Zoom V7 Search Query Response Specification

XML Response Specification for Zoom V7 Search Engine's Query

Document: Zoom V7 Search Query Response Specification
Author: S.K. Tan, R. Leung
Company: Wrensoft Web Development (http://www.wrensoft.com)
Date: 22 September 2014
Edition: Ed 1.0
File: Zoom V7 Response Spec Ed 1.0.doc

TABLE OF CONTENTS

REVISION HISTORY

Revision	Revision History	Author	Date
V7 Ed 1.0	Updated for V7. New major release.	RL	22/Sep/2014
V6 Ed 1.0	- Updated for V6. New major release.	RL	26/Feb/2009
V5 Ed 2.0	- Need to complete "Zoom Indexer Configuration Requirement" and "Zoom Search Query Syntax".	SK	28/Nov/2006
V5 Ed 1.0	- Renamed document to "Zoom V5 Response Spec Ed 1.0" - Should we omit or summarize "Chapter 3: Zoom Search Response Syntax"	SK	16/May/2006
	Document creation. Draft version - investigate why OpenSearch recommends empty element should end with "/>" - Zoom Search Query Syntax, what is the HTTP search request syntax? - Zoom Search Response Elements, do we want to create a web page to document Zoom's namespace? - Zoom Search Response Elements, give examples of each criteria? For example, properly nested, properly closed, attribute values in double quotes… - Zoom Search Response Elements, expand more on each element if necessary - Zoom Search Response Elements, what are the default values for missing elements? - Zoom Search Response Elements, do we want to elaborate more on Autodiscovery or refer to OpenSearch URL? - Zoom Indexer Configuration Requirement, what are the configurations when indexing your web pages so that all slaves return the same format?	SK	03/May/2006

1.0   INTRODUCTION

Wrensoft Web Development produces the Zoom Search Engine software package, which allows users to add customizable and comprehensive search functionality to their website and CDs. More information on Zoom is available at http://www.wrensoft.com/zoom.

Zoom Search Response is an XML response format that is returned by a search query from a web server that provides Zoom V5 search functionality. By introducing several OpenSearch™ Response elements and Zoom Search Response elements into RSS 2.0, it provides the data necessary for syndicating web results. RSS 2.0 was chosen as Zoom Response format due to its wide-spread support and compatibility with most web applications, as well as its extensibility.
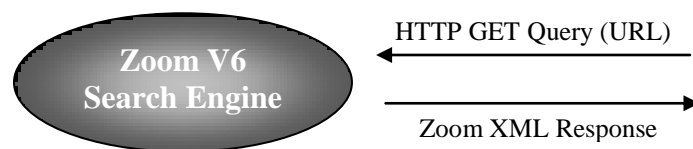
There are 2 output formats:
  i)   Basic - supports OpenSearch™ and RSS 2.0 web syndication. This comes standard with Zoom V6.
  ii)   Extended - supports OpenSearch™ and RSS 2.0 web syndication, plus MasterNode support. This comes standard on MasterNode package.

This document describes the Zoom Search Response XML format that is returned by a web server that provides Zoom V6 search functionality.

## 1.1   Example of applications that can use Zoom Search Response

1.1.1   Example 1: Basic OpenSearch Format

Internet Explorer 7 uses OpenSearch™ to integrate third-party search engines into the browser. Because Zoom Search Response includes these OpenSearch™ elements, you are able to add your Zoom search engine as a search provider in the Toolbar Search Box. More info can be found at Microsoft Internet Explorer 7 page.

Web browsers like Netscape, Safari, FireFox and Internet Explorer 7 has built-in support for RSS feeds. Because Zoom Search Response is an RSS 2.0 format, these web browsers will automatically understand and display basic Zoom Search Response format.

