CCNx: Content Centric Networking

Palo Alto Research Center (PARC)

Akmal Khan

Topics in Internet

SNU CSE Spring 2011

2011-03-28

Agenda

Project CCNx™

- ➢ Package contents
- ➢ Supported platforms and development tools
- ➢ Build and install instructions
- ➢ Running the programs(Apps)
- ➢ Developing your own Apps
- ➢ Conclusion

Project CCNx

- Project CCNx™ is an open source project exploring the next step in networking, based on one fundamental architectural change: replacing named hosts with named content as the primary abstraction
- *Early-stage specifications , software libraries and tools intended for use by researchers*
  - http://www.ccnx.org/content/download-releases
    - ccnx-0.3.0.tar.gz [2010-11-04]
  - https://github.com/ProjectCCNx/ccnx
    - https://github.com/ProjectCCNx/ccnx#readme
  - http://www.ccnx.org/content/docs

Agenda

Project CCNx™

- ➢ Package contents
- ➢ Supported platforms and development tools
- ➢ Build and install instructions
- ➢ Running the programs(Apps)
- ➢ Developing your own Apps
- ➢ Conclusion

Package contents

- C/POSIX and Java reference implementation of CCNx forwarder, library, primitive utilities, skeleton API docs, and unit test suite.
- Minimal sample Apps
  - (ccnChat) to demonstrate basic communication on local LAN.
  - file proxy (ccnFileProxy) to demonstrate basic communication on local LAN.
  - vlc (media transport) and wireshark (packet dissector) plugins
  - Android implementation for smartphones.
    - Android implementation has a service wrapper for ccnd and the repository. It also has a CCN Chat implementation.
- Documentation is built from source files of various kinds (using a combination of doxygen and asciidoc)

Agenda

Project CCNx™

- Package contents
- Supported platforms and development tools
- Build and install instructions
- Running the programs(Apps)
- Developing your own Apps
- Conclusion

Supported platforms

- Only Unix-like platforms are currently supported.
  - tested on Ubuntu Linux, MacOS, Solaris, and FreeBSD.
- Limited Cygwin and no support for Windows platforms yet.
- Android is supported

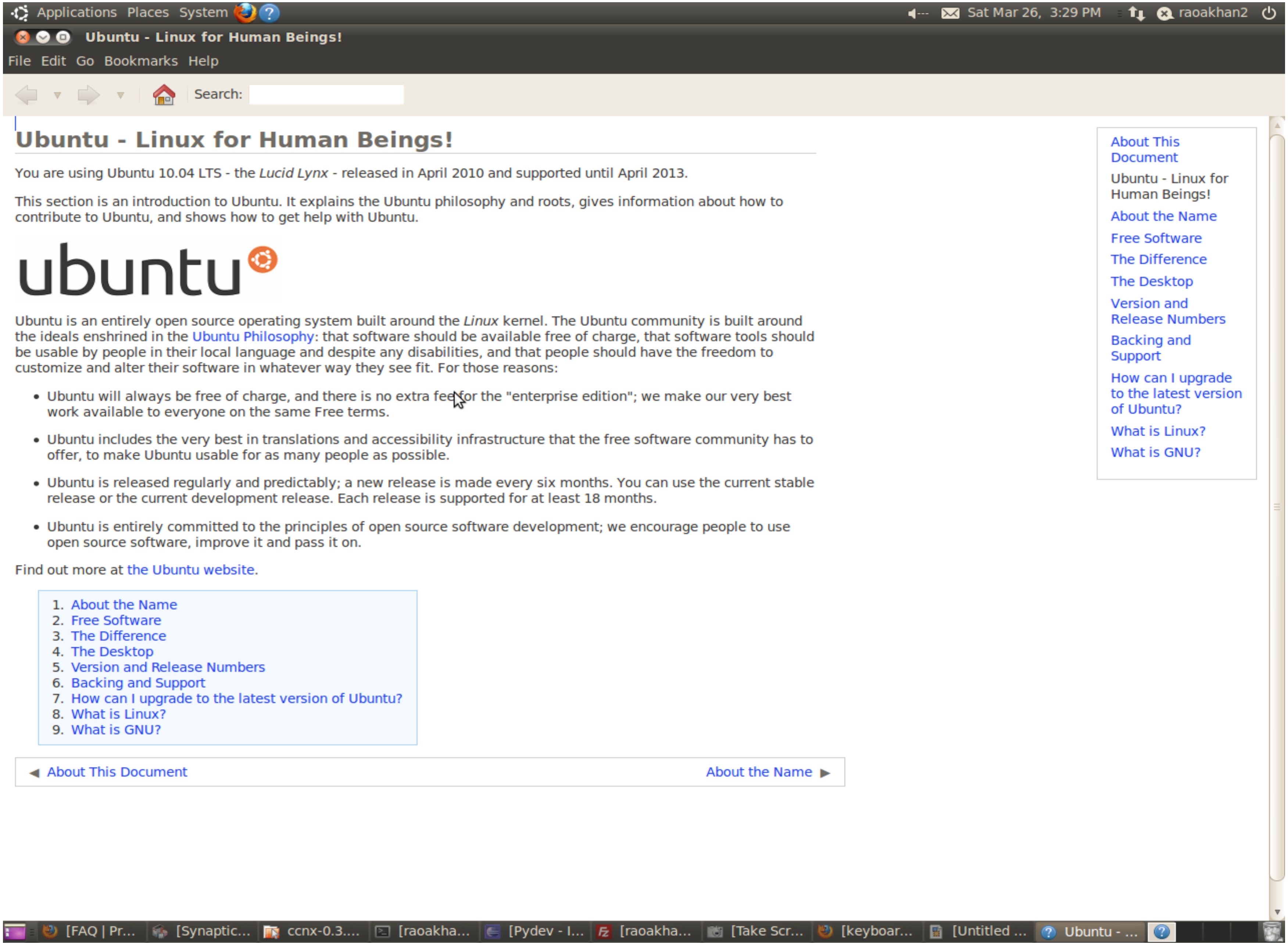
LANGUAGE REQUIREMENTS AND TOOLS

- Require a standard toolchain including gcc, make etc.
  - libcrypto >= 0.9.8 from openssl available from http://openssl.org/source/
  - expat available from http://sourceforge.net/projects/expat/
  - libpcap available from http://www.tcpdump.org (optional, needed for certain utilities only)
  - libxml2 available from xmlsoft.org
    - In addition, you will need vlc and wireshark to build and use the CCNx plugins for those packages.
- See csrc/README* files for further notes about what needs to be installed or configured on each OS.
- JAVA LANGUAGE REQUIREMENTS AND TOOLS
  - For parts of the system written in Java, CCNx code is tested with Sun Java JDK Java JDK 1.5 and 1.6 only
- Eclipse IDE for Java Developers for your own Apps
  - http://eclipse.org

Agenda

Project CCNx™

- ➢ Package contents
- ➢ Supported platforms and development tools
- ➢ Build and install instructions
- ➢ Running the programs(Apps)
- ➢ Developing your own Apps
- ➢ Conclusion



Ubuntu - Linux for Human Beings!

You are using Ubuntu 10.04 LTS - the *Lucid Lynx* - released in April 2010 and supported until April 2013.

This section is an introduction to Ubuntu. It explains the Ubuntu philosophy and roots, gives information about how to contribute to Ubuntu, and shows how to get help with Ubuntu.

ubuntu

Ubuntu is an entirely open source operating system built around the *Linux* kernel. The Ubuntu community is built around the ideals enshrined in the Ubuntu Philosophy: that software should be available free of charge, that software tools should be usable by people in their local language and despite any disabilities, and that people should have the freedom to customize and alter their software in whatever way they see fit. For those reasons:

