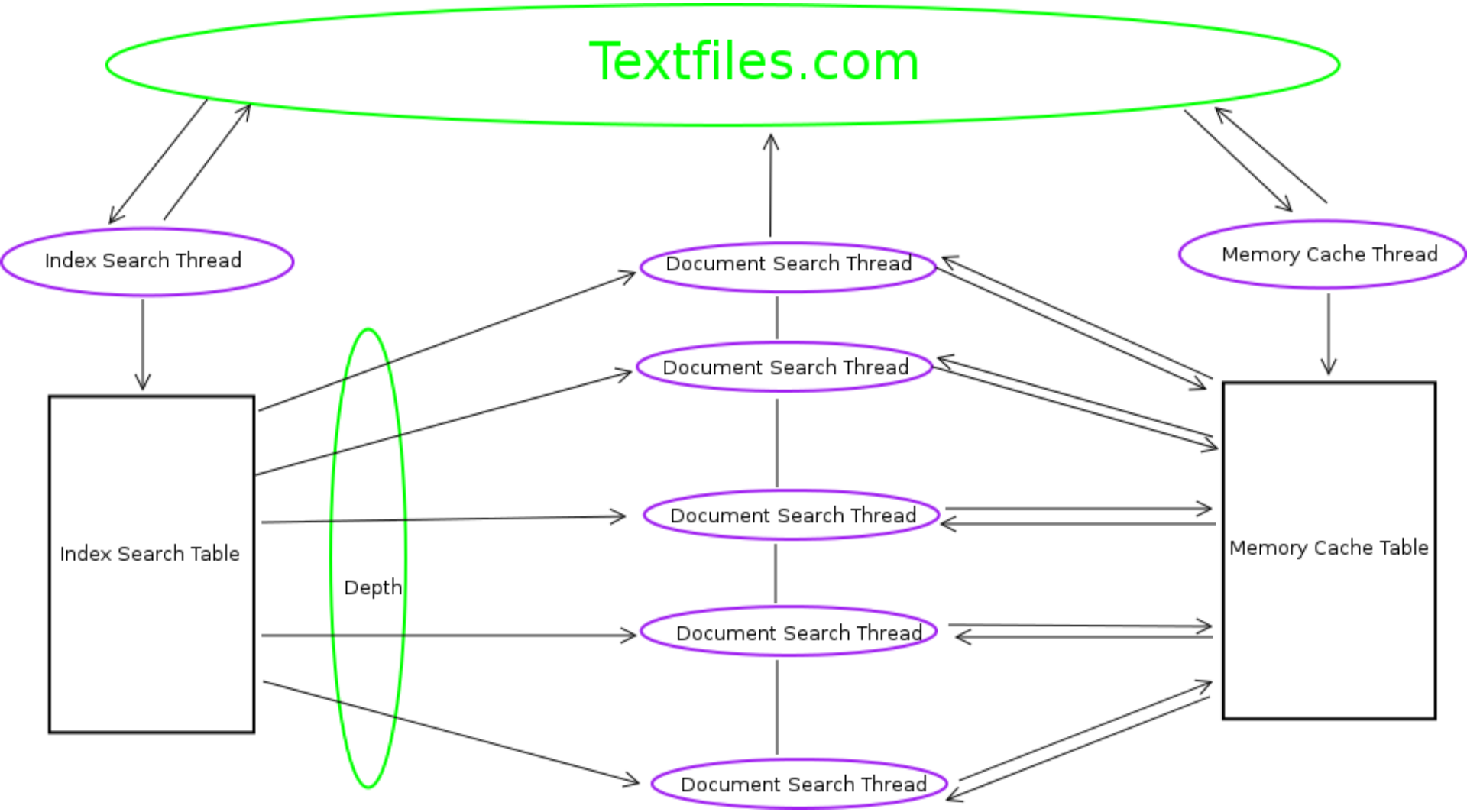
Aragog
A multithreaded search for Textfiles.com

Team : 3 musketeers
Parth Mehta
Yisong Wu
Priyank Desai

Goals

- Index Search
  - Search the document genres inside Textfiles.com for a particular keyword
- Document Search
  - Search the web pages and text files in Textfiles.com for a particular keyword

