

PingER Route Visualizer using Traceroutes

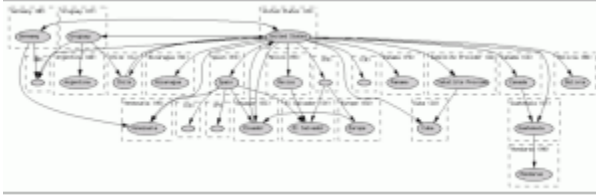
Purpose

This script generates topology graphs of traceroute data of a particular region/group of countries in the Guthrie database. You need to provide either the name of the traceroute server that you will use to do the traceroute or specify a file containing the raw traceroutes. In the absence of these arguments the program does traceroutes from the current machine.

Visualizing topology graphs can be very useful for seeing and demonstrating how direct or indirect routing is between countries (often routing between countries close to one another in developing regions can go via Europe or the US). It is also useful to discover hosts whose location is not properly known.

An example from running

245cottrell@pinger:~/bin/projects/topology/trunk/unit_tests>perl -d region_topology.pl is shown here:



Installation

In order to install the software from svn follow these steps:

- Create a new directory at the location where you want to install the software.

```
akbar@iepm-resp $ pwd
/afs/slac.stanford.edu/u/sg/akbar
akbar@iepm-resp $ mkdir myprojects
```

Now cd into the myprojects directory and download the iepm, topology and pinger projects using svn, by issuing the following commands:

