

DETAILED STUDY OF AGGREGATOR FOR UPDATES

Iva Chauhan¹, Priyanka Dashrathsinh Puvar², Bhagirath Parshuram Prajapati³, Kinjal Patel⁴

^{1,2,3,4} Computer Engineering Dept, ADIT, VVNagar

Abstract

Nowadays, personal weblogs, news web sites, and discussion forums are publishing RSS feeds for their subscribers to retrieve new postings. There has been a dramatic increase in the use of XML data to deliver information over the Web. Everyday millions of new articles are published and presented by RSS feeds. Also, news feeders, newsgators are available for subscribers or the readers by the individual sites, the news feeders regularly pull articles from the Web sites, the aggregated effect by all news feeders puts an enormous load on many sites. We propose a blog aggregator approach where a central aggregator monitors and retrieves new postings from different data sources and subsequently disseminates them to the subscribers to alleviate such a problem. We study how the aggregator should monitor the data sources to quickly retrieve new postings using minimal resources and to provide its subscribers with fast news alert. Thus, this reduces the load on the monitored data sources by a significant amount.

1. INTRODUCTION

RSS (Rich Site Summary or Really Simple Syndication) is an Extensible Markup Language (XML) file which is used by sites for syndication of their articles on the Internet [7]. RSS is a family of web feed formats used to publish frequently updated works—such as blog entries, news headlines, audio, and video—in a standardized format. An RSS document includes full or summarized text, plus metadata such as publishing dates and authorship. They benefit readers who want to subscribe to timely updates from favored websites or to aggregate feeds from many sites into one place [6]. RSS feeds can be read using software called an "RSS reader", "feed reader", or "aggregator", which can be web-based, desktop-based, or mobile-device-based. A standardized XML file format allows the information to be published once and viewed by many different programs. The RSS reader checks the user's subscribed feeds regularly for new work, downloads any updates that it finds, and provides a user interface to monitor and read the feeds. RSS allows users to avoid manually inspecting all of the websites they are interested in, and instead subscribe to websites such that all new content is pushed onto their browsers when it becomes available.

Everyday millions of new articles are published and presented by RSS feeds. Recently, there has been a dramatic increase in the use of XML data to deliver information over the Web. In particular, personal weblogs, news Web sites, and discussion forums are now delivering up-to-date postings to their subscribers using the RSS protocol [2].

As the popularity of the RSS feeds and news feeders grows, however, they have started to put an enormous load on many sites, since each news feeder contacts every subscribed Web site constantly – often more than once every hour so many sites have experienced enormously high loads [1].

Aggregator is client software or a web application which aggregates syndicated web content such as news headlines,

blogs, podcasts, and video blogs in one location for easy viewing. Aggregators reduce the time and effort needed to regularly check websites for updates, creating a unique information space or personal newspaper. Once subscribed to a feed, an aggregator is able to check for new content at user-determined intervals and retrieve the update.

1.1 Types Of Aggregators

There are different types of aggregators related to their functioning. They retrieve a particular type or a similar type of information for which they are coded. The following are types of aggregators:

- Data Aggregator.
- News aggregator.
- Poll aggregator.
- Review aggregator.
- Search aggregator.
- Social network aggregation.
- Video aggregator.
- Blog aggregator.
- Payment aggregator.

These are the types of aggregators but we are suggesting an aggregator which will retrieve almost all types of information, such that the subscribers don't have to take on many different aggregators.

We are designing a blog aggregator where a central aggregator monitors and retrieves new postings from different data sources like news weblogs, personal web blogs, discussion forums, and subsequently disseminates the new postings and updates to the subscribers [1].

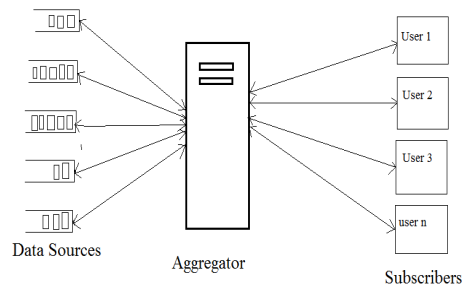


Fig.1 (a) Blog Aggregator Sample [1]

