

Enrich the E-publishing Community Website with Search Engine Optimization Technique

R. Vadivel¹ and Dr. K. Baskaran²

¹PhD Scholar, Dept. of Computer Science,
Karpagam University, Coimbatore,
Tamilnadu India 641 021

²Assoc. Professor, Dept. of CSE and IT,
Govt. College of Technology, Coimbatore,
Tamilnadu India 641 013

ABSTRACT

Internet has played vital role in the online business. Every business peoples are needed to show their information clients or end user. In search engines have million indexed pages. A search engine optimization technique has to implement both web applications static and dynamic. There is no issue for create search engine optimization contents to static (web contents does not change until that web site is re host) web application and keep up the search engine optimization regulations and state of affairs. A few significant challenges to dynamic content poses. To overcome these challenges to have a fully functional dynamic site that is optimized as much as a static site can be optimized. Whatever user search and they can get information their information quickly. In that circumstance we are using few search engine optimization dynamic web application methods such as User Friendly URL's, URL Redirector and HTML Generic. Both internal and external elements of the site affect the way it's ranked in any given search engine, so all of these elements should be taken into consideration. Implement these concepts to E-publishing Community Website that web site have large amount of dynamic fields with dynamic validations with the help of XML, XSL & Java script. A database plays a major role to accomplish this functionality. We can use 3D (static, dynamic and Meta) database structures. One of the advantages of the XML/XSLT combination is the ability to separate content from presentation. A data source can return an XML document, then by using an XSLT, the data can be transformed into whatever HTML is needed, based on the data in the XML document. The flexibility of XML/XLST can be combined with the power of ASP.NET server/client controls by using an XSLT to generate the server/client controls dynamically, thus leveraging the best of both worlds.

Keywords: Search Engine Optimization (SEO), community website, Dynamic web, Friendly URLs, ASP.Net, XML/XSLT and dynamic controls.

1. Introduction

If we have a website, we definitely need it to be a friend of search engines. There are several ways to attract

visitors to our website, but in order to make searchers know about our website, search engine is the tool where we need to prove our contents. If we are just having a static HTML content, then there is no much problem in promoting it. But where in today's world of Content Managed Websites and eCommerce Portals we need to look further and implement a few more techniques in order to make the site more prominent to robots. In this article we will discuss how we can develop a SEO Friendly website where the content is driven from the Database with a Content Management System which is developed using ASP.NET. We will learn to build a simple CMS driven site with no nonsense URL, which Search Engines invite.

Search Engine Optimization (SEO) is often considered the more technical part of Web marketing. This is true because SEO does help in the promotion of sites and at the same time it requires some technical knowledge – at least familiarity with basic HTML. SEO is sometimes also called SEO copyrighting because most of the techniques that are used to promote sites in search engines deal with text.

Generally, SEO can be defined as the activity of optimizing Web pages or whole sites in order to make them more search engine-friendly, thus getting higher positions in search results.

A Search Engine Optimization (SEO) is very popular term in web application industry. We can implement the SEO concepts to both applications static and dynamic web application. No matter for implement SEO to static web application. We have just followed up the SEO rules and conditions. We have to implement to dynamic / MVC web application it should be an insignificant complicate and use some tricky.

The specific objective is to develop the e-publishing community website with search engine optimization technique. There is no specified web technology in dynamic web applications. We can use any Microsoft or

any other corporation software. In my work .NET has played major role.

To understand dynamic content, it's important to have an idea of its opposite, static content. The term static content refers to web content that is generated without using a data source such as a database. Essentially, the site viewer sees exactly what is coded in the web page's HTML.

With dynamic pages, a site can display the same address for every visitor, and have totally unique content for each one to view. For example, when I visit the social networking site Face book (facebook.com), I see <http://www.facebook.com/home.php> as the address in my web browser, but I see a unique page that's different from what anyone else sees if they view that page at the same time. The site shows information about my friends in my account, and different information for each person in his account, or for someone who has no account.

Not all dynamically generated content is unique to every viewer, but all dynamic content comes from a data source, whether it's a database or another source, such as an XML file.

