

ALEXA-Seq
(www.AlexaPlatform.org)

Linux Installation Manual (v.1.17)

15 Feb 2012

Table of Contents

Introduction.....	3
Preamble	3
Prerequisites	3
Notes	3
A. Performing the ALEXA-Seq analysis	4
0. Installing Alexa-seq code base	4
1. Installing R	4
2. Installing R packages and BioConductor R libraries	5
3. Installing BLAST, BWA and mdust	5
BLAST	5
BWA	5
MDUST	5
4. Installing Cairo R library (requires cairo/pixman)	6
5. Installing BerkeleyDB	7
6. Installing mysql server (requires root privileges)	8
7. Updating .bashrc file	10
8. Creating and editing config files	10
B. Creating a new ALEXA-Seq annotation database	11
C. Set up the ALEXA-Seq data viewer	12
1. Installing Apache web server (requires root privileges)	12
2. Installing xapian-omega (requires root privileges)	15

Introduction

Preamble

This manual is meant to serve as a walkthrough for the installation of all software libraries required to operate the Alexa-seq pipeline. It was created for the CentOS 5.5 linux distribution. But, it should serve as a good general guideline for other distributions. Hopefully with this manual you will be able to get the Alexa-Seq pipeline working. However, if you have issues/problems not discussed in this manual please contact us through our website (www.alexaplatform.org). From this website you can also download a VMware virtual machine pre-configured with all dependencies.

Prerequisites

For many of the installation steps below, you will require C/C++ compilers and Perl. You will also need CVS. These are commonly included with linux distributions (e.g., Perl is installed by default in CentOS) and unless you are working on a very fresh linux installation they are likely already available. These should be installed in their preferred system-wide, default locations whereas many of the packages we will be installing can safely be installed in (for example) your own home directory. Therefore, please consult your sysadmin if you do not have Perl, CVS or C/C++ compilers available.

If you have root, you can install them yourself from packages. For example:

```
yum install gcc
yum install gcc-c++
yum install cvs
```

Notes

With all the following installations, it is important to consider whether the code will be running in a 32-bit or 64-bit environment. For most parts of the pipeline we assume 64-bit because we want to run these jobs on a modern, fast, large memory box. But consider carefully whether this is the case for your cluster nodes, webserver or MySQL server. For example if the cluster nodes are 32 bit, install blast on a 32-bit computer or it will not run on the nodes. The best approach is to always install/compile on the actual server where the code will run. This is particularly important for items such as xapian-omega which requires specific libraries for the code to run.

The following installation assumes that most things are being installed into a user's home directory (i.e., /home/your_user_name/). In the examples provided, this user is called 'alex-seq'. You will need to change all file paths accordingly in the following instructions.

The examples use 'vi' whenever a file needs to be edited. Any other text editor will work just as well.

A. Performing the ALEXA-Seq analysis

0. Installing Alexa-seq code base

Create folders for Alexa-seq software and associated files:

```
mkdir /home/alexa-seq/ALEXA
mkdir /home/alexa-seq/ALEXA/sequence_databases/
mkdir /home/alexa-seq/ALEXA/analysis/
mkdir /home/alexa-seq/ALEXA/config_files/
mkdir /home/alexa-seq/ALEXA/perl_storables/
mkdir /home/alexa-seq/ALEXA/www/
mkdir /home/alexa-seq/ALEXA/commands/
```

Explanation of folders:

ALEXA - All ALEXA related files

ALEXA/sequence_databases - ALEXA sequence databases (see below)

ALEXA/analysis - Analysis files for each project processed by ALEXA

ALEXA/config_files - Config files for the pipeline and each project to be processed

ALEXA/perl_storables - Temporary storage for Perl Storable files

ALEXA/www - Keep web files here if you can't write directly to webserver

ALEXA/commands - Store commands files for each project

Install Alexa-seq code base:

This will create an alexa_seq/ folder with all ALEXA code

```
cd /home/alexa-seq/ALEXA/
wget http://www.alexaplatform.org/alexa_seq/source_code/ALEXA_Seq_v.1.17.tar.gz
tar -zxvf ALEXA_Seq_v.1.17.tar.gz
```

1. Installing R

Go to CRAN, choose closest mirror, and download the latest source file. In this case I copied the link location and used 'wget' to download the file.