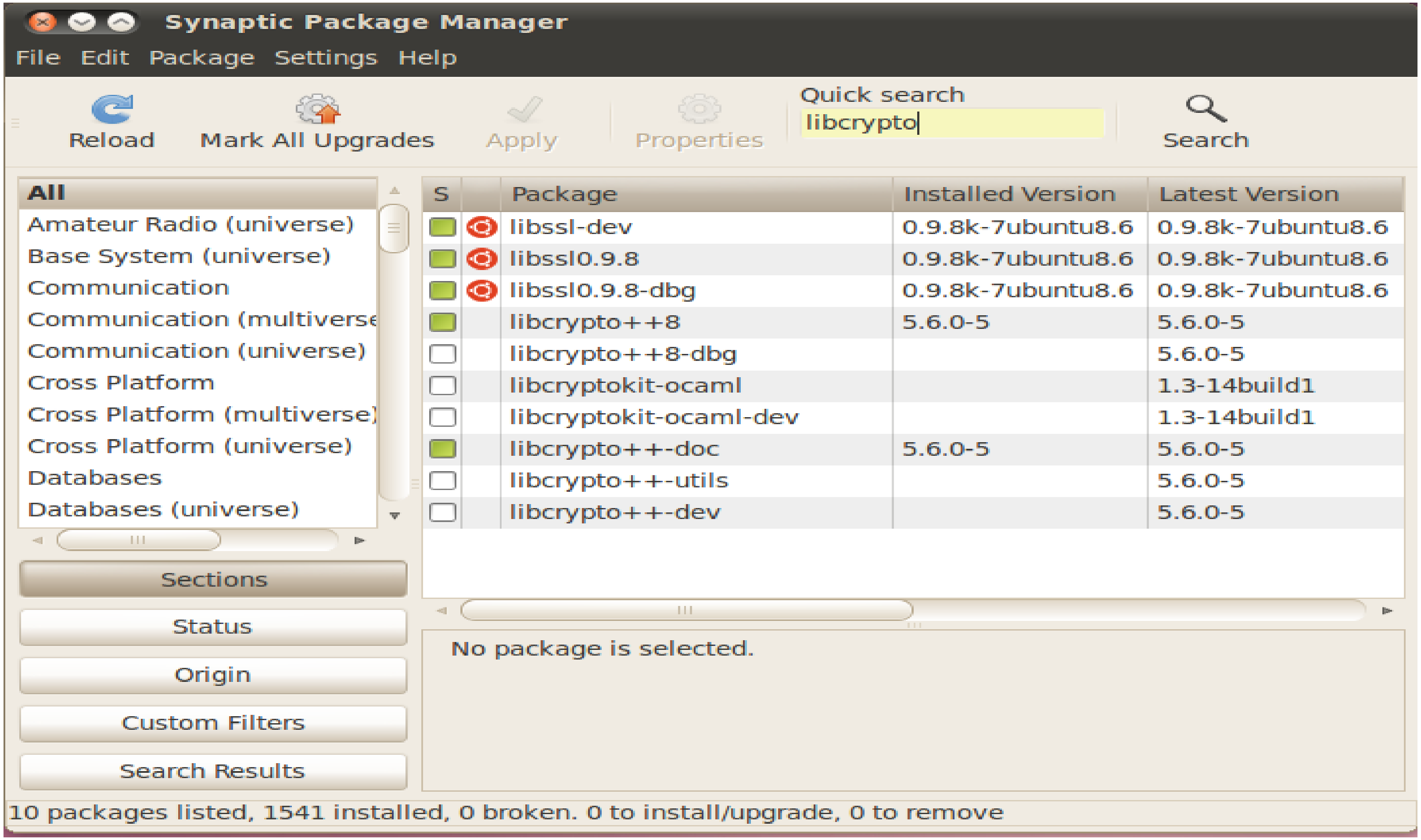
- Ubuntu will always be free of charge, and there is no extra fee for the "enterprise edition"; we make our very best work available to everyone on the same Free terms.
- Ubuntu includes the very best in translations and accessibility infrastructure that the free software community has to offer, to make Ubuntu usable for as many people as possible.
- Ubuntu is released regularly and predictably; a new release is made every six months. You can use the current stable release or the current development release. Each release is supported for at least 18 months.
- Ubuntu is entirely committed to the principles of open source software development; we encourage people to use open source software, improve it and pass it on.

Find out more at the Ubuntu website.

◀ About This Document
About the Name ▶

Install,remove and upgrade software packages

System->Administration->Synaptic Package Manager



Quick search: libcrypto

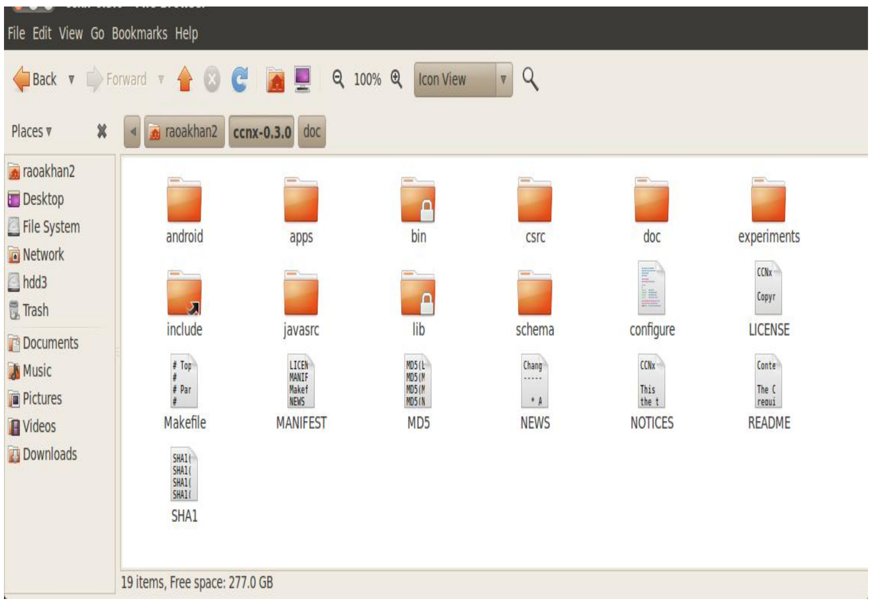
S		Package	Installed Version	Latest Version
🟩		libssl-dev	0.9.8k-7ubuntu8.6	0.9.8k-7ubuntu8.6
🟩		libssl0.9.8	0.9.8k-7ubuntu8.6	0.9.8k-7ubuntu8.6
🟩		libssl0.9.8-dbg	0.9.8k-7ubuntu8.6	0.9.8k-7ubuntu8.6
🟩		libcrypto++8	5.6.0-5	5.6.0-5
□		libcrypto++8-dbg		5.6.0-5
□		libcryptokit-ocaml		1.3-14build1
□		libcryptokit-ocaml-dev		1.3-14build1
🟩		libcrypto++-doc	5.6.0-5	5.6.0-5
□		libcrypto++-utils		5.6.0-5
□		libcrypto++-dev		5.6.0-5

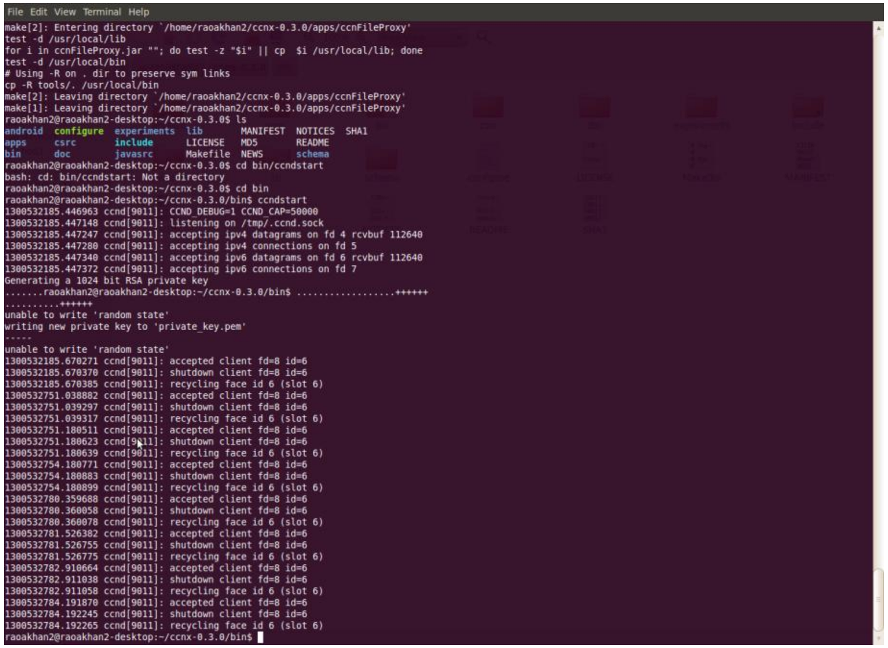
No package is selected.