1.1 SEO in web application

A web application has playing most important role in the online business.

A million of static and dynamic web pages are available in the internet and million users can have used those web pages for their required information.

In this circumstances search engine optimization is play most important play between user and web applications.

In Million web pages are available the user should need their specific search criteria such as business man have search the own needs, students have search their own needs and etc.,

Our aim is whatever user search and they can get information their information quickly. In that situation we are using few search engine optimization methods and concepts such as a crawler, an index (or catalog) and a search interface, search engine algorithms and page rank algorithms.

Search engines take advantage of reverse broadcast networks to help save you time and money. Search allows you to "sell what your customers want, when they want it!"

Search Engine Optimization is the science of customizing elements of your web site to achieve the best possible search engine ranking. That's really all there is to search engine optimization. But as simple as it sounds, don't let it fool you.

Both internal and external elements of the site affect the way it's ranked in any given search engine, so all of these

elements should be taken into consideration. Good Search Engine Optimization can be very difficult to achieve, and great Search Engine Optimization seems pretty well impossible at times.

Optimization involves making pages readable to search engines and emphasizing key topics related to your content. Basic optimization may involve nothing more than ensuring that a site does not unnecessarily become part of the invisible Web (the portion of the Web not accessible through Web search engines).

2. Existing System

Previously SEO have implemented in static commercial / non-commercial web sites. In this way there is no dynamically site map and have not well-defined RSS feed for those implementations and there is no specific way to find the back links.

Dirty URLs

Complex, hard-to-read URLs are often dubbed dirty URLs because they tend to be littered with punctuation and identifiers that are at best irrelevant to the ordinary user. URLs such as <http://www.example.com/cgi-bin/gen.pl?id=4&view=basic> are commonplace in today's dynamic web. Unfortunately, dirty URLs have a variety of troubling aspects, including:

URLs are difficult to type

The length, use of punctuation, and complexity of these URLs makes typos commonplace.

URLs do not promote usability

Because dirty URLs are long and complex, they are difficult to repeat or remember and provide few clues for average users as to what a particular resource actually contains or the function it performs.

URLs are a security risk

The query string which follows the question mark (?) in a dirty URL is often modified by hackers in an attempt to perform a front door attack into a web application. The very file extensions used in complex URLs such as .asp, .jsp, .pl, and so on also give away valuable information about the implementation of a dynamic web site that a potential hacker may utilize.

URLs impede abstraction and maintainability

Because dirty URLs generally expose the technology used (via the file extension) and the parameters used (via the query string), they do not promote abstraction. Instead of hiding such implementation details, dirty URLs expose the underlying "wiring" of a site. As a result, changing

from one technology to another is a difficult and painful process filled with the potential for broken links and numerous required redirects.

3. Related Works

There is a three technologies have been used that is 1. User Friendly URL's, 2. URL Redirector and 3. HTML Generic.

An e-publishing community website has been used Microsoft .NET web application with ASP.NET and C#. In this application has used data model and business layer in separate module and its like a DLL (Dynamic Link Library) and we have started to created and converted dynamic URL's into Static URLs.

The URLs converting code first we must grab the incoming URL and split the extension of the page. Which pages have ".html" extension we should redirect that page to related ".aspx" page on code behind they have executed business logic or data manipulation or whatever functionality need, and display to the end user exact content for that particular page with proper Meta description and keywords. In this time of period user can only view ".html" page but all other logics will execute the code behind.

3.1 Dynamic Content and SEO

SEO for dynamic content poses a few significant challenges. Luckily, you have ways to overcome these challenges to have a fully functional dynamic site that is optimized as much as a static site can be optimized. This section discusses the pitfalls of dynamic sites, and how to overcome them to create fully optimized dynamic sites.

3.2 Challenges for Optimizing Dynamic Content

Here are some common areas of dynamic sites that provide setbacks for humans as well as search engine spiders.