1.1.2   Example 2: Extended OpenSearch Format

Wrensoft's Zoom MasterNode is a distributed search application with parallel searching and caching. It takes search request and divides the work amongst its slave node machines, resulting in faster search time because you have the capability to split your search content into different set of index files. These unique index files of your web content are located on these slaves and they are the workers performing the searches and CPU-intensive tasks. MasterNode consolidates and processes the returned search results from these nodes and present their content in an inter-related and consistent manner on your web browser.
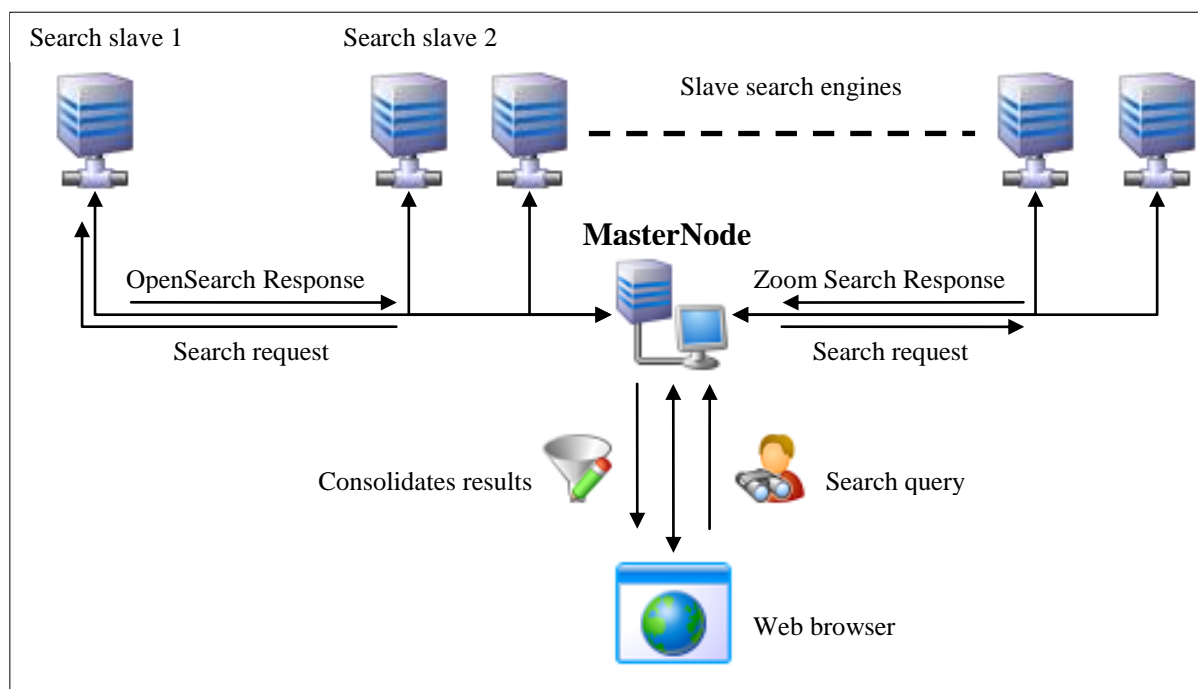

Zoom MasterNode and its Slaves

2.0   ZOOM SEARCH RESPONSE FORMAT OVERVIEW

A search response is a reply to a search query. A Zoom Search Response is a reply to a Zoom Search Query. A search response could be in a number of formats – HTML, XHTML, XML, RSS, etc.