10 packages listed, 1541 installed, 0 broken. 0 to install/upgrade, 0 to remove

Use the command line Applications->terminal

1. sudo ./configure
2. sudo make
3. sudo make test
4. sudo make install
5. cd bin/ccndstart



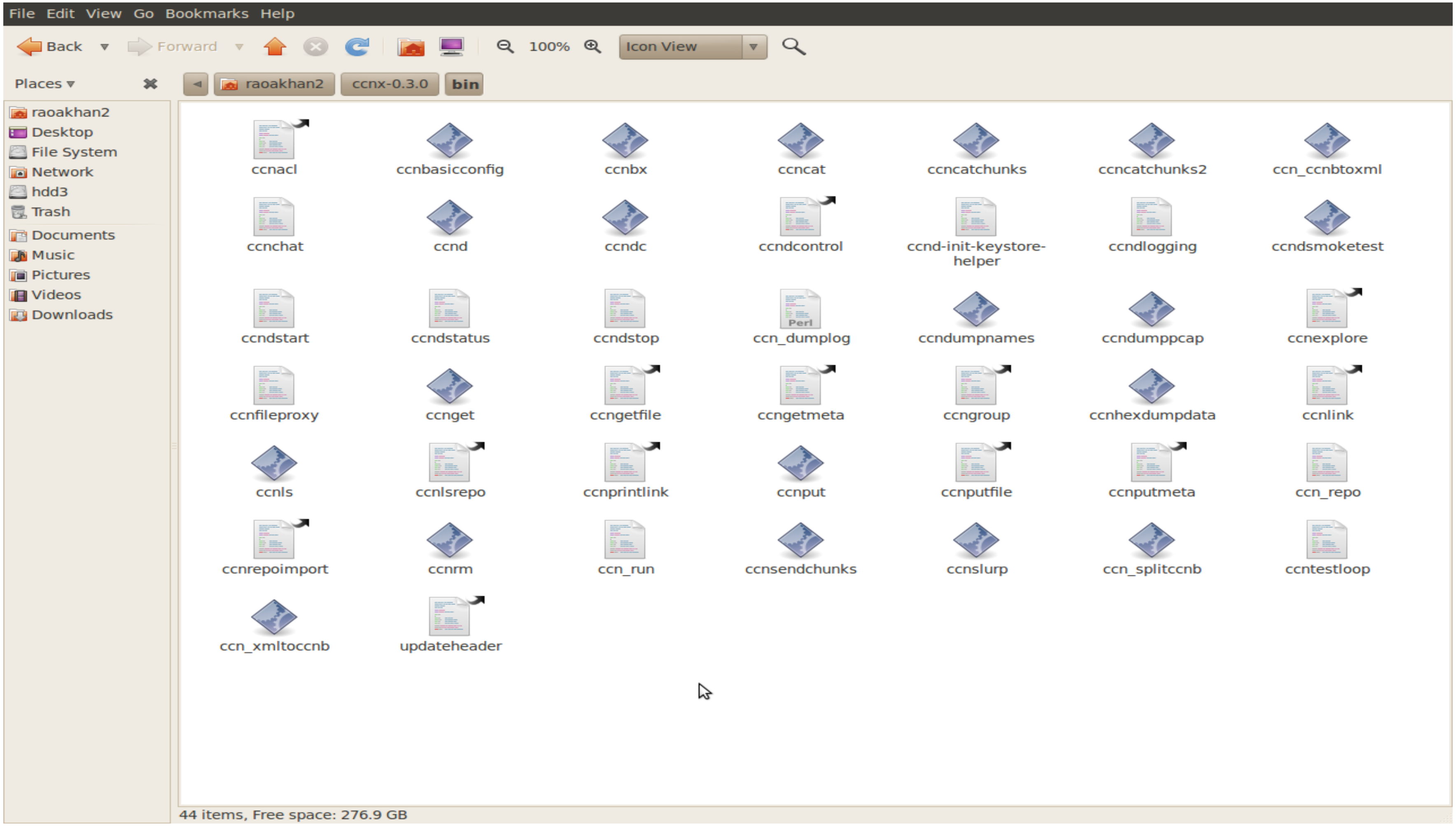


```
make[2]: Entering directory `/home/raoakhan2/ccnx-0.3.0/apps/ccnFileProxy'
test -d /usr/local/lib
for i in ccnFileProxy.jar ""; do test -z "$i" || cp  $i /usr/local/lib; done
test -d /usr/local/bin
# Using -R on . dir to preserve sym links
cp -R tools/. /usr/local/bin
make[2]: Leaving directory `/home/raoakhan2/ccnx-0.3.0/apps/ccnFileProxy'
make[1]: Leaving directory `/home/raoakhan2/ccnx-0.3.0/apps/ccnFileProxy'
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0$ ls
android  configure  experiments  lib       MANIFEST  NOTICES  SHA1
apps     csrc       include      LICENSE   MD5       README
bin      doc        javasrc      Makefile  NEWS      schema
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0$ cd bin/ccndstart
bash: cd: bin/ccndstart: Not a directory
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0$ cd bin
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$ ccndstart
1300532185.446963 ccnd[9011]: CCND_DEBUG=1 CCND_CAP=50000
1300532185.447148 ccnd[9011]: listening on /tmp/.ccnd.sock
1300532185.447247 ccnd[9011]: accepting ipv4 datagrams on fd 4 rcvbuf 112640
1300532185.447280 ccnd[9011]: accepting ipv4 connections on fd 5
1300532185.447340 ccnd[9011]: accepting ipv6 datagrams on fd 6 rcvbuf 112640
1300532185.447372 ccnd[9011]: accepting ipv6 connections on fd 7
Generating a 1024 bit RSA private key
.......raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$ ..................++++++
...........++++++
unable to write 'random state'
writing new private key to 'private_key.pem'
-----
unable to write 'random state'
1300532185.670271 ccnd[9011]: accepted client fd=8 id=6
1300532185.670370 ccnd[9011]: shutdown client fd=8 id=6
1300532185.670385 ccnd[9011]: recycling face id 6 (slot 6)
1300532751.038882 ccnd[9011]: accepted client fd=8 id=6
1300532751.039297 ccnd[9011]: shutdown client fd=8 id=6
1300532751.039317 ccnd[9011]: recycling face id 6 (slot 6)
1300532751.180511 ccnd[9011]: accepted client fd=8 id=6
1300532751.180623 ccnd[9011]: shutdown client fd=8 id=6
1300532751.180639 ccnd[9011]: recycling face id 6 (slot 6)
1300532754.180771 ccnd[9011]: accepted client fd=8 id=6
1300532754.180883 ccnd[9011]: shutdown client fd=8 id=6
1300532754.180899 ccnd[9011]: recycling face id 6 (slot 6)
1300532780.359688 ccnd[9011]: accepted client fd=8 id=6
1300532780.360058 ccnd[9011]: shutdown client fd=8 id=6
1300532780.360078 ccnd[9011]: recycling face id 6 (slot 6)
1300532781.526382 ccnd[9011]: accepted client fd=8 id=6
1300532781.526755 ccnd[9011]: shutdown client fd=8 id=6
1300532781.526775 ccnd[9011]: recycling face id 6 (slot 6)
1300532782.910664 ccnd[9011]: accepted client fd=8 id=6
1300532782.911038 ccnd[9011]: shutdown client fd=8 id=6
1300532782.911058 ccnd[9011]: recycling face id 6 (slot 6)
1300532784.191870 ccnd[9011]: accepted client fd=8 id=6
1300532784.192245 ccnd[9011]: shutdown client fd=8 id=6
1300532784.192265 ccnd[9011]: recycling face id 6 (slot 6)
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$
```

CCNx code base consists of

- Core network/routing daemon, *ccnd*. every CCNx node needs to run a ccnd.
  - control programs to set routes that determine how *ccnd* forwards traffic, *ccndc,* which basically populates static routes into ccnd's forwarding table.
  - load routes from a configuration file in the user's home directory (~/.ccnx/ccnd.conf) if one exists, or can be given new routes on the command line.
- *Repository*, currently in Java.
  - Think of ccnd's cache as ephemeral storage, and a repository as long-term storage analogous to a hard drive.
    - Anything other than the most simple or network-focused of applications will run a repository, locally or remotely.