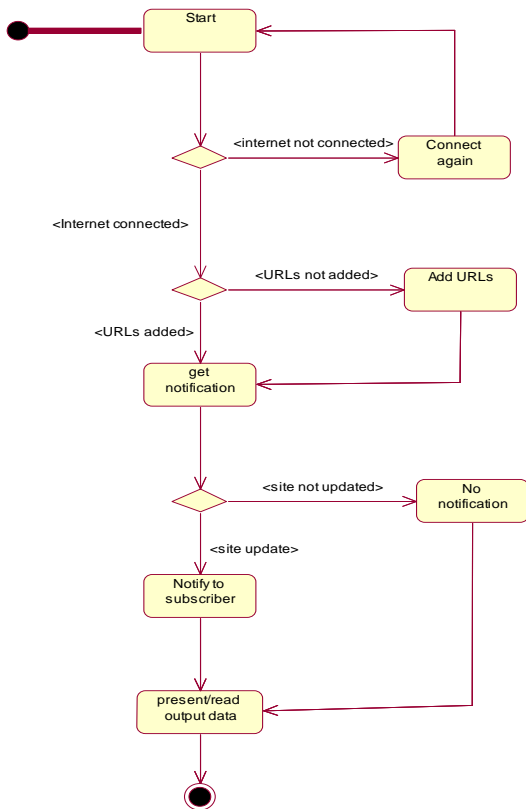
It works like this- the user subscribes to data sources, the aggregators knows what all site are subscribed. It connects to those sites i.e. their RSS feed links and gives the updates as soon as they arrive to those sites. It collects those information or the updates and notifies the subscribers about it and thus the subscribers can read those updates whenever they need.

2. TECHNICAL BACKGROUND

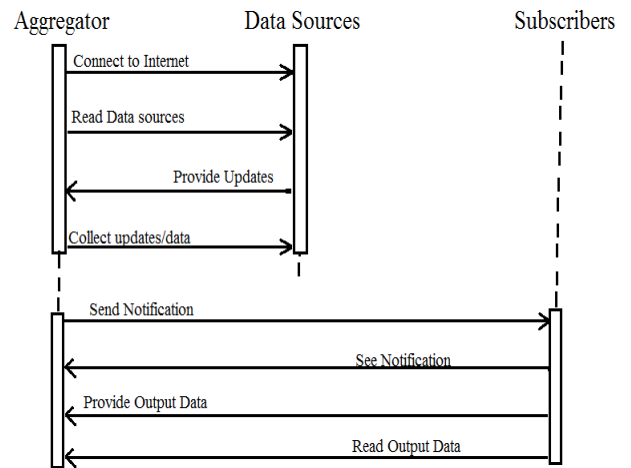
Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. Visual Studio includes a code editor supporting IntelliSense as well as code refactoring. Here we are going to use C#.NET in the coding part for developing the aggregator. For reading the RSS feeds we would be requiring XML for coding.

2.1 Design Analysis

Activity Diagram



Sequence Diagram



2.2 Implementation

Here, it is compulsory to be connected to the internet. We have built RSS reader that harvests the feeds from the selected channels (those are specified in the combo box) and lists them into the 'ListView' along with the time and date of being updated. The items or the feeds that appear are related to the specific channel only. The user can read the article or feed as he/she will be redirected to the web page of the site. The following is the flow of what is happening in the windows application form.

On loading the application, the user has to select the channels specified in the 'combo box' and then click on 'Update' button. As it will take some time to harvest the data, we have used Cursors. After this, there appears a list of news headlines in the 'ListView' under 'Item' and under 'Time' there appears the date and time of the feed whenever it was update on the site. Above 'ListView' is 'items' i.e. number of items in the ListView and 'Updated' shows when last time the channel in the application was updated. Below is 'TextBox' with multiline which shows some of the description about the item that is selected. One can double-click on the item or click on 'View' button in order to get redirected to the specific channels' link where the feed is present.

The following are the screen-shots that will provide the exact knowledge of the application which we have built. They show the list of feeds or sites provided at present for harvesting the updates, interaction of user with the application.