Zoom Search Response is an UTF-8(*) RSS 2.0 based response format that adds Zoom Search Response elements and includes OpenSearch™ RSS Response required elements. OpenSearch™ and RSS 2.0 applications (web feeds, aggregators, etc.) are able to process its contents.
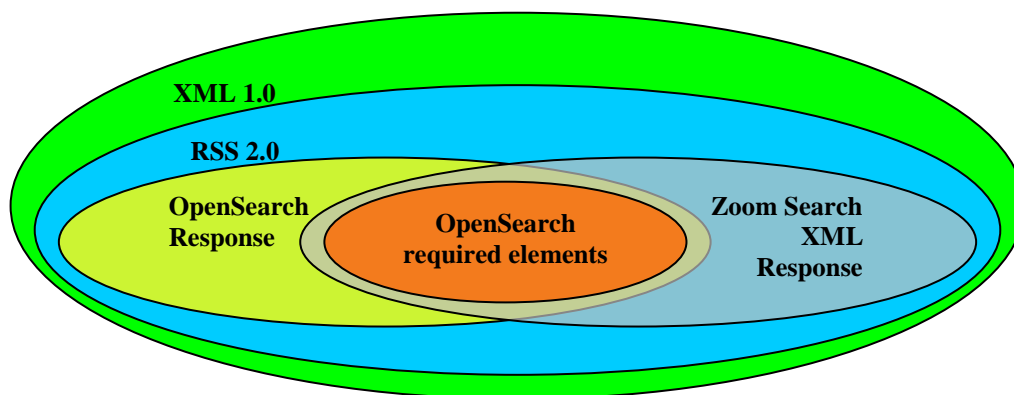
RSS (Really Simple Syndication) is a Web content syndication format. RSS is a dialect of XML. All RSS files must conform to the XML 1.0 specification, as published on the World Wide Web Consortium (W3C) website. Refer to http://blogs.law.harvard.edu/tech/rss for an intro on RSS.

OpenSearch™ is a collection of technologies that allow publishing of search results in a format suitable for syndication. By adding OpenSearch™ Response required elements to RSS, it is able to provide rich search syndication with a minimal amount of overhead. RSS with OpenSearch™ Response elements together form the OpenSearch™ Response. Refer to http://opensearch.a9.com for more on OpenSearch™.

(*) UTF-8:   Unicode is an industry standard designed to allow text and symbols from all languages to be consistently represented and manipulated by computers. Unicode characters can be encoded using any of several schemes termed Unicode Transformation Formats (UTF). UTF-8 (8-bit Unicode Transformation Format) is a variable-length character encoding for Unicode created by Ken Thompson and Rob Pike. It is able to represent any universal character in the Unicode standard, yet is backwards compatible with ASCII. For this reason, it is steadily becoming the preferred encoding for email, web pages, and other places where characters are stored or streamed.

3.0   ZOOM SEARCH RESPONSE SYNTAX

A well-formed tree
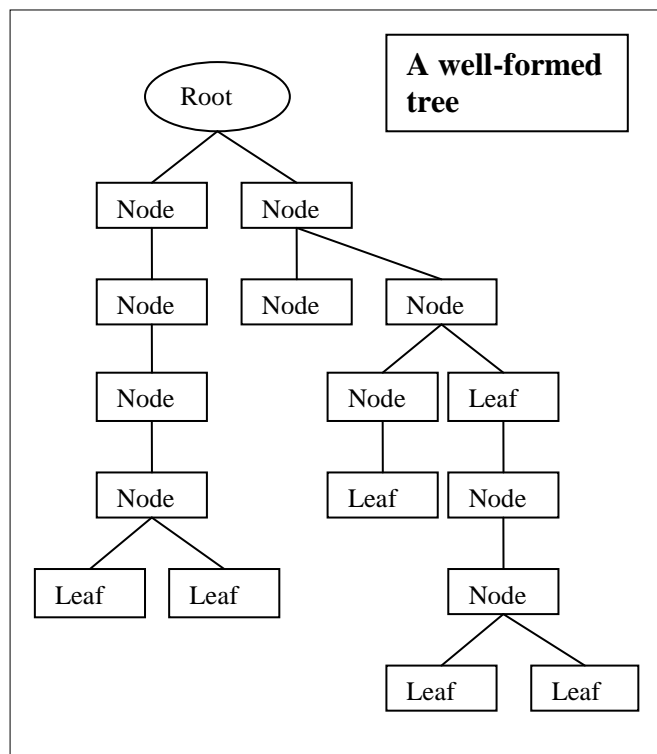
A Zoom Search Response document is "well-formed" (i.e. it obeys all the rules that describes a document in XML 1.0 Recommendation). Those rules essentially say that:

i)   An XML document has to contain a prolog and a single element which forms the root element of the document together with optional comments and processing instructions.

a)   There is exactly one root (or document) element, no part if which appears in the content of any other element.