3.2.1 Dynamic URL

A Dynamic URL is an address of a dynamic web page, as opposed to a Static URL, which is the address of a static web page. Dynamic URLs are typically fairly cryptic in their appearance. Here's an example from <http://www.meetingtechonline.com/article> for a product called Kindle:

http://www.meetingtechonline.com/article/product/B000F173MA/ref=amb_link_7646122_1?pf_rd_m=ATVPDKIKX0DER&pf_rd_s=center-1&pf_rd_r=1FYB35NGH8MSMESECBX7&pf_rd_t=101&pf_rd_p=450995701&pf_rd_i=507846

Notice that the URL doesn't contain any information about the item's product type, or anything about the item's name. For a well-trusted site like Amazon, this is not a problem at all. But for a new site, or for a site that's gaining credibility and popularity, a better solution can help search results by showing a searcher some relevant keywords in the page's URL. Here's an example of something a little more effective:

<http://www.meetingtechonline.com/article/products/electronics/kindle/>

While search engines may not have problems indexing URLs with variables, it's important to note that highly descriptive URLs like the one just shown can get more clicks in searches than cryptic URLs. If searchers can clearly see keywords that have to do with the content they're looking for in your page's URL.

3.2.2 Logins and other forms

Login forms can restrict access to pages not only to users, but also search engines. In some cases, you want pages behind logins made searchable. In those cases, you can place code in those pages that determines whether the person visiting has access to view that content, and determine what to do from there.

Other web forms, referring to content in <FORM> tags, can restrict access to pages as well. While Google has revealed that googlebot can go through simple HTML forms (see <http://googlewebmastercentral.blogspot.com/2008/04/crawling-through-html-forms.html>), not all search engines follow this same process, which means content hidden behind forms may or may not be indexed.

3.2.3 Cookies

Web cookies are small bits of data that are stored in a user's web browser. Cookies are used frequently on the Web for storing temporary data like shopping cart information or user preferences. Pages that require cookies can block spiders because spiders don't store cookies as web browsers do.

3.2.4 Session IDs

Session IDs are similar to cookies in that if you need them to view pages, then spiders don't index those pages.

3.2.5 Hidden pages

Sometimes, pages on a website are hidden from search engines because they're buried too deep in a site's architecture. For example, a page more than three clicks deep from the home page of a website may not be crawled without an XML sitemap. Other pages that may be hidden include pages only visible via a site search.

3.2.6 JavaScript

Search engines don't index content that requires full-featured JavaScript. Remember that spiders view content in much the same way as you would if you were using a browser with JavaScript disabled. Text that is created using JavaScript, and therefore only accessible with JavaScript enabled, will not be indexed.

3.3 Ways to Optimize Dynamic Content

Dynamic content is often necessary in websites. In addition, content that is easily changed through an outside data source helps keep a site's content fresh and relevant. This increases its value to search engines. You don't need to worry that because your site is dynamic, your content won't be indexed. You just need to make sure you're following the appropriate guidelines when using dynamic content in order to keep your site optimized. Here are some things you can do to optimize your sites that contain dynamic content.

3.3.1 Creating static URLs

Dynamic URLs, especially dynamic URLs with vague names, can be a turnoff to searchers. In order to have friendly URLs, you want to rewrite your dynamic URLs as static URLs.

Blogs powered by wordpress or Blogger make it easy to convert dynamic links to static links. Blogger automatically creates static URLs, and with wordpress you need only a simple change in your settings. For wordpress, log in to your administrator account, and then, under Settings, click the Permalink button. From there, you simply select a static URL publishing method or create a custom one and save the changes. Nice!

If your site isn't powered by a blogging application, you need to rewrite the URLs manually. The process is somewhat complex, and it requires modifying your .htm access file. Because modifying your .htm access file can cause permanent changes to your website, you want to either practice on a testing server or know exactly what you're doing before using these techniques on a production server. To test this process on a testing server, you can download and install a testing server (discussed in Chapter 4), and then download all or part of your website to your computer. That way, changes you make on your local computer don't affect your live site.