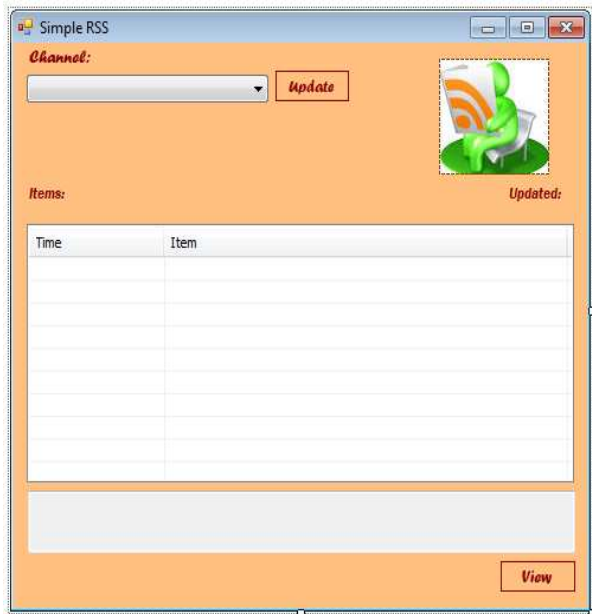


Fig.2.2 (a) Application Loaded

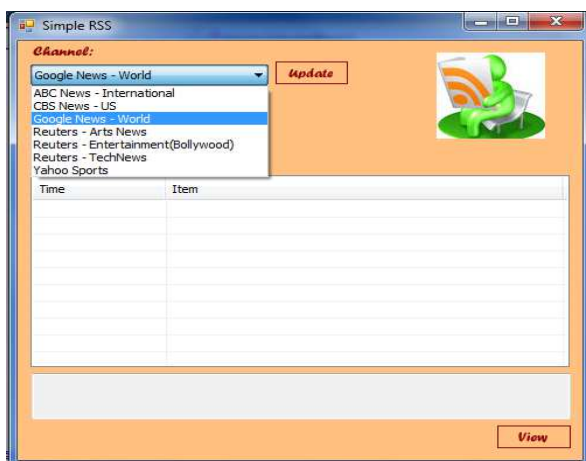


Fig.2.2 (b) Selecting Data Sources



Fig.2.2(c) Output after Deployment

If the internet is not connected, then following is displayed.

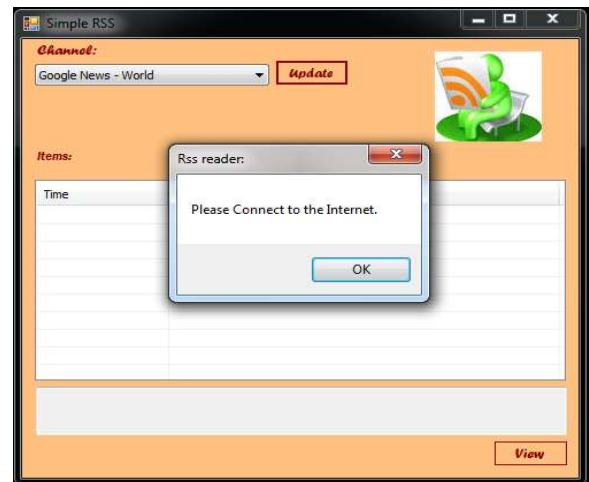


Fig.2.4 (d) Internet not connected

3. CONCLUSION

This is a brief idea of aggregator which uses only some of the selected data sources, but we intend to make such an aggregator that user just selects any of the URL which he/she wants to follow and pastes it into the aggregator so that it retrieves the updates whenever it is connected to the internet. Also, the expected output will be such that the users would be given notification about the updates that arrive at the data sources that are subscribed by the users. As and when the user is connected to internet, the application will harvest the new updates from data sources and notify the user. Aggregator can be made for mobile devices also as each and every person today uses mobile for their important work and to know all the world news. It will prove to be very much helpful for the people all over as they don't need to visit the particular site for checking the updates. They just have to paste the URL and rest of the job will be done by the aggregator.

REFERENCES

- [1] "Efficient Monitoring Algorithm for Fast News Alert" by Ka Cheung Sia Junghoo Cho UCLA Computer Science Department, Los Angeles, CA 90095
- [2] "Intelligent RSS Aggregator" by YasharMosfeghi, Dept. f Computer Science, University of Glasgow.
- [3] www.geekpedia.com/tutorial147_Creating-an-RSS-feed-reader-in-Csharp.html
- [4] www.dotnetspider.com/resources/43786-How-parse-RSS-feeds-ASP.NET.aspx
- [5] B. E. Brewington and G. Cybenko. How dynamic is the web? In WWW Conference, 2000.
- [6] <http://www.modulusfe.com/rss>
- [7] Ralph Mosely and M.T Savaliya "Developing Web Applications", Wiley India publications, edition 2012.