b)   For all other elements, if the start-tag is in the content of another element, then end-tag is in the content of the same element (nest properly within each other).

ii)   It has to meet all the Well-Formedness Constraints given in the XML 1.0 Recommendation specification.

a)   Parameter entities in the internal subset can only occur where the markup occur.

b)   The name in an element's end-tag must match the start-tag.

c)   An attribute name cannot occur more than once in the same start-tag.

d)   Attribute values cannot contain direct or indirect references to external entities.

e)   The replacement text of an entity referenced to directly or indirectly in an attribute value (other than "<") must not contain a "<".

f)   Characters referred to using character references must be legal characters.

g)   The name given in a entity reference must match that in an entity declaration.

h)   An entity reference must not contain the name of an unparsed entity.

i)   A parsed entity must not contain recursive reference to itself, either directly or indirectly.

j)   Parameter-entity references may only appear in the DTD.

iii)   Each of the parsed entities referenced directly or indirectly within the document is well-formed.
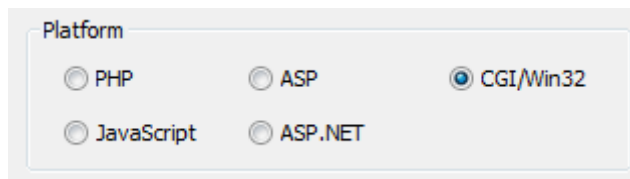
The basic syntax for one element in an XML is: <name attribute="value">content</name>
An empty element is represented by: <name attribute="value" />

Please refer to the World Wide Web Consortium's specification of XML 1.0.

4.0 ZOOM INDEXER CONFIGURATION REQUIREMENTS
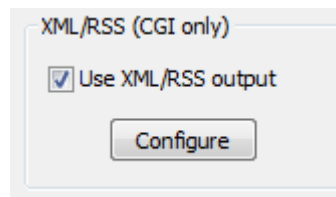
Zoom Search Engine (V5 and later) supports 2 types of query response: HTML and XML/OpenSearch (V4 and below supports only HTML). Zoom Search Engine needs to be configured to output in XML/OpenSearch Response format. Here are the configuration requirements:

1.　　Search platform needs to be configured to CGI. PHP, ASP and JavaScript do not support OpenSearch output.

2.　　Go to the Zoom Indexer Configuration's "Advanced" tab and check "Use XML/RSS.

3.　　Click on the "Configure" button and enter the "Title", "Description", "URL" and "Description file URL" field of the "Configure XML/RSS output" dialog. Click on the "Help" button if you need clarification on what the field means.

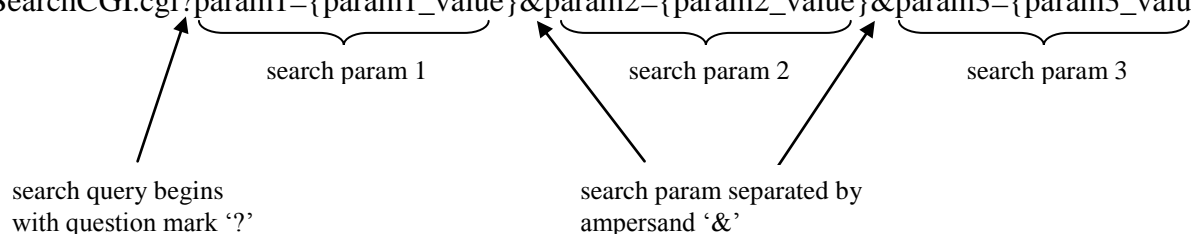You then index the files by clicking on the "Start Indexing" button on the main Zoom Indexer window.