3.3.2 Optimizing content hidden by forms

The fact that web forms can hide content can be a good thing, but sometimes forms hide content you may not want hidden. Login forms (forms that require a user name and password) can potentially block search engines if a login form is the only way to access that information. Of course, sometimes this feature is intentional, like for protecting bank account information on a banking site. For non-login forms, assuming that search engines index content that's accessible only by filling out text fields or other form elements is dangerous. Further, it's equally

dangerous to assume that search engines don't index content that's accessible only via non-login forms. If you want your form's hidden content to be indexed, make sure to give access to it in ways other than through a form alone. If you don't want the content to be indexed, make sure to hide it from search engines via robots.txt, or some other method.

Typically, content that's viewable only after a user is logged into an account isn't necessary to index. If you have content that you want indexed hidden in a login-only area, consider taking that content out of the restricted area so it can be indexed.

4. Results

Successfully implemented search optimization e-publishing community web application with help of those technologies. Here show the few mock-up screen shots.

Fig – 1 and 2 have found relevant search information on google search engine.

Fig – 3 has displayed detail information about that article which found in google search results.

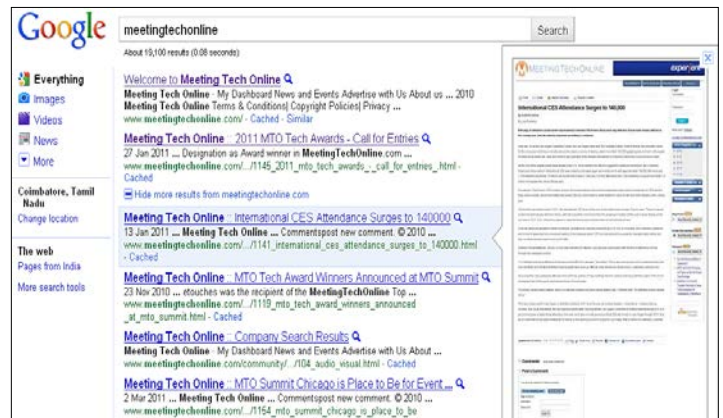


Fig 1: Getting relevant search information from google search engine– I.

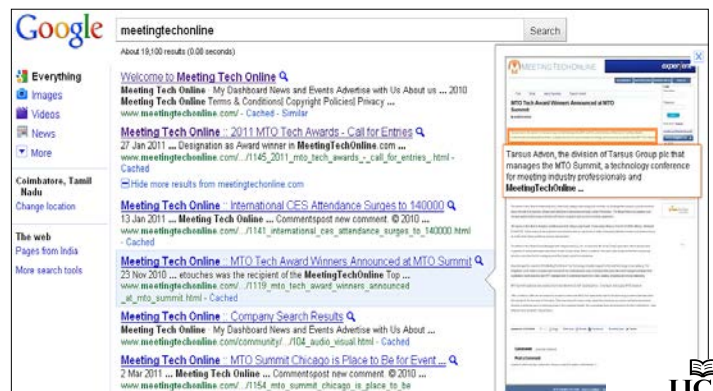


Fig 2: Getting relevant search information from google search engine– II.

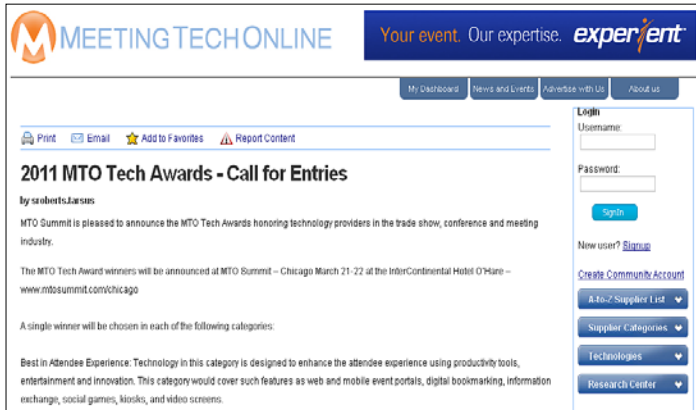


Fig 3: Detail information about that article which found in google results.