Ccnx3.0/bin

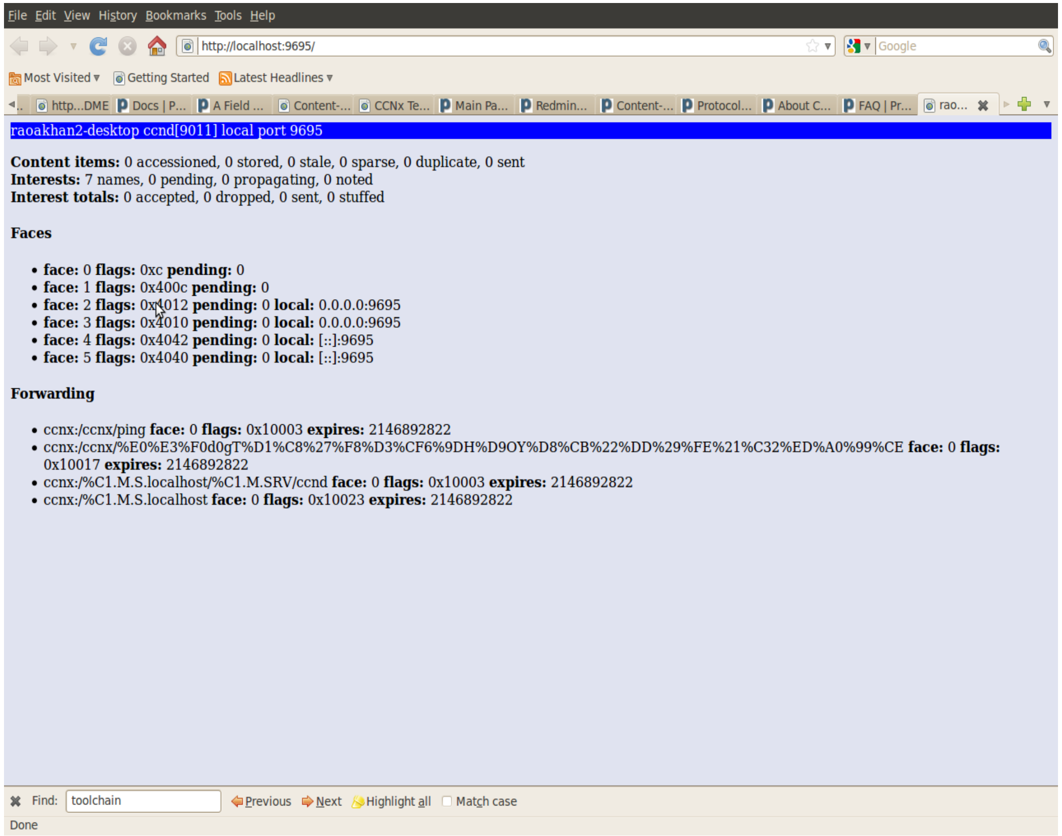


ccnacl	ccnbasicconfig	ccnbx	ccncat	ccncatchunks	ccncatchunks2	ccn_ccnbtoxml
ccnchat	ccnd	ccndc	ccndcontrol	ccnd-init-keystore-helper	ccndlogging	ccndsmoketest
ccndstart	ccndstatus	ccndstop	ccn_dumplog	ccndumpnames	ccndumppcap	ccnexplore
ccnfileproxy	ccnget	ccngetfile	ccngetmeta	ccngroup	ccnhexdumpdata	ccnlink
ccnls	ccnlsrepo	ccnprintlink	ccnput	ccnputfile	ccnputmeta	ccn_repo
ccnrepoimport	ccnrm	ccn_run	ccnsendchunks	ccnslurp	ccn_splitccnb	ccntestloop
ccn_xmltoccnb	updateheader					

44 items, Free space: 276.9 GB

Ccnx web interface

- web interface
  - cache statistics and forwarding table
  - http://localhost:9695



raoakhan2-desktop ccnd[9011] local port 9695

Content items: 0 accessioned, 0 stored, 0 stale, 0 sparse, 0 duplicate, 0 sent
Interests: 7 names, 0 pending, 0 propagating, 0 noted
Interest totals: 0 accepted, 0 dropped, 0 sent, 0 stuffed

Faces

- **face:** 0 **flags:** 0xc **pending:** 0
- **face:** 1 **flags:** 0x400c **pending:** 0
- **face:** 2 **flags:** 0x4012 **pending:** 0 **local:** 0.0.0.0:9695
- **face:** 3 **flags:** 0x4010 **pending:** 0 **local:** 0.0.0.0:9695
- **face:** 4 **flags:** 0x4042 **pending:** 0 **local:** [::]:9695
- **face:** 5 **flags:** 0x4040 **pending:** 0 **local:** [::]:9695

Forwarding

- ccnx:/ccnx/ping **face:** 0 **flags:** 0x10003 **expires:** 2146892822
- ccnx:/ccnx/%E0%E3%F0d0gT%D1%C8%27%F8%D3%CF6%9DH%D9OY%D8%CB%22%DD%29%FE%21%C32%ED%A0%99%CE **face:** 0 **flags:** 0x10017 **expires:** 2146892822
- ccnx:/%C1.M.S.localhost/%C1.M.SRV/ccnd **face:** 0 **flags:** 0x10003 **expires:** 2146892822
- ccnx:/%C1.M.S.localhost **face:** 0 **flags:** 0x10023 **expires:** 2146892822

Agenda

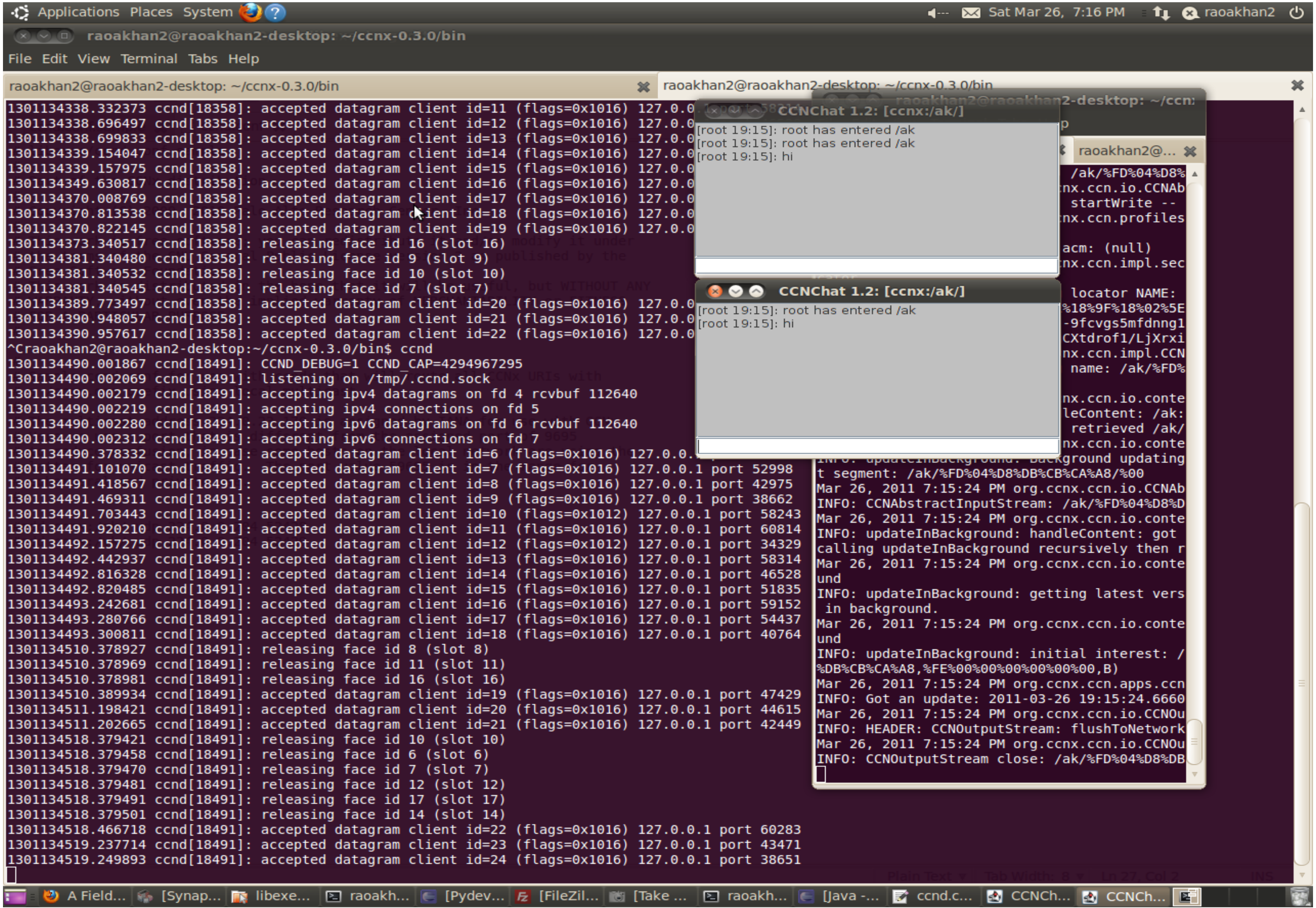
Project CCNx™

- Package contents
- Supported platforms and development tools
- Build and install instructions
- Running the programs(Apps)
- Developing your own Apps
- Conclusion

Play with some simple apps

- ccnChat
  - sudo ccnchat ccnx:/akmal/
  - You can run more than one ccnChat on the same machine and they will talk to each other;
  - if your forwarding is set up properly
    - you can have ccnChats on multiple machines talk back and forth.