5.0   ZOOM SEARCH QUERY SYNTAX

Zoom Search CGI (Common Gateway Interface) is a high performance search application that processes search request based on its search parameters. These search parameters instructs the CGI what it is searching for and how results should be displayed. Search parameters trail the search application's URL with the question mark character '?' with each search parameter separated by the ampersand character '&' like this:

http://mySearchDomain.com/cgi-bin/mySearchCGI.cgi?param1={param1_value}&param2={param2_value}&param3={param3_value}

search param 1   search param 2   search param 3

search query begins   search param separated by
with question mark '?'   ampersand '&'

where "http://mySearchDomain.com/" is the server portion, "cgi-bin/mySearchCGI.cgi" is the segment portion and "?param1={param1_value}&param2={param2_value}&param3={param3_value}" is the query portion respectively. The curly braces around the parameter value are not needed.

The following are the search parameters that Zoom Search CGI supports and their meanings:

zoom_query
The value for this parameter holds the search terms that the user is searching for. For example, if user wants to find any document that has the word "memory", the search parameter will contain "zoom_query=memory". **Note that this parameter needs to be present for Zoom Search CGI to execute the search.**

zoom_page
The value for this parameter determines the page number that Zoom Search CGI should display the matched items. For example, if user wishes to display searched results on page 2, the search parameter will contain "zoom_page=2". Default value for this parameter is "1".

zoom_per_page
The value for this parameter determines the number of matched items to display for each page. For example, if user wishes to display 15 matched items per page, then the search parameter will contain "zoom_per_page=15". Default value for this parameter is "10".

zoom_and

The value for this parameter determines if the search should "match all search words" or "match any search words". This parameter will only affect search term that contains 2 words or more. If the value of this parameter is "0", then Zoom will return all items that match **any of the search words** in the search term. If the value of this parameter is "1", then Zoom will return only those items that match **all the search words** in the search term.

zoom_xml

The value for this parameter determines the output response of the search results. If this value is set to "1", then Zoom Search CGI will output search results in OpenSearch Response RSS/XML format. If this value is set to "0", then Zoom Search CGI will output search results in HTML format.

zoom_sort

This parameter is only effective if <Provide option to "Sort results by date"> is checked in the "Search Page" tab of the Zoom Indexer Configuration dialog:

☑ Provide option to "Sort results by date"

If you have built your index files with this option, Zoom Search CGI is able to sort matched items either by relevance (i.e. scores) or date/time. To sort results by date time, set this search parameter value to "1". This value defaults to "0", where it will sort by relevance.

zoom_cat[] (i.e. "zoom_cat" followed by the open square bracket "[" and close square bracket "]")

Zoom supports category searching in that user can search for items that match a search term only if the item belongs to a particular category. This parameter is only effective if you have built your index files with categories turned on. Categories can be turned on from the "Categories" tab of the Zoom Indexer Configuration dialog:

☑ Enable categories

For example, if you have configured 3 categories, "News", "Sports" and "Music":

Name	Pattern	Description
News	news	News articles
Sports	sports	Sports articles
Music	music	Music gossip

and user wishes to search all sports articles that has "rugby" in it, then the search parameter will contain "zoom_query=rugby&zoom_cat[]=1". The category value is the zero-based index of the category list. In this example, "zoom_cat[]=0" will search for matches belonging only to "News articles" and "zoom_cat[]=2" will search for matches belonging only to "Music gossip".

Zoom also supports "multiple categories" searching in that it allows user to search for matches that belong to more than 1 category. To enable this feature, make sure to check "Allow searching in multiple categories" in the same preference tab: ☑ Allow searching in multiple categories

Following from the last example, if this option is enabled and a user wishes to search for any sports or news "rugby" articles, then the search parameter will contain "zoom_query=rugby& zoom_cat[]=0&zoom_cat[]=1". Each category is added as a separate search parameter.

The zoom_cat search parameter's value defaults to -1, meaning it will search "All".

Custom Meta Fields
When you have Custom Meta Fields specified in Zoom, additional parameters are introduced to correspond to each custom meta field. These allow the end user to submit a query with each of these fields as criteria.

