not contain any debugging tools for applications that could reduce the credibility test. The application was therefore only work with those codes that were supplied with it. This way you can test the portability of the application. Application for transfer to another computer worked after transferring msvcr100d.dll library. Without this library can not import other libraries and the reported absence of the library. After connecting the library to function properly. As a whole, it is necessary to test the application as a whole.

The application has been started and tested on a random server that was written. Applications under the assumption loaded HTML page, select an appropriate parser, parsed the page and display the result in the application window. The screen 7 and screen 8 shown the difference between an application on Windows 7 and Windows XP. Application of the tested system functioned properly with minor differences in appearance of applications. The developed application behaves according to the requirements that were expected from it. After the first launch is expected to open by default a defined window. After you press the Go button, HTML page was retrieved, and result was displayed. Application correctly identified exceeded and sent an email to the selected address. The application has been tested on other servers and there were no errors during testing when running the program.
VI. CONCLUSIONS / ZÁVĚRY

The aim was to develop and test an application that on the remote web server looks for items that are intelligently processed and the results displayed to the user. I managed to program the application to the three remote servers, collecting information on pollutants in the air. For each page I wrote my own parser using DLL libraries that the code page gets the necessary information. Input text from the parser was processed to clear the form and display in your application.

A major problem with this application is the processing speed of long strings. To process was used DLLs and application rate was increased. Benefits of applying for admission to me was coded HTML pages, email, object-oriented programming, word processing, threads and capabilities of language C#.

Applications can be developed further and the possibility of its use, where they need to know some data, without interfering with the system. Application can be used as a gadget for Windows, or as part of a Web service.

REFERENCES / REFERENCE

[1] Saul C. Leite, Marcelo D. Fragoso: Heavy traffic analysis of state-dependent parallel queues with triggers and an application to web search systems. Department of Systems and Control, National Laboratory for Scientific Computing (LNCC), Av. Getulio Vargas 333, Petropolis, RJ, CEP:25651-075, Brazil


[3] Mauricio Marin, Veronica Gil-Costa: Sync/Async parallel search for the efficient design and construction of web search engines. Informatic Engineering Department, University of Santiago of Chile, Chile