Top Level Design



Indexr Thread

Start from the page http://textfiles.com/directory.html

For each URL in found in the directory page store them in the "Index Search Vector" and "Index Search Table"

Load the page at the URL and find links in the page which are not text files and add them in the "Index Search Vector" and "Index Search Table"

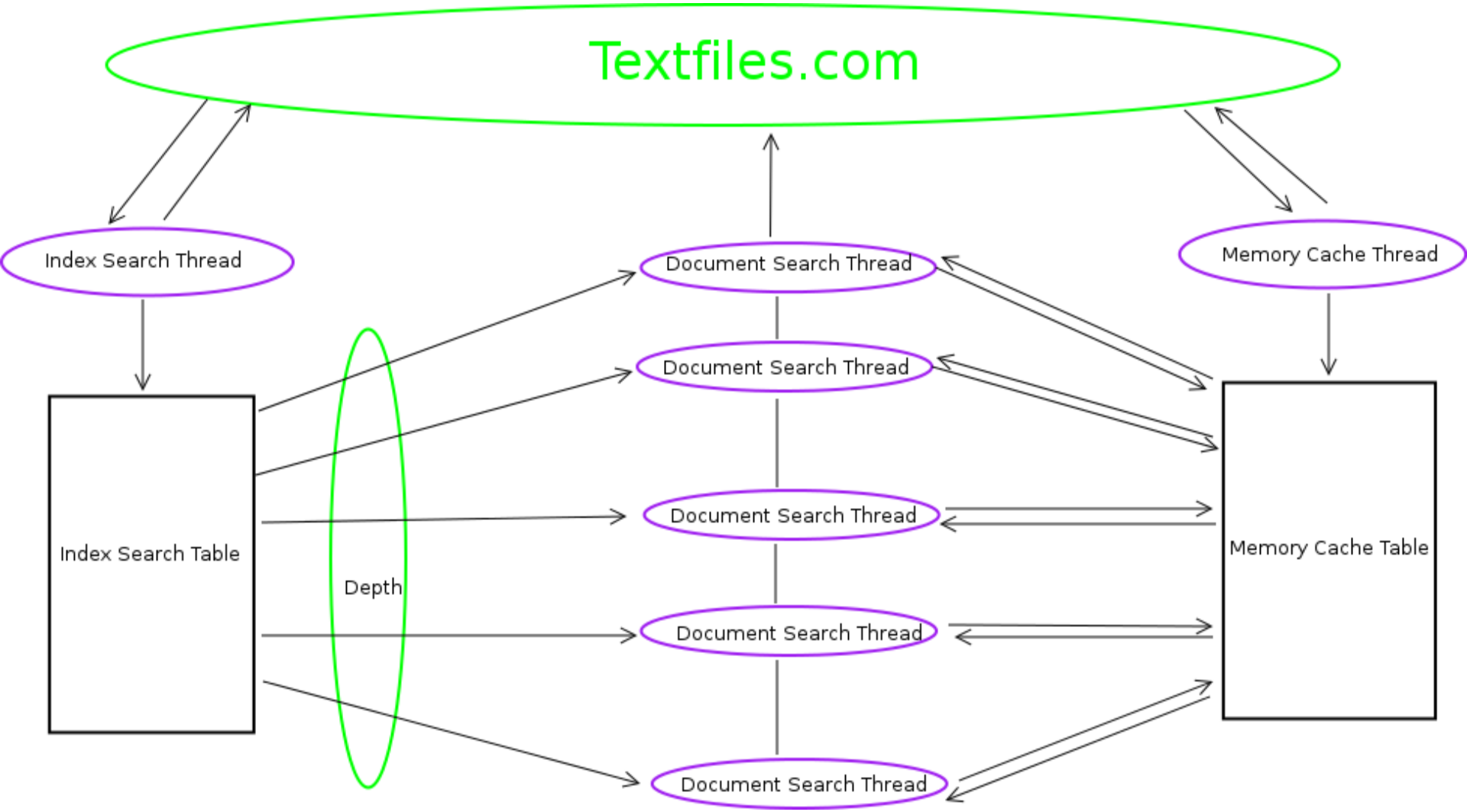
Index Search Vector

URLs
http://textfiles.com/100
http://textfiles.com/anarchy
http://textfiles.com/anarchy/CARDING
http://textfiles.com/apple
http://textfiles.com/fun
http://textfiles.com/hacking