The parameter names will match the "Meta name" for your Custom Meta Field. For example, if you have a text Custom Meta Field named "AUTHOR", then a query will look something like: "zoom_query=&zoom_per_page=10&AUTHOR=Bob"

If the Custom Meta Field is of type "Multi-select", then the parameter can be repeated to submit multiple values. In this case, the parameter name will be followed with a pair of open and close square brackets (e.g. "AUTHOR[]=Bob"), and a sample query may look something like:
"zoom_query=biography&zoom_per_page=10&AUTHOR[]=Bob&AUTHOR[]=Suze"

6.0   ZOOM SEARCH RESPONSE ELEMENTS

6.1   Example of a Zoom Response Document

```
<?xml version="1.0" encoding="UTF-8"?>
<!—Zoom Search Engine Version 6.0 (1010) STD-->
<rss version="2.0" xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/" xmlns:zoom="http://www.wrensoft/zoom/response/5.0/schema/">
  <channel>
    <title>WrenSoft – Search this website</title>
    <link>http://www.wrensoft.com/search.php</link>
    <description>Search results for "blah"</description>
    <zoom:searchquery>zoom engine</zoom:searchquery>
    <zoom:searchcategory>All</zoom:searchcategory>
    <opensearch:totalResults>122</opensearch:totalResults>
    <opensearch:startIndex>21</opensearch:startIndex>
    <opensearch:itemsPerPage>10</opensearch:itemsPerPage>
    <opensearch:link rel="search"
href="http://www.wrensoft.com/opensearchdescription.xml" type="application/opensearchdescription+xml"/>
    <zoom:spelling>Bob</zoom:spelling>
    <zoom:spelling>bobby</zoom:spelling>
    <zoom:spelling>robert</zoom:spelling>
    <item>
       <zoom:recommended value="yes" />
       <title>Zoom Product Page</title>
       <link>http://www.wrensoft.com/zoom/index.html</link>
       <description>Information on Zoom</description>
       <pubDate>Sun, 19 May 2002 15:21:36 GMT</pubDate>
    </item>
    <item>
       <title>Users Guide for &lt;Search Engine&gt; 4.1 (PDF)</title>
       <link>http://www.wrensoft.com/ftp/zoom.pdf</link>
       <description> A comprehensive PDF Users Guide to using the Zoom Search Engine 4.1 package</description>
       <category>Documentation</category>
       <pubDate>Sun, 19 May 2002 15:21:36 GMT</pubDate>
       <zoom:context>...WRENSOFT Zoom Search EngineTM Custom website search engine Version 4.2 Users Guide WRENSOFT &#xA9; Copyright WrenSoftTM
2000-2005 http :// www.wrensoft.com 17 / Aug / 2005 Page 1 of 58 WRENSOFT Contents CHAPTER 1. OVERVIEW. 4 1.1....</zoom:context>
```

```
        <zoom:score>1313</zoom:score>
        <zoom:termsMatched>1</zoom:termsMatched>
        <zoom:fileSize>1044480</zoom:fileSize>
        <zoom:imageURL>http://www.mysite.com/home/images/icons/icon_dog.png</zoomsearch:imageURL>
      </item>
      <!-- 9 additional <item>s elements appear here -->
    </channel>
</rss>
```

6.2   Elements

			Y: Yes	N: No	O: Optional

Element	Description	Attributes	Required elements/attributes?		
			RSS 2.0	**Zoom**	**OpenSearch™**
?xml	XML declaration, specifies the version of XML and it's encoding	version – must equal "1.0"	Y	Y	Y
		encoding – must equal "UTF-8"	Y	Y	Y
rss	RSS identification tag, specifies the version of RSS	version – must equal "2.0"	Y	Y	Y
		xmlns:opensearch – must equal "http://a9.com/-/spec/opensearch/1.1/"	N	O	Y
		xmlns:zoom – must equal "http://www.wrensoft/zoom/response/5.0/schema/"	N	Y	N
channel	Information about the search engine		Y	Y	Y
Sub-elements of <channel>					
title	Name of the search engine		Y	Y	Y
link	URL to the HTML website corresponding to the search engine		Y	Y	Y
description	Phrase or sentence describing the search engine		Y	Y	Y
lastBuildDate	Last time search engine was indexed		O	N	O
opensearch:totalResults	Maximum number of results available for these search terms.		N	Y	Y
opensearch:startIndex	Index of the first item returned in the result		N	O	Y