Hence we have successfully implemented Search Engine Optimization technique for e-publishing community web application.

5. Conclusions

A Search Engine Optimization has been implemented in E-Publishing community website. It has been used model view controller patterns to develop the web application and their techniques such as URL Redirector, HTML Generic, .NET security tools. The proposed is implementing the multiple query searches and personalized concept based clustering.

Most of the tips presented here are fairly straightforward, with the partial exception of URL cleaning and rewriting. All of them can be accomplished with a reasonable amount of effort. The result of this effort should be cleaned URLs that are short, understandable, permanent, and devoid of implementation details. This should significantly improve the usability, maintainability and security of a web site. The potential objections that developers and administrators might have against next generation URLs will probably have to do with any performance problems they might encounter using server filters to implement them or issues involving search

engine compatibility. As to the former, many of the required technologies are quite mature in the Apache world, and their newer IIS equivalents are usually explicitly modelled on the Apache exemplars, so that bodes well.

As to the search engine concerns, fortunately, Google so far has not shown any issue at all with cleaned URLs. At this point, the main thing standing in the way of the adoption of next generation URLs is the simple fact that so few developers know they are possible, while some who do are too comfortable with the status quo to explore them in earnest. This is a pity, because while these improved URLs may not be the mythical URN-style keyword always promised to be just around the corner, they can substantially improve the web experience for both users and developers alike in the long run.

Reference

- [1] Kenneth Wai-Ting Leung, Wilfred Ng, and Dik Lun Lee, "Personalized Concept-Based Clustering of Search Engine Queries", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 20, NO. 11, NOVEMBER 2008
- [2] AARON MATTHEW WALL, "Search Engine Optimization", JUNE 2008
- [3] Ernest Ackermann & Karen Lartman, "The information Specialist Guide to Searching & researching on the Internet and World Wide Web", Fitzroy Dearbon Publishers, 1999.
- [4] R.Elmasri and S.B. Navathe, "Fundamentals of Database Systems", 2nd Edition, Menlo Park, CA: Addison- Wesley 1994.
- [5] Jeff Ferguson, Brian Patterson, Pierre Boutquin " C# Bible", John Wiley and Sons, June 2002
- [6] Wei Meng Lee, "C#.net Web Developer's Guide", Syngress, January 1970
- [7] Jose Mojica, C# Web Development for ASP.NET, Peachpit Press, March 2003
- [8] Ke Yi , Feifei Li , Graham Cormode , Marios Hadjieleftheriou , George Kollios , Divesh Srivastava, Small synopses for group-by query verification on outsourced data streams, ACM Transactions on Database Systems (TODS), v.34 n.3, p.1-42, August 2009
- [9] HweeHwa Pang , Jilian Zhang , Kyriakos Mouratidis, Scalable verification for outsourced dynamic databases, Proceedings of the VLDB Endowment, v.2 n.1, August 2009
- [10] Alberto Trombetta, Danilo Montesi, "Equivalences and Optimizations in an Expressive XSLT Fragment" IEEE

TRANSACTIONS ON KNOWLEDGE AND DATA
ENGINEERING, VOL. 20, NO. 11, JULY 2009

- [11] Kyriakos Mouratidis , Dimitris Sacharidis , Hweehwa Pang, Partially materialized digest scheme: an efficient verification method for outsourced databases, The VLDB Journal — The International Journal on Very Large Data Bases, v.18 n.1, p.363-381, January 2009
- [12] HweeHwa Pang , Kyriakos Mouratidis, Authenticating the query results of text search engines, Proceedings of the VLDB Endowment, v.1 n.1, August 2008
- [13] <http://www.macronimous.com/> 2009
- [14] <http://www.webtop.com.au/seo> 2009
- [15] <http://www.seocompany.ca/seo/seo-techniques.html>
- [16] <http://searchengineland.com/21-essential-seo-tips-techniques-11580>
- [17] <http://msdn.microsoft.com/en-us/library/ms972974.aspx>