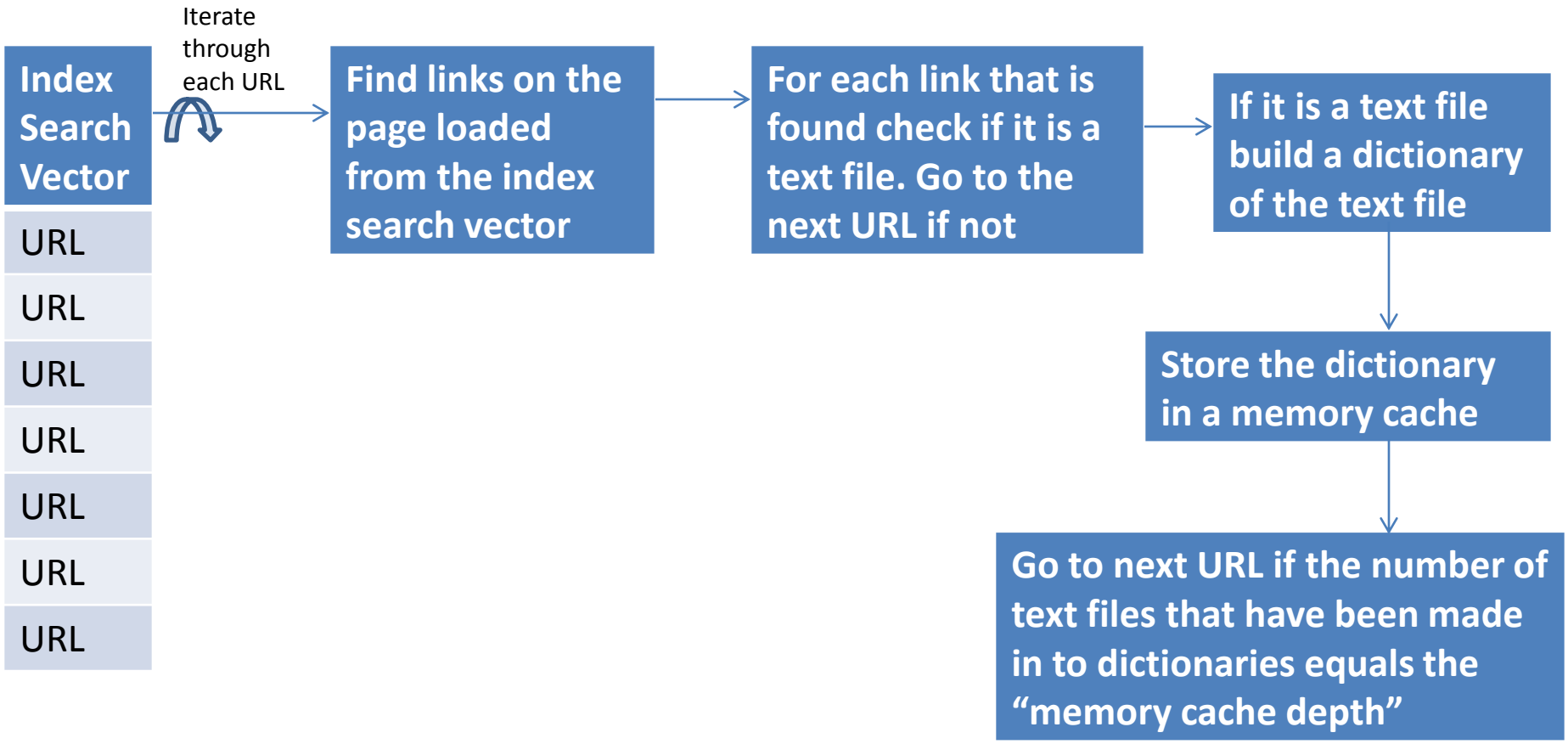
Index Search Table

Keyword (key)	URL (Value)
100	http://textfiles.com/100
Anarchy	http://textfiles.com/anarchy
CARDING	http://textfiles.com/anarchy/CARDING
Apple II	http://textfiles.com/apple