Remember to log in to the appropriate computer if not your personal workstation where R will actually be run before installing. This warning applies to all subsequent steps as well...

For a non-default-location R installation (e.g., in your home directory). Create folder for R, move/download source code to there, unzip, unpack and install as follows:

```
mkdir /home/alexa-seq/bin/R64
cd /home/alexa-seq/bin/R64
wget http://cran.cnr.berkeley.edu/src/base/R-2/R-2.10.1.tar.gz
tar -zxvf R-2.10.1.tar.gz
cd R-2.10.1/
./configure --prefix=/home/alexa-seq/bin/R64/R-2.10.1/
make
make install
```

2. Installing R packages and BioConductor R libraries

```
/home/alexa-seq/bin/R64/R-2.10.1/bin/R
```

At R prompt:

```
install.packages("RColorBrewer")
install.packages("gplots")
source("http://bioconductor.org/biocLite.R")
biocLite()
```

3. Installing BLAST, BWA and mdust

Note: We recommend that you use the following versions of BLAST and BWA to ensure compatibility with the indexed databases and commands provided with ALEXA package. If you wish to use the latest version, you might need to re-index the sequence databases (not necessarily).

BLAST

```
mkdir /home/alexa-seq/bin/BLAST64/
cd /home/alexa-seq/bin/BLAST64/
wget ftp://ftp.ncbi.nlm.nih.gov/blast/executables/release/2.2.18/blast-
2.2.18-x64-linux.tar.gz
tar zxvf blast-2.2.18-x64-linux.tar.gz
```

BWA

```
mkdir /home/alexa-seq/bin/BWA/
cd /home/alexa-seq/bin/BWA/
wget http://sourceforge.net/projects/bio-bwa/files/bwa-
0.5.8a.tar.bz2/download
bunzip2 bwa-0.5.8a.tar.bz2
tar xvf bwa-0.5.8a.tar
cd bwa-0.5.8a
make
```

MDUST

Copy and install the mdust package provided in the ALEXA package:

```
cp /home/alexa-seq/ALEXA/alexa_seq/external_tools/mdust.tar.gz /home/alexa-
seq/bin/
cd /home/alexa-seq/bin
tar -zxvf mdust.tar.gz
cd mdust
make
```

4. Installing Cairo R library (requires cairo/pixman)

4.1 Locate source files from cairo website <http://cairographics.org>:

For example,

<http://cairographics.org/releases/LATEST-cairo-1.8.10>

<http://cairographics.org/releases/LATEST-pixman-0.17.14>

4.2 Make target directory and unpack both to here:

```
mkdir /home/alexa-seq/lib/  
mkdir /home/alexa-seq/lib/Cairo_x64/  
cd /home/alexa-seq/lib/Cairo_x64/  
wget http://cairographics.org/releases/LATEST-cairo-1.8.10  
wget http://cairographics.org/releases/LATEST-pixman-0.17.14  
tar -zxvf LATEST-pixman-0.17.14  
tar -zxvf LATEST-cairo-1.8.10
```

4.3 Compile pixman

```
cd /home/alexa-seq/lib/Cairo_x64/pixman-0.17.14/  
./configure --prefix=/home/alexa-seq/lib/Cairo_x64/  
make  
make install
```

4.4 Compile Cairo

```
cd /home/alexa-seq/lib/Cairo_x64/cairo-1.8.10/  
export PKG_CONFIG_PATH=/home/alexa-  
seq/lib/Cairo_x64/lib/pkgconfig:$PKG_CONFIG_PATH  
./configure --prefix=/home/alexa-seq/lib/Cairo_x64/  
make  
make install
```