ccnChat



Applications Places System
Sat Mar 26, 7:16 PM
raoakhan2
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
File Edit View Terminal Tabs Help
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
raoakhan2@raoakhan2-desktop: ~/ccn
raoakhan2@...
CCNChat 1.2: [ccnx:/ak/]
CCNChat 1.2: [ccnx:/ak/]
A Field... [Synap... libexe... raoakh... [Pydev... [FileZil... [Take ... raoakh... [Java -... ccnd.c... CCNCh... CCNCh...

```
1301134338.332373 ccnd[18358]: accepted datagram client id=11 (flags=0x1016) 127.0.0
1301134338.696497 ccnd[18358]: accepted datagram client id=12 (flags=0x1016) 127.0.0
1301134338.699833 ccnd[18358]: accepted datagram client id=13 (flags=0x1016) 127.0.0
1301134339.154047 ccnd[18358]: accepted datagram client id=14 (flags=0x1016) 127.0.0
1301134339.157975 ccnd[18358]: accepted datagram client id=15 (flags=0x1016) 127.0.0
1301134349.630817 ccnd[18358]: accepted datagram client id=16 (flags=0x1016) 127.0.0
1301134370.008769 ccnd[18358]: accepted datagram client id=17 (flags=0x1016) 127.0.0
1301134370.813538 ccnd[18358]: accepted datagram client id=18 (flags=0x1016) 127.0.0
1301134370.822145 ccnd[18358]: accepted datagram client id=19 (flags=0x1016) 127.0.0
1301134373.340517 ccnd[18358]: releasing face id 16 (slot 16)
1301134381.340480 ccnd[18358]: releasing face id 9 (slot 9)
1301134381.340532 ccnd[18358]: releasing face id 10 (slot 10)
1301134381.340545 ccnd[18358]: releasing face id 7 (slot 7)
1301134389.773497 ccnd[18358]: accepted datagram client id=20 (flags=0x1016) 127.0.0
1301134390.948057 ccnd[18358]: accepted datagram client id=21 (flags=0x1016) 127.0.0
1301134390.957617 ccnd[18358]: accepted datagram client id=22 (flags=0x1016) 127.0.0
^Craoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$ ccnd
1301134490.001867 ccnd[18491]: CCND_DEBUG=1 CCND_CAP=4294967295
1301134490.002069 ccnd[18491]: listening on /tmp/.ccnd.sock
1301134490.002179 ccnd[18491]: accepting ipv4 datagrams on fd 4 rcvbuf 112640
1301134490.002219 ccnd[18491]: accepting ipv4 connections on fd 5
1301134490.002280 ccnd[18491]: accepting ipv6 datagrams on fd 6 rcvbuf 112640
1301134490.002312 ccnd[18491]: accepting ipv6 connections on fd 7
1301134490.378332 ccnd[18491]: accepted datagram client id=6 (flags=0x1016) 127.0.0.
1301134491.101070 ccnd[18491]: accepted datagram client id=7 (flags=0x1016) 127.0.0.1 port 52998
1301134491.418567 ccnd[18491]: accepted datagram client id=8 (flags=0x1016) 127.0.0.1 port 42975
1301134491.469311 ccnd[18491]: accepted datagram client id=9 (flags=0x1016) 127.0.0.1 port 38662
1301134491.703443 ccnd[18491]: accepted datagram client id=10 (flags=0x1012) 127.0.0.1 port 58243
1301134491.920210 ccnd[18491]: accepted datagram client id=11 (flags=0x1016) 127.0.0.1 port 60814
1301134492.157275 ccnd[18491]: accepted datagram client id=12 (flags=0x1012) 127.0.0.1 port 34329
1301134492.442937 ccnd[18491]: accepted datagram client id=13 (flags=0x1016) 127.0.0.1 port 58314
1301134492.816328 ccnd[18491]: accepted datagram client id=14 (flags=0x1016) 127.0.0.1 port 46528
1301134492.820485 ccnd[18491]: accepted datagram client id=15 (flags=0x1016) 127.0.0.1 port 51835
1301134493.242681 ccnd[18491]: accepted datagram client id=16 (flags=0x1016) 127.0.0.1 port 59152
1301134493.280766 ccnd[18491]: accepted datagram client id=17 (flags=0x1016) 127.0.0.1 port 54437
1301134493.300811 ccnd[18491]: accepted datagram client id=18 (flags=0x1016) 127.0.0.1 port 40764
1301134510.378927 ccnd[18491]: releasing face id 8 (slot 8)
1301134510.378969 ccnd[18491]: releasing face id 11 (slot 11)
1301134510.378981 ccnd[18491]: releasing face id 16 (slot 16)
1301134510.389934 ccnd[18491]: accepted datagram client id=19 (flags=0x1016) 127.0.0.1 port 47429
1301134511.198421 ccnd[18491]: accepted datagram client id=20 (flags=0x1016) 127.0.0.1 port 44615
1301134511.202665 ccnd[18491]: accepted datagram client id=21 (flags=0x1016) 127.0.0.1 port 42449
1301134518.379421 ccnd[18491]: releasing face id 10 (slot 10)
1301134518.379458 ccnd[18491]: releasing face id 6 (slot 6)
1301134518.379470 ccnd[18491]: releasing face id 7 (slot 7)
1301134518.379481 ccnd[18491]: releasing face id 12 (slot 12)
1301134518.379491 ccnd[18491]: releasing face id 17 (slot 17)
1301134518.379501 ccnd[18491]: releasing face id 14 (slot 14)
1301134518.466718 ccnd[18491]: accepted datagram client id=22 (flags=0x1016) 127.0.0.1 port 60283
1301134519.237714 ccnd[18491]: accepted datagram client id=23 (flags=0x1016) 127.0.0.1 port 43471
1301134519.249893 ccnd[18491]: accepted datagram client id=24 (flags=0x1016) 127.0.0.1 port 38651
```

```
[root 19:15]: root has entered /ak
[root 19:15]: root has entered /ak
[root 19:15]: hi
```

```
[root 19:15]: root has entered /ak
[root 19:15]: hi
```

```
 /ak/%FD%04%D8%
nx.ccn.io.CCNAb
 startWrite --
nx.ccn.profiles

acm: (null)
nx.ccn.impl.sec

 locator NAME:
%18%9F%18%02%5E
-9fcvgs5mfdnng1
CXtdrof1/LjXrxi
nx.ccn.impl.CCN
 name: /ak/%FD%

nx.ccn.io.conte
leContent: /ak:
 retrieved /ak/
nx.ccn.io.conte
INFO: updateInBackground: Background updating
t segment: /ak/%FD%04%D8%DB%CB%CA%A8/%00
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.io.CCNAb
INFO: CCNAbstractInputStream: /ak/%FD%04%D8%D
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.io.conte
INFO: updateInBackground: handleContent: got
calling updateInBackground recursively then r
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.io.conte
und
INFO: updateInBackground: getting latest vers
 in background.
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.io.conte
und
INFO: updateInBackground: initial interest: /
%DB%CB%CA%A8,%FE%00%00%00%00%00%00,B)
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.apps.ccn
INFO: Got an update: 2011-03-26 19:15:24.6660
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.io.CCNOu
INFO: HEADER: CCNOutputStream: flushToNetwork
Mar 26, 2011 7:15:24 PM org.ccnx.ccn.io.CCNOu
INFO: CCNOutputStream close: /ak/%FD%04%D8%DB
```