Top Level Design



Memory Cache Thread



Dictionary and Memory Cache

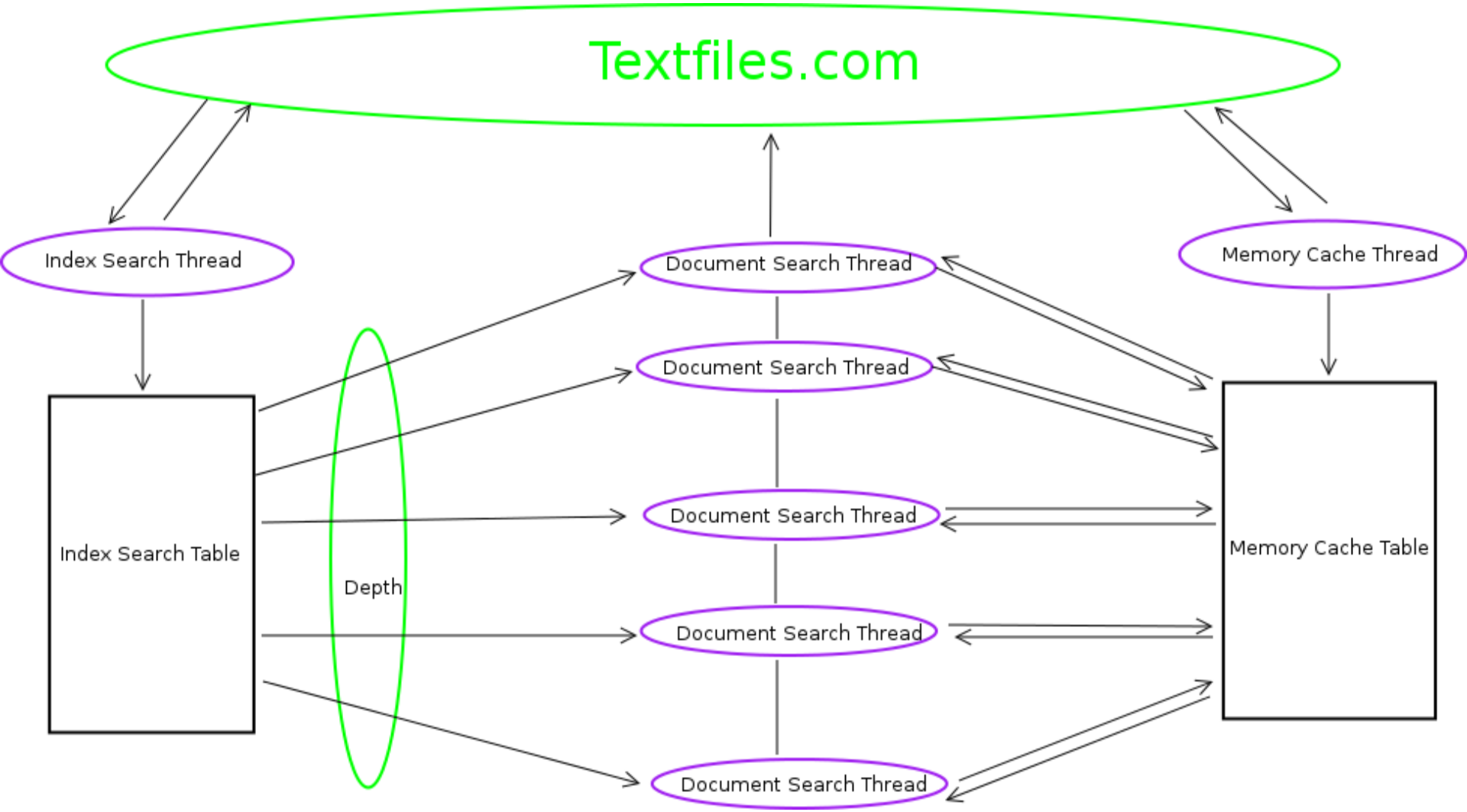
Dictionary

WORD(key)	Value
Computer	1
Science	1
For	1
Engineers	1

Memory Cache

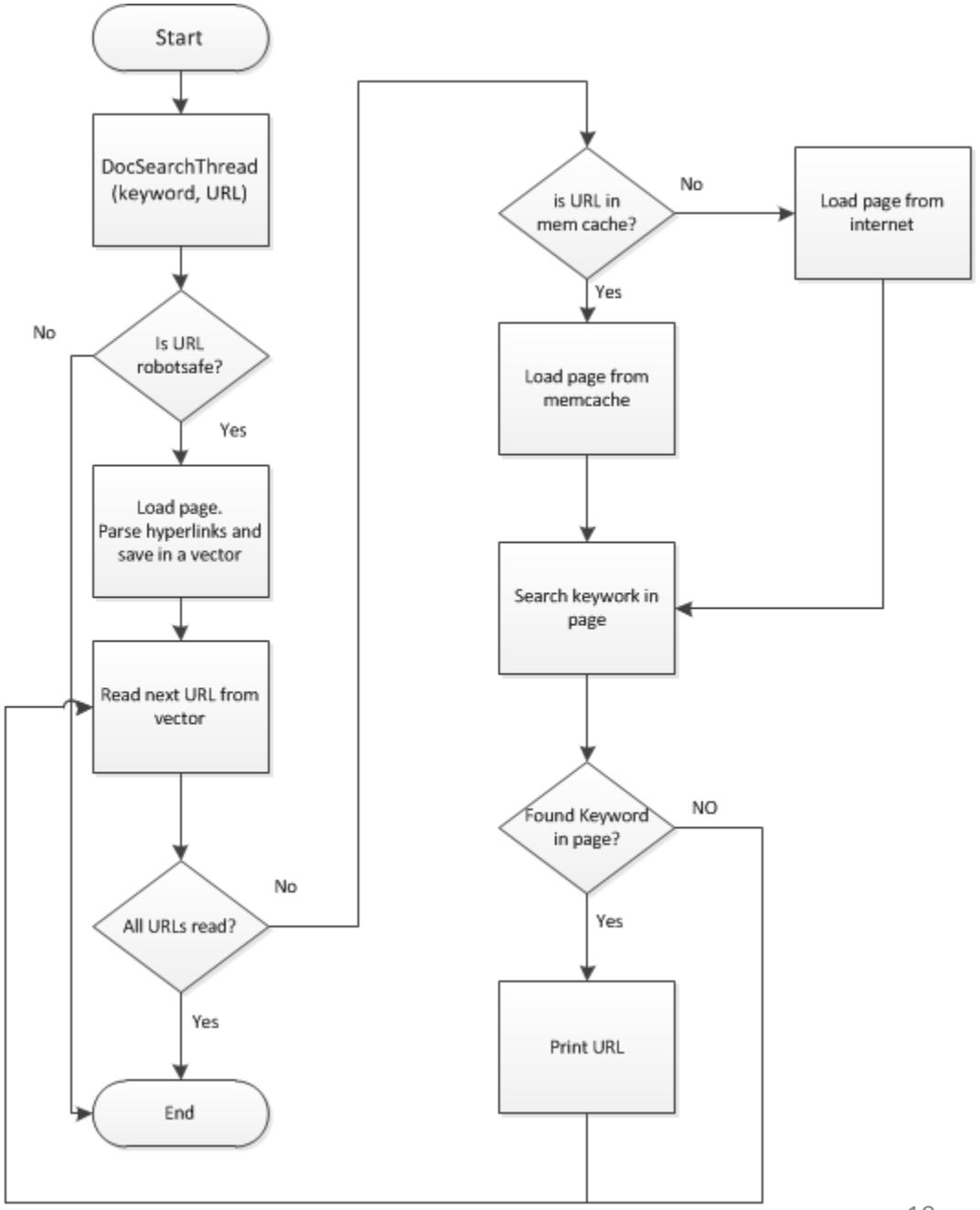
URL(key)	Dictionary(Value)
http://textfiles.com/anarchy/aoa4.txt	Dictionary for file aoa4.txt
http://textfiles.com/anarchy/badmind2.txt	Dictionary for file badmind2.txt
http://textfiles.com/food/all_grai	Dictionary for all_grai
http://textfiles.com/food/aphrodis.txt	Dictionary for aphrodis.txt
http://textfiles.com/food/beer-g	Dictionary for beer-g

Top Level Design



Document Search Thread

Loads the URL from the Index search vector and searches through all text files in the page pointed to by the URL.