4.5 Set libraries for Cairo

The following will need to be added to your .bashrc file to make them persistent

```
export PKG_CONFIG_PATH=/home/alexa-seq/lib/Cairo_x64/lib/pkgconfig:$PKG_CONFIG_PATH  
export LD_LIBRARY_PATH=/home/alexa-seq/lib/Cairo_x64/lib:$LD_LIBRARY_PATH
```

4.6 Start R and install Cairo packages

```
/home/alexa-seq/bin/R64/R-2.10.1/bin/R
```

At R prompt:

```
install.packages("Cairo")
```

5. Installing BerkeleyDB

5.1 Install Oracle Berkeley DB (free registration required)

Download package from:

<http://www.oracle.com/technology/software/products/berkeley-db/index.html>

Install as follows:

```
mkdir /home/alexa-seq/lib/BerkeleyDB/
cp /home/alexa-seq/Desktop/db-4.8.26.tar.gz /home/alexa-seq/lib/BerkeleyDB/
cd /home/alexa-seq/lib/BerkeleyDB/
tar -zxvf db-4.8.26.tar.gz
cd /home/alexa-seq/lib/BerkeleyDB/db-4.8.26/build_unix
mkdir /home/alexa-seq/lib/BerkeleyDB_x64
../dist/configure --prefix=/home/alexa-seq/lib/BerkeleyDB_x64/
make
make install

#Fix LD path (probably not necessary if you did a global install)
su
cd /usr/lib
ln -s /home/alexa-seq/lib/BerkeleyDB_x64/lib/libdb-4.8.so
echo /home/alexa-seq/lib/BerkeleyDB_x64/lib/ >> /etc/ld.so.conf
/sbin/ldconfig
```

5.2 Install BerkeleyDB Perl Module

Download from:

<http://search.cpan.org/dist/BerkeleyDB/BerkeleyDB.pod.P>

Create folder, move and unpack package

```
mkdir /home/alexa-seq/perl/
cp /home/alexa-seq/Desktop/BerkeleyDB-0.42.tar.gz /home/alexa-seq/perl/
cd /home/alexa-seq/perl/
tar -zxvf BerkeleyDB-0.42.tar.gz
cd /home/alexa-seq/perl/BerkeleyDB-0.42
```

Edit the following lines in the 'config.in' file to specify the correct include and lib folders for BerkeleyDB as installed above:

```
INCLUDE = /home/alexa-seq/lib/BerkeleyDB_x64/include
LIB      = /home/alexa-seq/lib/BerkeleyDB_x64/lib
```

Install the Perl module:

```
perl Makefile.PL PREFIX=/home/alexa-seq/perl
make
make install
```

Add the following to your .bashrc file for BerkeleyDB and make sure the version of perl specified below is correct (i.e. these paths must actually exist):

```
export PERL5LIB=/home/alexa-seq/perl/lib64/perl5/site_perl/5.8.8/x86_64-linux-thread-multi:$PERL5LIB
export LD_LIBRARY_PATH=/home/alexa-seq/lib/BerkeleyDB_x64/lib:$LD_LIBRARY_PATH
```

6. Installing mysql server (requires root privileges)

If mysql is not already installed on your system, do so (requires root).

```
yum install mysql-server
yum install mysql-devel
```

6.1 Start MySQL server and required databases

```
/usr/bin/mysqld_safe --user=mysql &
```

Notes:

To stop server:

```
/etc/init.d/mysqld stop
```

To restart server:

```
/etc/init.d/mysqld restart
```

To start automatically on boot up:

```
/sbin/chkconfig mysqld on
```

6.2 Run script to secure database (server must be running; requires root)

```
/usr/bin/mysql_secure_installation
```

6.3 Set up user accounts

Login to mysql as root (password should have been created in secure step):

```
mysql -u root -p
```

Create a user (e.g., dbadmin) with all privileges except GRANT OPTION. Use this account for all database administration that doesn't require granting or revoking user privileges. Both root and dbadmin should only be used locally.