Element	Description	Attributes	Required elements/attributes?		
			RSS 2.0	**Zoom**	**OpenSearch™**
`opensearch:itemsPerPage`	Maximum number of items that can appear in one page of results		N	O	Y
`opensearch:link`	Reference back to the OpenSearch™ Description file. This is to enable Autodiscovery of your OpenSearch™.	rel – must equal "search"	N	O	O
		href – URL to the search engine's OpenSearch description file	N	O	O
		type – mime type of the description file, must equal "application/opensearchdescription+xml"	N	O	O
`item`	Information about a search item. A channel may contain any number of <item>s.		Y	Y	Y
`language`	The language channel is written in		O	O	O
`zoom:categorySummary`	Category summary. A channel may contain a variable number of <zoom:categorySummary> elements. The maximum number corresponds with the number of categories specified.		N	O	O
`zoom:spelling`	Spelling suggestions. A channel may contain a configurable number of <zoom:spelling> elements.		N	O	O
`zoom:recommended`	Recommended link. A channel may contain a configurable number of <zoom:recommended>		N	O	O
Sub-elements of <item>					
`title`	Title of the item		One of <title> or <description> must be present	Y	One of <title> or <description> must be present
`description`	Description of the item		One of <title> or	O	One of <title> or

Element	Description	Attributes	Required elements/attributes?		
			RSS 2.0	Zoom	OpenSearch™
			\<description\> must be present		\<description\> must be present
link	URL of the item		O	Y	O
zoom:context	Context string for search result		N	O	N
zoom:imageURL	URL of thumbnail image		N	O	N
zoom:score	Score (weight) of URL		N	O	N
zoom:termsMatched	Number of terms matched on page		N	O	N
zoom:fileSize	Size of page (in bytes)		N	O	N
pubDate	Last modification date of URL		O	O	O
category	Category it belongs to		O	O	O
zoom:{META_NAME}	Custom Meta Field for this page. An \<item\> may contain any number of custom meta elements.		N	O	O
Sub-elements of \<zoom:recommended\>					
title	Title of the recommended link		N	Y	O
description	Description of the recommended link		N	O	O
link	URL of the recommended link		N	Y	O
zoom:imageURL	URL of thumbnail image		N	O	N

6.3   The XML Declaration

```
<?xml version="1.0" encoding="UTF-8"?>
```

Identifies the file as an XML file conforming to XML 1.0 standards and specifying UTF-8 as the encoding for the character sets we are using in the format.

6.4   The XML Body

6.4.1   <rss> element

This is the root element of the Zoom Search Response format. There can only be one <rss> element and it must be the first element after the prolog.

```
<rss version="2.0" xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/">
```

The <rss> element identifies this format as belonging to the XML family of application called RSS.

Its version attribute:

```
version="2.0"
```

says that it conforms to RSS 2.0 specification (RSS 2.0 is a dialect of XML 1.0)

Lastly, the xmlns:opensearch attribute:

```
xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/"
```

associates all elements starting with "opensearch:" to have the namespace enclosed by the double quotes.

6.4.2   <channel> element

This element contains metadata about your search engine, its title, description, URL, etc. There can only be one <channel> element and it must be the first element after the <rss> element.

6.4.2.1 Required <channel> elements

One of each element here must be present as a child of the <channel> element.

<title> element
When this element is a child of the <channel> element, it is the title of the search engine.

<link> element
When this element is a child of the <channel> element, it is the URL of the search engine.

<description> element
When this element is a child of the <channel> element, it is the description of the search engine.

<lastBuildDate> element
The last time the search index was built.

<opensearch:totalResults> element
Maximum number of search results returned from the search terms.

<item> element
This element contains metadata about each search result item, its title, description, context, file size, score, URL, etc. There can be multiple <item> elements nested within the <channel> element.

6.4.2.2 Optional <channel> elements

These elements are optional as a child of the <channel> element.

<language> element
This is the language the channel is written in. For example, English (Australia) is "en-au" and English United States is "en-us".