Crawler Engine class

CrawlerEngine Class	
Class Variable	**Data Type**
newURLs	Vector<URL>
knownURLs	Hashtable<URL, Integer>
death	Int (for killing threads)
Method	**Description**
RobotSafe (URL url)	Check is a URL is robot safe
AddNewUrl (URL oldURL, String newUrlString)	Add a URL to “newURLs” and “knownURLs”
LoadPage (URL url)	Load a page at the give url
ProcessPageForTextFiles (URL url, String page)	Used by Doc search thread to extract urls on the page and store in the newURLs vector
processTextFiles (String searchString, JTextArea textArea, int depth)	Used by doc search to search through text files in the newURLs vector. This method uses the memCacheCheck method to find a hit in the memory cache
parseDirectoryPage (URL initUrl, String directoryPage, Hashtable<String, URL> indexSearchTable, Vector<URL> indexSearchVector)	Used by the indexr thread to parse the textfilex.com/directory page.
ProcessPage2 (URL url, String page, Hashtable<String, URL> indexSearchTable, Vector<URL> indexSearchVector)	A helper method for parseDirectory page to extrace urls from a page which are not text files
procssPageforMemCache (URL nextUrl, String page,Queue<URL> memCacheQ)	Used by memcache thread to process urls and build dictioonaries of text files and store in the memory cache

Crawler GUI

CrawlerGui	
Class Variable	**Data Type**
indexSearchTable	Hashtable<String, URL>
indexSearchVector	Vector <URL>
memCache	Hashtable <URL,Hashtable<String, Integer>>
Classes	**Description**
IndexrThread	Creates the indexSearchTable and indexSearchVector by using the processDirectoryPage method of crawler engine class
DocSearchThread	Searches the keyword given in the page at the url in the index search vector using the processPageForTextFiles and processTextFiles methods.
MemCacheThread	This method uses the processPageForMemCache
Method	**Description**
MemCacheCheck(URL url, String keyword)	This method checks the url if it is present in the memory in the cache

How search works

- Index Search:
  - Looks up the keyword in the Index Search Table created by Indexr thread
- Document Search:
  - Starts “depth” number of DocSearchThreads where each thread looks through all the text files as well as the page pointed to by the URL extracted from Index Search Vector

Complexity Analysis

Indexr thread:

- O (n) in time to build an index from TextFiles.com where n is the number of links in textfiles.com
- O (n) in memory to build an Index Search Table and Index Search Vector

DocSearch thread:

- O(1) to look up URL from Index Search Vector
- O(n) in time to look through all text files in a URL
- O(1) in memory to look for keyword in a file