At MySQL prompt (replace XXXXXX with password of your choice):

```
CREATE USER 'dbadmin'@'localhost' IDENTIFIED BY 'XXXXXX';
GRANT ALL ON *.* TO 'dbadmin'@'localhost';
```

Create another user (e.g., viewer) with limited privileges. This user will be used by scripts to access (read only) the database.

At MySQL prompt:

```
CREATE USER 'viewer'@'localhost' IDENTIFIED BY 'XXXXXX';
CREATE USER 'viewer'@'%' IDENTIFIED BY 'XXXXXX';
GRANT SELECT, SHOW DATABASES ON *.* TO 'viewer'@'localhost';
GRANT SELECT, SHOW DATABASES ON *.* TO 'viewer'@'%';
```


6.4 Enable remote access (requires root)

This is necessary if you will be running pipeline on a different computer from the MySQL server). First, add a line to iptables to open mysql port (3306) in your firewall.

```
vi /etc/sysconfig/iptables
```

Insert the following with other RH-Firewall-1 rules:

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT
```

Then, restart firewall:

```
/etc/init.d/iptables restart
```

Test by trying to connect to mysql from another host

```
mysql -h your.host.name -u viewer -p
```

6.5 Modify mysql configuration

Modify or add to the mysql configuration file as follows:

```
#Increase the max allowed packet size
```

```
max_allowed_packet = 10M
```

```
#Increase the total number of simultaneous connections allowed (optional)
```

```
max_connections = 750
```

```
#Inactivate query cache (optional)
```

```
query_cache_size = 0
```

```
#Turn off binary logging by commenting out the line (optional):
```

```
#log-bin
```

6.6 Troubleshooting tips

If can't connect after doing the above, you might try adding the external host to your hosts file (e.g., /etc/hosts) or changing parameters in the mysql config file (/etc/my.cnf) such as 'port' and 'bind'. Remember to restart the mysql daemon for changes to take effect.

7. Updating .bashrc file

The following file can be used as a template:
/home/alexa-seq/ALEXA/alexa_seq/INSTALL.txt

The lines to be added will look something like this:

```
#####
#ENVIRONMENT VARIABLES NEEDED FOR ALEXA-SEQ PIPELINE - add to your .bashrc
#Make sure these make sense for your local system
#Variables for Berkeley DB perl module
export PERL5LIB=/home/alexa-seq/perl/lib64/perl5/site_perl/5.8.8/x86_64-linux-thread-
multi:$PERL5LIB
export LD_LIBRARY_PATH=/home/alexa-seq/lib/BerkeleyDB_x64/lib:$LD_LIBRARY_PATH

#Variables for x64 Cairo installation
export PKG_CONFIG_PATH=/home/alexa-seq/lib/Cairo_x64/lib/pkgconfig/:$PKG_CONFIG_PATH
export LD_LIBRARY_PATH=/home/alexa-seq/lib/Cairo_x64/lib/:$LD_LIBRARY_PATH

#Desired location to place perl storable files
export PERL_STORABLES_PATH=/home/alexa-seq/ALEXA/perl_storables/

#Location of Ensembl API and BioPerl files.
export ENSEMBL_API_PATH=/home/alexa-seq/ALEXA/alexa_seq/ensembl_api/

#location of R binaries
export PATH=./home/alexa-seq/bin/R64/R-2.10.1/bin:$PATH

#ENVIRONMENT VARIABLES NEEDED FOR ALEXA-SEQ PIPELINE
#####
```

8. Creating and editing config files