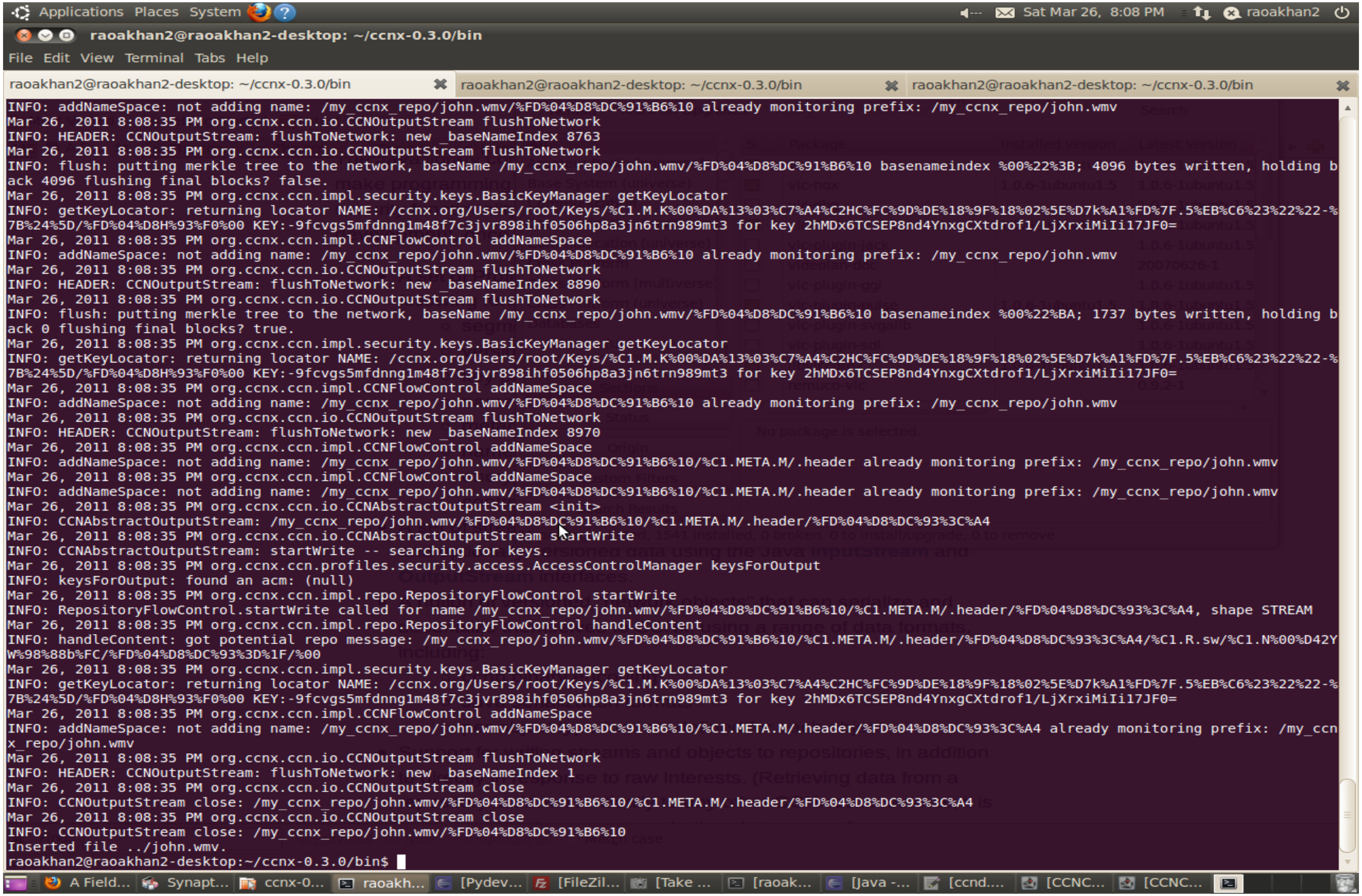
REPOSITORY

- persistent storage of CCNx content backed by a file system, and responds to interests in the content it has available.
  - bin/ccn_repo (give a usage message showing options.)
- Start a repository with the name of a directory to use for its backing store.
  - bin/ccn_repo ~/my_ccnx_repo
  - ccn_repo will turn into a daemon.
  - Note the PID given in the output so you can terminate it later
    - bin/ccn_repo stop 17328
- Do not run two repositories on the same backing store directory at the same time.
  - The experimental implementation has no protection to prevent this.

ccnputfile/ccngetfile

- pair of programs to write files from the file system or web into CCNx, and read file-oriented data out again.
  - If started at the same time, they can write directly to each other;
  - otherwise ccnputfile can be used to load data into a repository, and ccngetfile to read data out again.

ccnputfile -unversioned ccnx:/my_ccnx_repo /john.wmv ../john.wmv


Applications Places System
Sat Mar 26, 8:08 PM
raoakhan2
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
File Edit View Terminal Tabs Help
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
raoakhan2@raoakhan2-desktop: ~/ccnx-0.3.0/bin
A Field...
Synapt...
ccnx-0...
raoakh...
[Pydev...
[FileZil...
[Take ...
[raoak...
[Java -...
[ccnd....
[CCNC...
[CCNC...

```
INFO: addNameSpace: not adding name: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10 already monitoring prefix: /my_ccnx_repo/john.wmv
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream flushToNetwork
INFO: HEADER: CCNOutputStream: flushToNetwork: new _baseNameIndex 8763
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream flushToNetwork
INFO: flush: putting merkle tree to the network, baseName /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10 basenameindex %00%22%3B; 4096 bytes written, holding b
ack 4096 flushing final blocks? false.
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.security.keys.BasicKeyManager getKeyLocator
INFO: getKeyLocator: returning locator NAME: /ccnx.org/Users/root/Keys/%C1.M.K%00%DA%13%03%C7%A4%C2HC%FC%9D%DE%18%9F%18%02%5E%D7k%A1%FD%7F.5%EB%C6%23%22%22-%
7B%24%5D/%FD%04%D8H%93%F0%00 KEY:-9fcvgs5mfdnng1m48f7c3jvr898ihf0506hp8a3jn6trn989mt3 for key 2hMDx6TCSEP8nd4YnxgCXtdrof1/LjXrxiMiIi17JF0=
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.CCNFlowControl addNameSpace
INFO: addNameSpace: not adding name: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10 already monitoring prefix: /my_ccnx_repo/john.wmv
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream flushToNetwork
INFO: HEADER: CCNOutputStream: flushToNetwork: new _baseNameIndex 8890
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream flushToNetwork
INFO: flush: putting merkle tree to the network, baseName /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10 basenameindex %00%22%BA; 1737 bytes written, holding b
ack 0 flushing final blocks? true.
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.security.keys.BasicKeyManager getKeyLocator
INFO: getKeyLocator: returning locator NAME: /ccnx.org/Users/root/Keys/%C1.M.K%00%DA%13%03%C7%A4%C2HC%FC%9D%DE%18%9F%18%02%5E%D7k%A1%FD%7F.5%EB%C6%23%22%22-%
7B%24%5D/%FD%04%D8H%93%F0%00 KEY:-9fcvgs5mfdnng1m48f7c3jvr898ihf0506hp8a3jn6trn989mt3 for key 2hMDx6TCSEP8nd4YnxgCXtdrof1/LjXrxiMiIi17JF0=
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.CCNFlowControl addNameSpace
INFO: addNameSpace: not adding name: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10 already monitoring prefix: /my_ccnx_repo/john.wmv
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream flushToNetwork
INFO: HEADER: CCNOutputStream: flushToNetwork: new _baseNameIndex 8970
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.CCNFlowControl addNameSpace
INFO: addNameSpace: not adding name: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header already monitoring prefix: /my_ccnx_repo/john.wmv
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.CCNFlowControl addNameSpace
INFO: addNameSpace: not adding name: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header already monitoring prefix: /my_ccnx_repo/john.wmv
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNAbstractOutputStream <init>
INFO: CCNAbstractOutputStream: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header/%FD%04%D8%DC%93%3C%A4
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNAbstractOutputStream startWrite
INFO: CCNAbstractOutputStream: startWrite -- searching for keys.
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.profiles.security.access.AccessControlManager keysForOutput
INFO: keysForOutput: found an acm: (null)
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.repo.RepositoryFlowControl startWrite
INFO: RepositoryFlowControl.startWrite called for name /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header/%FD%04%D8%DC%93%3C%A4, shape STREAM
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.repo.RepositoryFlowControl handleContent
INFO: handleContent: got potential repo message: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header/%FD%04%D8%DC%93%3C%A4/%C1.R.sw/%C1.N%00%D42Y
W%98%88b%FC/%FD%04%D8%DC%93%3D%1F/%00
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.security.keys.BasicKeyManager getKeyLocator
INFO: getKeyLocator: returning locator NAME: /ccnx.org/Users/root/Keys/%C1.M.K%00%DA%13%03%C7%A4%C2HC%FC%9D%DE%18%9F%18%02%5E%D7k%A1%FD%7F.5%EB%C6%23%22%22-%
7B%24%5D/%FD%04%D8H%93%F0%00 KEY:-9fcvgs5mfdnng1m48f7c3jvr898ihf0506hp8a3jn6trn989mt3 for key 2hMDx6TCSEP8nd4YnxgCXtdrof1/LjXrxiMiIi17JF0=
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.impl.CCNFlowControl addNameSpace
INFO: addNameSpace: not adding name: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header/%FD%04%D8%DC%93%3C%A4 already monitoring prefix: /my_ccn
x_repo/john.wmv
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream flushToNetwork
INFO: HEADER: CCNOutputStream: flushToNetwork: new _baseNameIndex 1
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream close
INFO: CCNOutputStream close: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10/%C1.META.M/.header/%FD%04%D8%DC%93%3C%A4
Mar 26, 2011 8:08:35 PM org.ccnx.ccn.io.CCNOutputStream close
INFO: CCNOutputStream close: /my_ccnx_repo/john.wmv/%FD%04%D8%DC%91%B6%10
Inserted file ../john.wmv.
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$ 
```

VLC: plugin

- plugin to the standard video player VLC that can read ccnx data.
  - Start a repo, load some content into it, and use VLC to play it.
    - ccnputfile -unversioned ccnx:/ my_ccnx_repo / john.wmv ../john.wmv
    - vlc ccn:/// my_ccnx_repo /john.wmv
    - Note: you need the triple-slash, because VLC doesn't even try to find an access module, and, I just noticed -- the name for the VLC CCNx access scheme is "ccn" where it should be "ccnx".