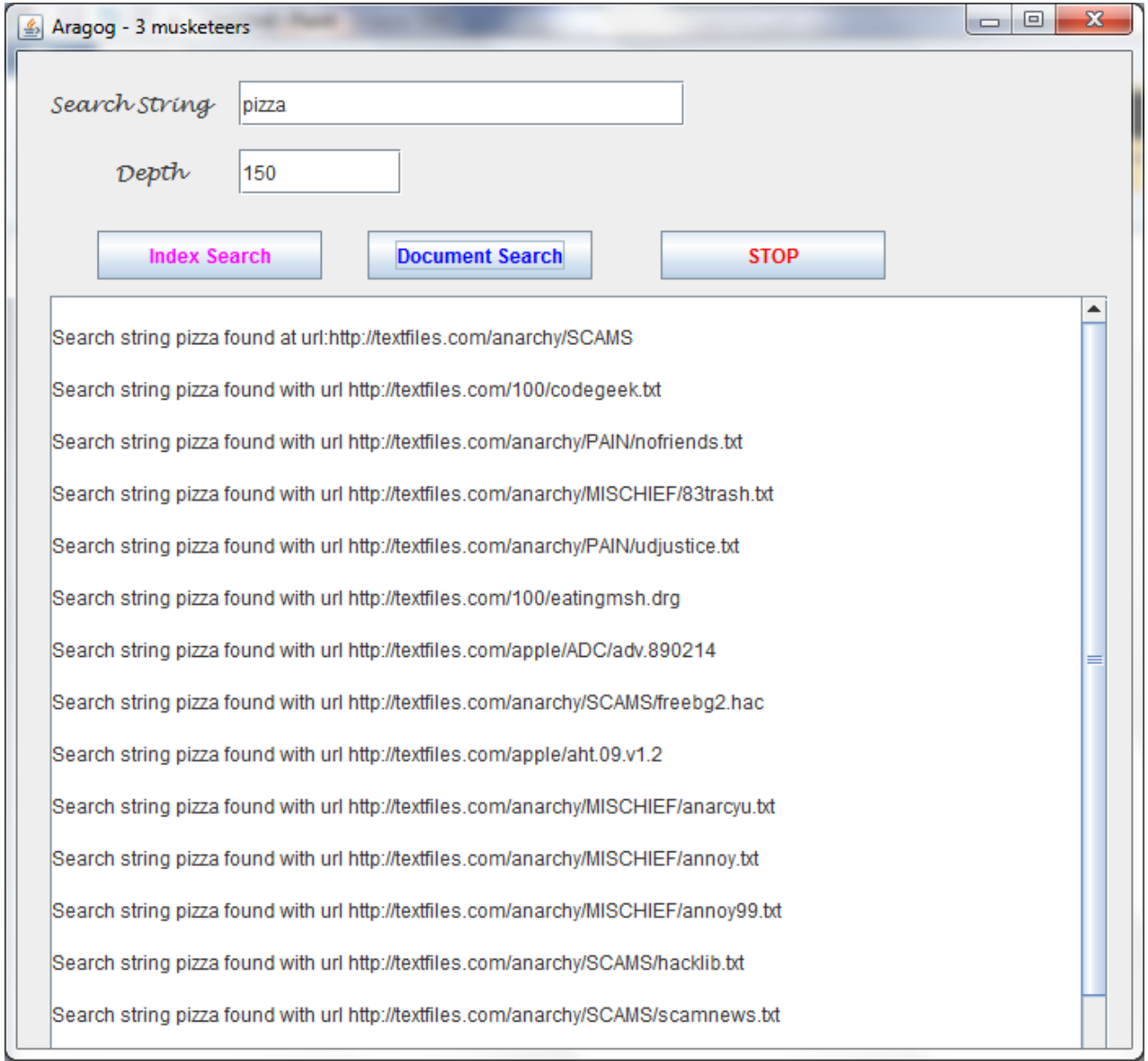
memCache thread:

- O(n) in time to go through every URL in the Index Search Vector
- O(n) in memory to save files from each category on directory page

Searching page for keyword:

- O (1) in time

Aragog


Aragog - 3 musketeers
Search String
pizza
Depth
150
Index Search
Document Search
STOP
Search string pizza found at url:http://textfiles.com/anarchy/SCAMS
Search string pizza found with url http://textfiles.com/100/codegeek.txt
Search string pizza found with url http://textfiles.com/anarchy/PAIN/nofriends.txt
Search string pizza found with url http://textfiles.com/anarchy/MISCHIEF/83trash.txt
Search string pizza found with url http://textfiles.com/anarchy/PAIN/udjustice.txt
Search string pizza found with url http://textfiles.com/100/eatingmsh.drg
Search string pizza found with url http://textfiles.com/apple/ADC/adv.890214
Search string pizza found with url http://textfiles.com/anarchy/SCAMS/freebg2.hac
Search string pizza found with url http://textfiles.com/apple/aht.09.v1.2
Search string pizza found with url http://textfiles.com/anarchy/MISCHIEF/anarcyu.txt
Search string pizza found with url http://textfiles.com/anarchy/MISCHIEF/annoy.txt
Search string pizza found with url http://textfiles.com/anarchy/MISCHIEF/annoy99.txt
Search string pizza found with url http://textfiles.com/anarchy/SCAMS/hacklib.txt
Search string pizza found with url http://textfiles.com/anarchy/SCAMS/scamnews.txt

How to use the tool:

Index Search:

1. Enter keyword to search in ‘Search String’ textbox.
2. Click on **Index Search button**

Document Search:

1. Enter keyword to search in ‘Search String’ textbox.
2. Enter depth(optional)
3. Click on **Document Search button**
4. Click on **Stop** button to stop the search.