```
cp /home/alexa-seq/ALEXA/alexa_seq/config_files/examples/ALEXA_Seq_PIPELINE.conf
/home/alexa-seq/ALEXA/config_files/
cp /home/alexa-seq/ALEXA/alexa_seq/config_files/examples/ALEXA_Seq_ExampleProject.conf
/home/alexa-seq/ALEXA/config_files/ALEXA_Seq_YourProject.conf
```

Edit the two newly created conf files to reflect all your own paths as per the installations just completed.

ALEXA-Seq Analysis Steps

See Analysis section of user manual.

B. Creating a new ALEXA-Seq annotation database

Install local copy of Ensembl database

NOTE: This step is only required if you wish to create your own ALEXA-Seq annotation database. It is much easier to simply download one of the pre-computed annotation databases from the list here: (http://www.alexaplatform.org/alexa_seq/downloads.htm). Alternatively, if the species or genome version you want is missing, contact us and we can create it for you. Finally, if you wish to create it yourself install the Ensembl database for your species of interest only (not a complete mirror of Ensembl).

Log into mysql with sufficient privileges and create appropriate database.

At mysql prompt:

```
create database homo_sapiens_core_53_36o;  
exit;
```

Make temporary folder to hold files, download, and import into mysql

```
mkdir /home/alexa-seq/downloads/homo_sapiens_core_53_36o/  
cd /home/alexa-seq/downloads/homo_sapiens_core_53_36o/  
wget ftp://ftp.ensembl.org/pub/release-53/mysql/homo_sapiens_core_53_36o/*  
gunzip ./*  
mysql -u root -p homo_sapiens_core_53_36o < homo_sapiens_core_53_36o.sql  
mysqlimport -u root -p homo_sapiens_core_53_36o -L *.txt
```

ALEXA-Seq Annotation Steps

See section 3-C of the ALEXA-Seq user manual.

C. Set up the ALEXA-Seq data viewer

1. Installing Apache web server (requires root privileges)

1.1 Install Apache, Apache manuals, Secure Sockets Layer Support, and PHP

```
yum install httpd
yum install httpd-manual
```

Optional - install Secure Sockets Layer Support, and PHP

```
yum install mod_ssl
yum install php*
```

1.2 Set Apache to start on bootup

```
/sbin/chkconfig --level 235 httpd on
/sbin/chkconfig --list httpd
```

1.3 Configure Apache Server Settings (httpd.conf)

```
cd /etc/httpd/conf
cp httpd.conf httpd.conf.old
vi httpd.conf
```

Find and edit the following lines in httpd.conf (Near line number 235):

```
ServerAdmin your.email@domain.tld
```

```
ServerName your.ip.address
```

or

```
ServerName your_hostname.domain
```

Comment out the following line(s) as they will be set in the vhosts files below.
(Near line number 570):

```
ScriptAlias and <Directory "/var/www/cgi-bin"> ... </Directory>
```

Find and edit the following lines in httpd.conf (Near line number 1005):

```
NameVirtualHost your_hostname.domain:80
```

At very end of file:

```
<VirtualHost your_hostname.domain:80>
    ServerAdmin your.email@domain.tld
    ServerName your.ip.address
    DocumentRoot /www
    ErrorLog logs/error_log
    CustomLog logs/access_log combined
</VirtualHost>
```

include VirtualHosts config files

```
Include conf/vhosts/*.conf
```

1.4 Create the Web Site Directory Structure

```
mkdir /www
mkdir /www/alexa-seq
mkdir /www/alexa-seq/html
mkdir /www/alexa-seq/cgi-bin
mkdir /www/alexa-seq/logs
```

1.5 Optional - Add a Linux User Account specifically for the Web Site

```
/usr/sbin/useradd -d /www/alexa-seq/ -s /bin/bash alexa
```

These settings can be changed with CentOS GUI (System -> Administration -> Users and Groups). For example, for security reasons you can prevent ssh access for this account by setting the login shell to /sbin/nologin

Set password for new account :