Other Apps

- ccnsendchunks
  - produces its chunks as it receives interests, or once per second, whichever is faster.
  - Its data can be read (at least) by ccncatchunks and ccncatchunks2.
- ccnFileProxy
  - front end that makes content in the file system accessible to CCNx.
  - Not highly optimized, but an alternative to loading a repository to get some data.
- Content Explorer: (ccnexplore)
  - GUI for browsing data stored in CCNx, basically like a simple file browser.
- ccnls to list the contents of ccnd's cache
- ccnlsrepo to list the contents of a repository or other name enumeration protocol responder from the command line

Agenda

Project CCNx™

- Package contents
- Supported platforms and development tools
- Build and install instructions
- Running the programs(Apps)
- Developing your own Apps
- Conclusion

CCNx Java Library

- Java library contains a number of higher-level APIs designed to make programming CCNx simpler, as well as more sophisticated security functionality.
- Profiles, conventions on names and data
  - Segmentation/versioning, including retrieval of latest versions of content
  - key publishing/name enumeration/metadata publication
  - /namespace management /encryption-based access control
- stream abstractions for reading and writing segmented, and optionally versioned data
  - writing streams and objects to repositories, in addition to directly in response to raw Interests.
- versioned "network objects" that can serialize and deserialize themselves to CCNx, using a range of data formats, including:
  - ccnb binary encoding/Java serialization formats/per-object type formats defined for each object
- Low-level support for content encryption and decryption using application-supplied keys.
  - Preliminary support for automated encryption and decryption of content using pluggable key distribution policies.

CCNx Java Library

- CCNx's notion of Links, including preliminary support for automated dereferencing of Links to particular pieces of segmented content.
- Automated signing and verification using both per-packet signing and support for aggregating signatures across multiple packets.
  - Library currently uses Merkle Hash Trees as its default aggregation technique
- Automated key retrieval and verification of all packets.
  - Packets that fail to verify are discarded.
- Mechanisms for hooking application-level trust decisions into the packet retrieval process, to ensure applications retrieve only packets they deem as trustworthy.
  - These latter pieces of functionality can be combined to implement a variety of trust models;

CCNx **C Library**

- support for segmentation/versioning and retrieval of latest versions of content
- per-packet signature generation, and verification of data signed either with per-packet signatures or with Merkle Hash Tree signature aggregation
- some metadata support (header retrieval)
- Most importantly, note that the C library has no facilities for encrypting or decrypting data encrypted by the Java library.
  - Some C-based CCNx applications, such as VoCCN, have application-level support for encryption.
  - It also does not include support for writing data to repositories, though it can read data from them

If you want to write some code...

Install eclipse IDE and import projects

- In Java: ccnChat and ccnFileProxy.
  - ccnFileProxy demonstrates filter registration, interest handling, and the use of the stream API.
  - ccnChat demonstrates the use of "network objects" -- versioned objects that use CCNx, rather than a database, as a backing store.
- In C: start with VoCCN
  - demonstrates basic use of the C library, and how to integrate CCNx with an existing C application.
  - amount of CCNx code is relatively small, and well localized.
  - It also uses a CCNx encapsulation of RTP that might be useful for other things.

Areas of Future Work

Project ideas

- Improving OS integration for better performance, e.g. through kernel drivers
- Content Routing
- Trust Management
- Apps Development
  - CCNWWW
  - ???

Conclusion

- CCNx technology is still at a ***very*** early stage of development, with pure infrastructure and no applications, best suited to researchers like us.
  - Project CCNx is an invitation to join and participate in this exploration of the frontier of content networking.
- Where to look for help
  - **Search and Subscribe CCNx Mailing lists**
    - **ccnx-dev** - core development discussion
    - **ccnx-users** - discussion of general development and use

References

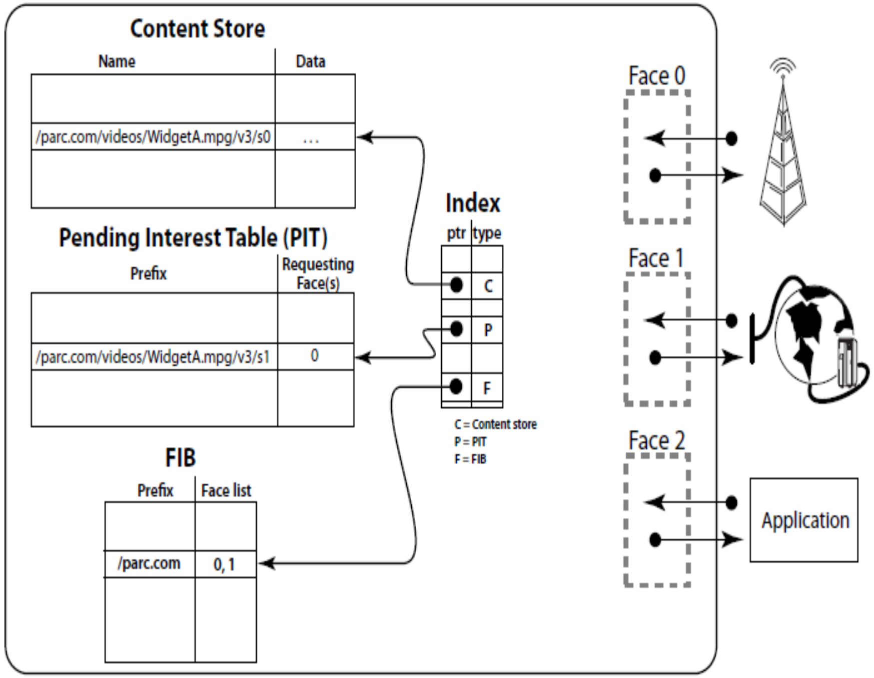
- http://www.ccnx.org
- http://www.ccnx.org/content/field-guide-ccnx-release
- http://blog.rungeek.com/post/1711470902/project-ccnx-how-to

Backup Slides

- README ,configure - master configure script.
- doc/ - documentation tree.
  - doc/technical - specifications ,doc/ccode - API documentation generated from C code ,doc/javacode - API documentation generated from Java code
- csrc/ - C code tree
  - csrc/include/ccn - C header files ,csrc/lib - C application library implementation ,csrc/ccnd - CCN daemon, the user-space forwarder implementation
  - csrc/libexec - connectivity utilities, especially ccndc, the ccnd configurator and connectivity agent ,csrc/cmd - simple command-line utilities
  - csrc/conf - OS-specific configuration scripts etc. ,csrc/contrib - third-party library needed for certain platforms that are POSIX-deficient
  - csrc/test - C test suite csrc/util - launch script support
- javasrc/ - Java code tree. The usual Java conventions are used for mapping package names to the file tree, with root package org.ccnx.ccn.
  - javasrc/src - Java source tree javasrc/lib - third-party libraries ,javasrc/tools - convenience scripts schema - XML schema and DTD files apps –
- experimental/sample apps tree
  - apps/ccnChat - simple text chat sample in Java ,apps/ccnFileProxy - simple proxy making local files available via CCNx
  - apps/vlc - vlc plugin for media transport experiments ,apps/wireshark - wireshark plugin dissector for decoding CCNx packets
  - experiments/multicast - scripts for running multi-machine experiments in content distribution performance over local multicast group.
- android/ - The Android implementaiton
  - android/external - External libraries needed for ARM ,android/CCNx-Android-Lib - A common Android Library for working with CCNx ,android/CCNx-Android-Services - Wrappers for ccnd and repository ,android/apps - Android applications that use CCNx
  - android/apps/CCNx-Android-Chat - The CCNx Chat application for Android

CCN forwarding Engine model

- *Interest* packets
  - specify the prefix of the name of the desired content
  - and a set of rules by which to determine what of the content under that prefix to return.



Content Store

Name	Data
/parc.com/videos/WidgetA.mpg/v3/s0	...