Default: "en=us"

<opensearch:startIndex> element

Index of first item returned in the search results. To be compatible with OpenSearch™ aggregators, this needs to be present.

Default: 1

<opensearch:itemsPerPage> element
Maximum number of items that can appear on one page. To be compatible with OpenSearch™ aggregators, this needs to be present.

Default: 10

<opensearch:link> element
Reference back to the OpenSearch™ Description file. This will enable "Autodiscovery" of your OpenSearch™.

Attributes of <opensearch:link>:
rel: Must equal "search".
href: URL to the OpenSearch™ Description file.
type: Must equal "application/opensearchdescription+xml".

6.4.2.3 Required <item> elements
One of each element here must be present as a child of the <item> element.

<title> element
When this element is a child of the <item> element, it is the title of the search item.

<link> element
When this element is a child of the <item> element, it is the URL of the search item.

<description> element
When this element is a child of the <item> element, it is the description of the search item.

<pubDate> element
 Published date of the URL. It could also be the last modified date of the URL.

6.4.2.4 Optional <item> elements

These elements are optional as a child of the <item> element.

<category> element

Category the search item belongs to.

<zoom:context> element

Context string of the search item.

<zoom:imageURL> element

URL of thumbnail to be displayed with the search item.

<zoom:score> element

The score of the search item.

<zoom:termsMatched> element

Number of terms matching the search terms in this item.

<zoom:fileSize> element

File size (in bytes) of the URL of this item.

<zoom:{META_NAME}> element

This element will only appear if *Custom Meta Fields* were enabled and specified at the time of indexing.
The {META_NAME} part of the element name will match the "Meta Name" specified in the Indexer for the Custom Meta Field.
Each <item> may contain zero or more <zoom:{META_NAME}> tags.
The value contained with this element is the value of the meta field indexed for this page.

<enclosure> element

This element contains information about the type of media that is enclosed with this item.

Attributes of <enclosure>:
url: The URL of the media.
length: File size of media (in bytes).
type: Mime type of media enclosed.

7.0   COMPATIBILITY WITH OPENSEARCH™

Zoom Search Response (when implemented correctly) is a well-formed RSS 2.0 format. The specification for Zoom Search Response format is extensible in that you are free to include application-specific elements (as long as the element names are unique in its own namespace) in your response. For OpenSearch™ aggregators to be able to process your Zoom Search Response, you must:

1.   Specify OpenSearch namespace in the attribute of the root <rss> element.
```
<rss version="2.0" xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/">
```

2.   Include the following elements as children of the <channel> element:
   i)   <zoom:startIndex>
   ii)   <zoom:itemsPerPage>
   iii)   <zoom:link>
            Attributes:
            a)   rel: must equal "search"
            b)   href: URL to the OpenSearch Description file
            c)   type: must equal "application/opensearchdescription+xml"

The <link> element is optional but recommended by OpenSearch™. It allows for "autodiscovery" of your OpenSearch™ by other OpenSearch™ aggregators.

Example of a simple OpenSearch Description file:
```
<?xml version="1.0" encoding="UTF-8"?>
    <OpenSearchDescription xmlns="http://a9.com/-/spec/opensearch/1.1/">
    <ShortName>Web Search</ShortName>
    <Description>Use Example.com to search the Web.</Description>
    <Tags>example web</Tags>
    <Contact>admin@example.com</Contact>
    <Url type="application/rss+xml" template="http://example.com/?q={searchTerms}&amp;pw={startPage?}&amp;format=rss"/>
</OpenSearchDescription>
```

For more information on OpenSearch Description file, refer to http://opensearch.a9.com/spec/1.1/description/.

8.0 CONTACT DETAILS

PassMark Software Pty Ltd
T/A Wrensoft
Suite 202, Level 2
35 Buckingham Street
Surry Hills, 2010
Sydney, Australia
ACN 099 321 392
Phone + 61 2 9690 0444
Fax + 61 2 9690 0445
http://www.wrensoft.com/

INDEX OF ELEMENTS