```
passwd alexa
```

Change website directory ownership & permissions

```
cd /www
chown -R alexa:alexa alexa-seq/
```

1.6 Create VirtualHost Configuration Files

```
cd /etc/httpd/conf
mkdir vhosts
cd vhosts
vi alexa-seq.80.conf
```

The vhost file contents should look something like:

```
### Start VirtualHost file ###
<VirtualHost your_hostname.domain:80>
    ServerAdmin your_email@domain.tld
    ServerName alexa-seq
    ServerAlias www.alexa-seq.domain
    DocumentRoot /www/alexa-seq/html
    ScriptAlias /cgi-bin/ /www/alexa-seq/cgi-bin/
    ErrorLog /www/alexa-seq/logs/error_log
    CustomLog /www/alexa-seq/logs/access_log combined
</VirtualHost>

#Include the following to allow CGI scripts
#This is necessary for omega search to work properly (see step 10)
<Directory "/www/alexa-seq/cgi-bin/">
    Options +ExecCGI
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

AddHandler cgi-script .cgi
AddHandler cgi-script .pl
```

```
#Include the following to allow svgz support
#This is necessary for display of many ALEXA images
AddType image/svg+xml .svg
AddType image/svg+xml .svgz
AddEncoding gzip .svgz
<FilesMatch \.svgz$>
  <IfModule mod_gzip.c>
    mod_gzip_on No
  </IfModule>
</FilesMatch>
### End VirtualHost file ###
```

1.7 Start Apache and configure firewall

```
/sbin/service httpd start
```

Open port 80 for http if not already:

```
vi /etc/sysconfig/iptables
```

Add the following to iptables:

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp --dport 80 -j ACCEPT
```

Restart the firewall:

```
/etc/init.d/iptables restart
```

Go to browser and load:

http://your_hostname.domain/ or <http://your.ip.address/>

You should see the apache test page

Once websites are actually in place, they will be viewable at:

http://your_hostname.domain/alexa-seq/html/

You will need a test index.html file in this directory to test this

2. Installing xapian-omega (requires root privileges)

Login and install on webserver so that omega binary can find necessary libraries (you will likely need root):

```
mkdir /usr/lib/Xapian
cd /usr/lib/Xapian
```

2.1 Get current packages from <http://xapian.org/download> and unpack

```
wget http://oligarchy.co.uk/xapian/1.0.18/xapian-core-1.0.18.tar.gz
wget http://oligarchy.co.uk/xapian/1.0.18/xapian-omega-1.0.18.tar.gz
tar -zxvf xapian-core-1.0.18.tar.gz
tar -zxvf xapian-omega-1.0.18.tar.gz
```

2.2 Install xapian-core

```
mkdir xapian-omega-x64/
cd /usr/lib/Xapian/xapian-core-1.0.18
./configure --prefix=/usr/lib/Xapian/xapian-omega-x64
make
make install
```

2.3 Install xapian-omega

```
cd /usr/lib/Xapian/xapian-omega-1.0.18
./configure XAPIAN_CONFIG=/usr/lib/Xapian/xapian-omega-x64/bin/xapian-config
--prefix=/usr/lib/Xapian/xapian-omega-x64
make
make install
```

Now switch to a non-root user that has write privileges for the '/www/' directory

2.4 Create search data folders for omega

```
mkdir /www/alexaseq/cgi-bin/searchdata/
mkdir /www/alexaseq/cgi-bin/searchdata/cdb/
mkdir /www/alexaseq/cgi-bin/searchdata/templates/
mkdir /www/alexaseq/cgi-bin/searchdata/logs/
mkdir /www/alexaseq/cgi-bin/searchdata/data/
mkdir /www/alexaseq/cgi-bin/searchdata/data/default
mkdir /www/alexaseq/cgi-bin/xapian-omega/
mkdir /www/alexaseq/cgi-bin/xapian-omega/lib/
mkdir /www/alexaseq/cgi-bin/xapian-omega/lib/xapian-omega/
mkdir /www/alexaseq/cgi-bin/xapian-omega/lib/xapian-omega/bin/
```

copy template files from xapian-omega source dir to new templates dir:

```
cp -r /usr/lib/Xapian/xapian-omega-1.0.18/templates/* /www/alexaseq/cgi-
bin/searchdata/templates/
```