Pending Interest Table (PIT)

Prefix	Requesting Face(s)
/parc.com/videos/WidgetA.mpg/v3/s1	0

FIB

Prefix	Face list
/parc.com	0, 1

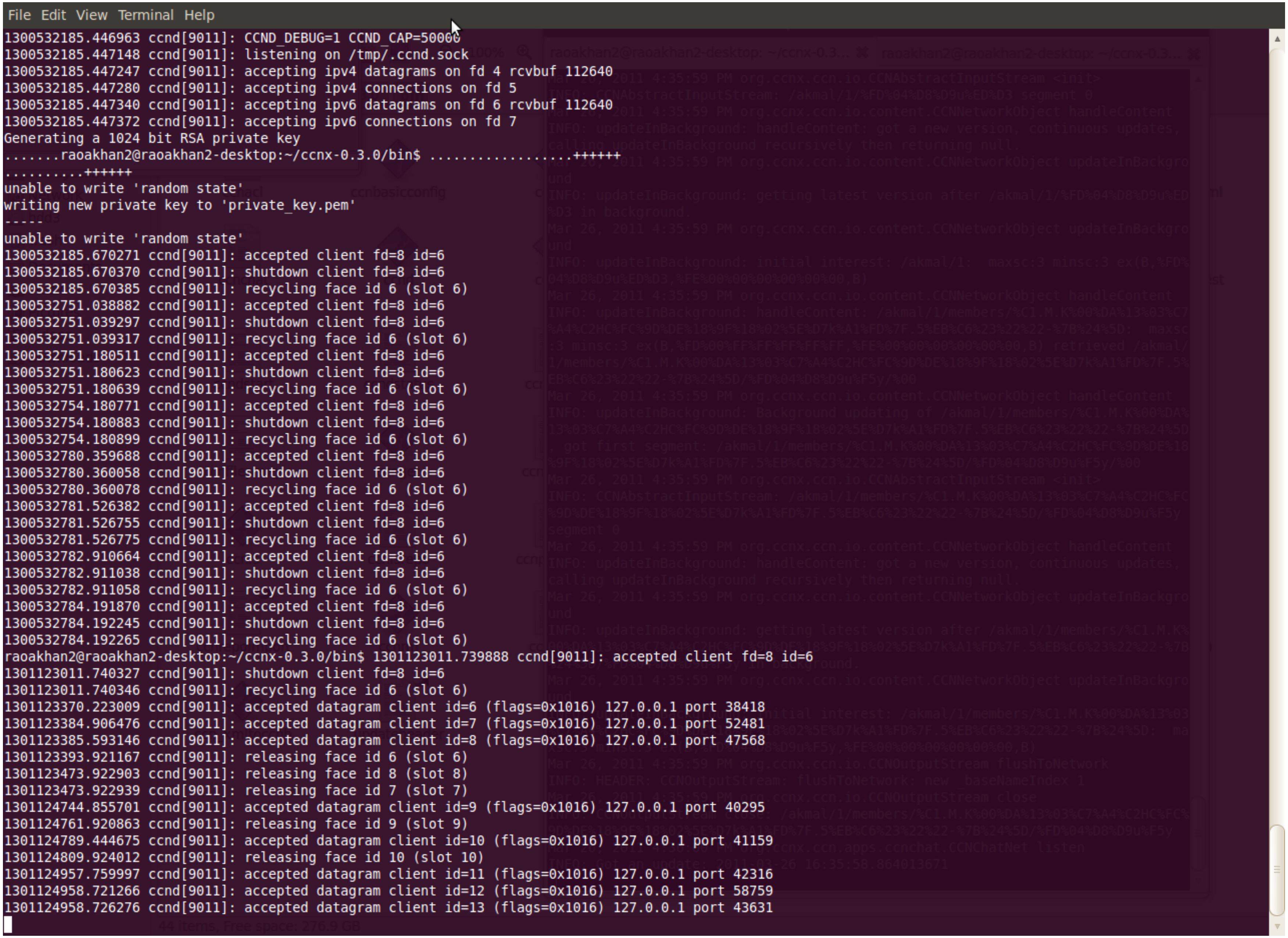
Index

ptr	type
●	C
●	P
●	F

- CCN does not require that data be published and registered with the infrastructure before it can be retrieved;
  - Interests and Data flow in lock-step, each Interest retrieving a single data packet.

Common to both Java and C library

- support for encoding/decoding *ccnb*, a compact binary encoding for XML. ccnb encoding is used for CCNx's messages on the wire, and is available for applications to tailor to their own use for stored and transmitted content. the core CCNx Interest-Data protocol, embodied in the following operations: asynchronous data retrieval API ("expressInterest"): registering an Interest, and calling back into the application when corresponding data is received. Both libraries take care of automatically reexpressing Interests if they time out without a response. Automated reexpression can be canceled.
- synchronous data retrieval API ("get"): presenting a blocking get method, where an application can express an Interest and block until data is returned or it times out
- asynchronous interest retrieval API: registering a *filter* causes ccnd to send the registeree interests that match that filter (rather than data); the registeree can then choose to send back or generate data in response
- support for automated key generation support for basic signing and verification (see details below)



```
1300532185.446963 ccnd[9011]: CCND_DEBUG=1 CCND_CAP=50000
1300532185.447148 ccnd[9011]: listening on /tmp/.ccnd.sock
1300532185.447247 ccnd[9011]: accepting ipv4 datagrams on fd 4 rcvbuf 112640
1300532185.447280 ccnd[9011]: accepting ipv4 connections on fd 5
1300532185.447340 ccnd[9011]: accepting ipv6 datagrams on fd 6 rcvbuf 112640
1300532185.447372 ccnd[9011]: accepting ipv6 connections on fd 7
Generating a 1024 bit RSA private key
.......raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$ .................++++++
..........++++++
unable to write 'random state'
writing new private key to 'private_key.pem'
-----
unable to write 'random state'
1300532185.670271 ccnd[9011]: accepted client fd=8 id=6
1300532185.670370 ccnd[9011]: shutdown client fd=8 id=6
1300532185.670385 ccnd[9011]: recycling face id 6 (slot 6)
1300532751.038882 ccnd[9011]: accepted client fd=8 id=6
1300532751.039297 ccnd[9011]: shutdown client fd=8 id=6
1300532751.039317 ccnd[9011]: recycling face id 6 (slot 6)
1300532751.180511 ccnd[9011]: accepted client fd=8 id=6
1300532751.180623 ccnd[9011]: shutdown client fd=8 id=6
1300532751.180639 ccnd[9011]: recycling face id 6 (slot 6)
1300532754.180771 ccnd[9011]: accepted client fd=8 id=6
1300532754.180883 ccnd[9011]: shutdown client fd=8 id=6
1300532754.180899 ccnd[9011]: recycling face id 6 (slot 6)
1300532780.359688 ccnd[9011]: accepted client fd=8 id=6
1300532780.360058 ccnd[9011]: shutdown client fd=8 id=6
1300532780.360078 ccnd[9011]: recycling face id 6 (slot 6)
1300532781.526382 ccnd[9011]: accepted client fd=8 id=6
1300532781.526755 ccnd[9011]: shutdown client fd=8 id=6
1300532781.526775 ccnd[9011]: recycling face id 6 (slot 6)
1300532782.910664 ccnd[9011]: accepted client fd=8 id=6
1300532782.911038 ccnd[9011]: shutdown client fd=8 id=6
1300532782.911058 ccnd[9011]: recycling face id 6 (slot 6)
1300532784.191870 ccnd[9011]: accepted client fd=8 id=6
1300532784.192245 ccnd[9011]: shutdown client fd=8 id=6
1300532784.192265 ccnd[9011]: recycling face id 6 (slot 6)
raoakhan2@raoakhan2-desktop:~/ccnx-0.3.0/bin$ 1301123011.739888 ccnd[9011]: accepted client fd=8 id=6
1301123011.740327 ccnd[9011]: shutdown client fd=8 id=6
1301123011.740346 ccnd[9011]: recycling face id 6 (slot 6)
1301123370.223009 ccnd[9011]: accepted datagram client id=6 (flags=0x1016) 127.0.0.1 port 38418
1301123384.906476 ccnd[9011]: accepted datagram client id=7 (flags=0x1016) 127.0.0.1 port 52481
1301123385.593146 ccnd[9011]: accepted datagram client id=8 (flags=0x1016) 127.0.0.1 port 47568
1301123393.921167 ccnd[9011]: releasing face id 6 (slot 6)
1301123473.922903 ccnd[9011]: releasing face id 8 (slot 8)
1301123473.922939 ccnd[9011]: releasing face id 7 (slot 7)
1301124744.855701 ccnd[9011]: accepted datagram client id=9 (flags=0x1016) 127.0.0.1 port 40295
1301124761.920863 ccnd[9011]: releasing face id 9 (slot 9)
1301124789.444675 ccnd[9011]: accepted datagram client id=10 (flags=0x1016) 127.0.0.1 port 41159
1301124809.924012 ccnd[9011]: releasing face id 10 (slot 10)
1301124957.759997 ccnd[9011]: accepted datagram client id=11 (flags=0x1016) 127.0.0.1 port 42316
1301124958.721266 ccnd[9011]: accepted datagram client id=12 (flags=0x1016) 127.0.0.1 port 58759
1301124958.726276 ccnd[9011]: accepted datagram client id=13 (flags=0x1016) 127.0.0.1 port 43631
```