GoogleSummerOfCode SitemapCrawler

Title :		GSOC 2015 Proposal
Issue :		NUTCH-1741 - Support Sitemap Crawler in Nutch 2.x
Student :	Cihad Güzel - cguzelg@gmail.com	
Mentors :		Lewis John McGibbney, Talat Uyarer

Abstract

The url's can be got from only pages that were scanned before in nutch crawler system. This method is expensive. Also, the degrees of importance and "change frequance" of these urls are not known only guessed. But, it is possible to find the whole of urls in a up-to-date sitemap file. For this reason, sitemap files in website should be crawled. Nutch project will have that support of sitemap crawler thanks to this development.

Introduction

Sitemap is a file guiding to crawl website in a better way and it has different file formats (such as simple text format, xml format, rss 2.0, atom 0.3 & 1.0).

It is possible to find the whole of urls in a up-to-date sitemap file. Websites can be crawled faster by means of sitemap crawler that will be developed. In addition, some knowledge can be detected such as "change frequance", "last update time" and "the priority" of the pages. Shortly, a better url list will be got easily and fast from sitemap file thanks to this software. It is another advantage that this process is under the control of the user. Finally, when the project concluded;

- Nutch project will have that support of sitemap crawler thanks to this development.
- Better url list will be got by eliminating the sitemaps according to criteria of quality.
- The sitemaps not wanted can be ignored
- Sitemap crawler can be followed by reporting the errors occuring during crawling.
- The management and configuration of sitemap crawler are under the control of user.

Project Details:

It is aimed to power nutch project by sitemap crawler support. The main target is to detect the sitemap having correct urls and to be crawled. It is easy and fast to find correct ursl by sitemap crawler. The software will make following features possible.

1. **sitemap detection:** Sitemap files will be detected automatically, if available.

- **sitemap list injection:** Sitemap urls will be injected by using Nutch injection
- **sitemap blacklist:** The sitemaps urls that are not wanted to be crawled can be blocked.
- **sitemap ranking mechanism & sitemap filter:** The sitemaps will be eliminated according to criteria of quality. The urls in sitemap will be prioritised according to priority points.

- **The current nutch ranking mechanism will not be changed according to the sitemap urls.**
- **The crawler can be support diffently sitemap format(simple text format, xml format, rss 2.0, atom 0.3 & 1.0) :** Only XML formats will be crawled for now. The support of other formats is out of project.
- **"Change frequence" mechanism must be supported by the crawler.**
- **Errors and info data will be reported.**
- **Supporting multi-sitemap.**
- **Sitemap constraint:** The maximum sitemap size can not be greater than 10 MB and the maximum urls can not be greater than 50,000 in a sitemap file.
- **Sitemaps must have only inlink. Outlinks must be ignored.**
- **Firstly sitemap crawler will support HBase. Other DB's are not supported in this project.**

- **Sitemap crawler is the part of Nutch Life Cycle [3].** Sitemap crawler is designed according to these cases:
    - Sitemap urls can be injected from seedlist.
    - Sitemap files can be detected automatically from sites crawled.
    - It can be wanted to crawl only sitemaps.
    - It can be wanted to crawl urls except sitemap.
    - A sitemap file can give referance another sitemap file.
    - Sitemap file can be in zip format.
    - Sitemap file may be larger than 50mb. In case of this some limits must be defined.
    - Sitemaps file may include more url than 50,000. In case of this some limits must be defined.

The advatages of the process of developing project.

1. The new features that will be developed can be entegrated easily thanks to the nutch pluginer design and nutch life cycle.

- The current nutch plugins can be used.

- There are some studies about sitemap crawler in Nutch project (NUTCH-1741 [1], NUTCH-1465 [2]). The process improves by taking hand the weak and strong sides of the project

Timeline:

Project development process can be divided into two steps. Firstly, nutch crawler life cycle will be updated for sitemap crawler. Sitemap will be crawled in a simple way before midterm. In the next stage, Other issues will be completed such as sitemap detection, filter & ranking mechanizm, documentation and tests.

Pre-GSoC : The studies and the comments on NUTCH-1741 [1] and NUTCH-1465 [2] will be followed.

- Week1 (25May-31May): sitemap url injection will be done.
- Week2 (1June-7June): Sitemap detection will be done. FetcherJob will be updated for sitemap.
- Week3-4 (8June-21June): The parser process is updated for sitemap file parser. New parser plugins can be developed.
- Week5 (22June-28June): DbUpdaterJob is updated for sitemap.
- Midterm(26June-3 July): Up to this stage, sitemap life cycle has been developed according to the outline. Sitemap crawler runs simply. The process until now and from now on will be evaluated.
- Week6-7 (29June-12July): Sitemap ranking mechanism will be developed.
- Week8 (13July-19July): Sitemap black list, sitemap file detection yaplacak ve error detection yaplacak
- Week9 (20July-26July): Frequent mechanism will be developed
- Week10 (27July-2August): The filter plugins will be updated or new filter plugins is will be developed.
- Week11 (3August-9August): Code review and code cleaning.
- Week12-13 (10August-23August): Further refine tests and documentation for the whole project.

  Features that will be developed after GSOC: Sitemap crawler report page, Sitemap monitoring page, Video Sitemaps crawler.

Reference:

*[1] https://issues.apache.org/jira/browse/NUTCH-1741
*[2] https://issues.apache.org/jira/browse/NUTCH-1465
*[3] https://issues.apache.org/jira/secure/attachment/12707721/SitemapCrawlerLifeCycle.pdf

Reports

- Weekly Report
- Midterm Report
- Final Report

Documentation

Documents will be added here.

Source Code

source code on github

Jira Issues

- https://issues.apache.org/jira/browse/NUTCH-1741
- https://issues.apache.org/jira/browse/NUTCH-1465