#Make this files executable

```
chmod +x /www/alexaseq/cgi-bin/searchdata/templates/*
chmod +x /www/alexaseq/cgi-bin/searchdata/templates/inc/*
```

2.5 Copy the omega executable and config file to your cgi-bin folder

```
cp /usr/lib/Xapian/xapian-omega-x64/lib/xapian-omega/bin/omega /www/alexaseq/cgi-bin/  
cp /usr/lib/Xapian/xapian-omega-x64/etc/omega.conf /www/alexaseq/cgi-bin/
```

Rename omega to omega.cgi (In many cases, apache requires a file extension to know what to do with script):

```
mv /www/alexaseq/cgi-bin/omega /www/alexaseq/cgi-bin/omega.cgi
```

2.6 Edit the omega.conf file to reflect these newly created data folders

```
vi /www/alexaseq/cgi-bin/omega.conf
```

It should look something like this:

```
### Begin omega.conf file ###  
#Directory containing Xapian databases  
#Warning: point to dir containing default/ dir, not to default dir itself  
database_dir /www/alexaseq/cgi-bin/searchdata/data  
  
#Directory containing OmegaScript templates:  
template_dir /www/alexaseq/cgi-bin/searchdata/templates  
  
#Directory to write Omega logs to:  
log_dir /www/alexaseq/cgi-bin/searchdata/logs  
  
#Directory containing any cdb files for the $lookup OmegaScript command:  
cdb_dir /www/alexaseq/cgi-bin/searchdata/cdb  
### End omega.conf file ###
```

2.7 Create index for website

Note: If you do not have a website yet, create dummy pages for testing purposes.

Note: you may need to play with the first parameter after --url to get links to work properly

```
/usr/lib/Xapian/xapian-omega-x64/bin/omindex --db /www/alexaseq/cgi-  
bin/searchdata/data/default --url /alexaseq/html/ /www/alexaseq/html/
```

2.8 Test omega

Test at command line:

```
cd /www/alexaseq/cgi-bin  
./omega "P=some_search_phrase" HITSPPERPAGE=10
```

Then test in browser:

```
http://your_hostname.domain/alexaseq/cgi-bin/omega.cgi
```

2.9 Customize

If you would like to be able to link to omega without the .cgi extension showing in browsers, you can add something like the following to your httpd.conf or vhosts file within the <Directory "/www/alexaseq/cgi-bin/"> </Directory> tags:

```
RewriteEngine on  
RewriteRule ^omega$ omega.cgi
```


To customize the look of the omega search page, edit the template files in:
/www/alex-seq/omega/templates (start with 'query')

You can also use the alexa templates provided as a starting point:
/home/alex-seq/ALEXA/alex-seq/website/web_files/xapian_omega/templates

2.10 Apache CGI trouble-shooting

If the omega does not run in your browser, check error logs for a hint as to what is going on. Most likely omega does not have permission/access to a required library.

```
tail /etc/httpd/logs/error_log
```

2.11 Creating multiple independent instances of omega search engine

The instructions outlined so far, assume that a single instance of the omega search engine will be used for all projects analyzed in the ALEXA pipeline. This has the advantage that if you search for a gene, you will see search results for all processed projects. If you wish to have search results that are independent for each project, you will need to create separate instances of the omega binary and search indexes. This can be done by creating project subfolders within the omega/ and cgi-bin/ folders. You will need to edit omega.conf, httpd.conf (or vhost file), and ALEXA conf files accordingly.