- - `svn co file:///afs/slac.stanford.edu/g/scs/net/netmon/repo/svn/topology topology`
 - `svn co file:///afs/slac.stanford.edu/g/scs/net/netmon/repo/svn/iepm iepm`
- - `svn co file:///afs/slac.stanford.edu/g/scs/net/netmon/repo/svn/pinger pinger`

```
akbar@iepm-resp $ cd myprojects/
akbar@iepm-resp $ pwd
/afs/slac.stanford.edu/u/sg/akbar/myprojects
akbar@iepm-resp $ svn co file:///afs/slac.stanford.edu/g/scs/net/netmon/repo/svn/topology topology
A    topology/trunk
A    topology/trunk/unit_tests
A    topology/trunk/unit_tests/test_iepm-db-topology.pl
A    topology/trunk/unit_tests/regiontopology.conf
A    topology/trunk/unit_tests/test_canvas.pl
A    topology/trunk/unit_tests/test_topology.pl
A    topology/trunk/unit_tests/test_tools-tracerouteASN.pl
A    topology/trunk/unit_tests/lookup.txt
A    topology/trunk/unit_tests/test_canvasWithASS.pl
A    topology/trunk/unit_tests/region_topology.pl
A    topology/trunk/unit_tests/region_topology_wrapper.pl
A    topology/trunk/unit_tests/test_canvas-graphvizWithASN.pl
A    topology/trunk/unit_tests/test_canvas-graphviz.pl
A    topology/trunk/www
A    topology/trunk/www/html
A    topology/trunk/www/html/traceanal.css
A    topology/trunk/www/html/traceanal.js
A    topology/trunk/www/html/wz_tooltip.js
A    topology/trunk/doc
A    topology/trunk/install
A    topology/trunk/lib
A    topology/trunk/lib/IEPM
```

```

A topology/trunk/lib/IEPM/Topology
A topology/trunk/lib/IEPM/Topology/Node
A topology/trunk/lib/IEPM/Topology/Node/Graphviz.pm
A topology/trunk/lib/IEPM/Topology/Node/GraphvizCountry.pm
A topology/trunk/lib/IEPM/Topology/Canvas.pm
A topology/trunk/lib/IEPM/Topology/ASGroup.pm
A topology/trunk/lib/IEPM/Topology/Canvas
A topology/trunk/lib/IEPM/Topology/Canvas/Graphviz.pm
A topology/trunk/lib/IEPM/Topology/Canvas/GraphvizCountry.pm
A topology/trunk/lib/IEPM/Topology/CountryGroup.pm
A topology/trunk/lib/IEPM/Topology/Link
A topology/trunk/lib/IEPM/Topology/Link/Graphviz.pm
A topology/trunk/lib/IEPM/Topology/Link/GraphvizCountry.pm
A topology/trunk/lib/IEPM/Topology/Link/ASN.pm
A topology/trunk/lib/IEPM/Topology/Link/Country.pm
A topology/trunk/lib/IEPM/Topology/Traceroute
A topology/trunk/lib/IEPM/Topology/Traceroute/ASN
A topology/trunk/lib/IEPM/Topology/Traceroute/ASN/Graphviz.pm
A topology/trunk/lib/IEPM/Topology/Traceroute/Country
A topology/trunk/lib/IEPM/Topology/Traceroute/Country/GraphvizCountry.pm
A topology/trunk/lib/IEPM/Topology/Traceroute/Graphviz.pm
A topology/trunk/bin
A topology/trunk/bin/traceroute_topology.pl
A topology/trunk/bin/traceroute_analysis.pl
Checked out revision 552.
akbar@iepm-resp $ svn co file:///afs/slac.stanford.edu/g/scs/net/netmon/repo/svn/iepm iepm
A iepm/trunk
A iepm/trunk/unit_tests
A iepm/trunk/unit_tests/test_table.pl
A iepm/trunk/unit_tests/test_tools-traceroute.pl
A iepm/trunk/unit_tests/test_cgi-typechecking.pl
A iepm/trunk/unit_tests/test_ping.pl
A iepm/trunk/unit_tests/test_time.pl
A iepm/trunk/unit_tests/test_distribution.pl
A iepm/trunk/unit_tests/test_topology-node.pl
A iepm/trunk/unit_tests/test_tcpmon.pl
A iepm/trunk/unit_tests/test_topology-nodeAndLink.pl
A iepm/trunk/unit_tests/test_jitter.pl
A iepm/trunk/unit_tests/test_jitter_tcpmon.pl
A iepm/trunk/unit_tests/test_traceroute-comaprisson-metadata.pl
A iepm/trunk/unit_tests/test_traceanal-htmltable.pl
A iepm/trunk/doc
A iepm/trunk/doc/README.asn
A iepm/trunk/install
A iepm/trunk/install/Makefile.asn
A iepm/trunk/lib
A iepm/trunk/lib/IEPM
A iepm/trunk/lib/IEPM/Tools
A iepm/trunk/lib/IEPM/Tools/ASN
A iepm/trunk/lib/IEPM/Tools/ASN/WhoIs.pm
A iepm/trunk/lib/IEPM/Tools/Jitter
A iepm/trunk/lib/IEPM/Tools/Jitter/ITU.pm
A iepm/trunk/lib/IEPM/Tools/Jitter/IPDV.pm
A iepm/trunk/lib/IEPM/Tools/Table.pm
A iepm/trunk/lib/IEPM/Tools/Ping.pm
A iepm/trunk/lib/IEPM/Tools/Traceroute.pm
A iepm/trunk/lib/IEPM/Tools/Latency.pm
A iepm/trunk/lib/IEPM/Tools/TCPMon.pm
A iepm/trunk/lib/IEPM/Tools/ASN.pm
A iepm/trunk/lib/IEPM/Tools/Jitter.pm
A iepm/trunk/lib/IEPM/Tools/Traceroute
A iepm/trunk/lib/IEPM/Tools/Traceroute/ASN.pm
A iepm/trunk/lib/IEPM/Tools/Traceroute/Country.pm
A iepm/trunk/lib/IEPM/Tools/Country.pm
A iepm/trunk/lib/IEPM/CGI
A iepm/trunk/lib/IEPM/CGI/TypeChecking.pm
A iepm/trunk/lib/IEPM/Statistics
A iepm/trunk/lib/IEPM/Statistics/Distribution
A iepm/trunk/lib/IEPM/Statistics/Distribution/Cumulative.pm
A iepm/trunk/lib/IEPM/Statistics/Distribution/Latency.pm
A iepm/trunk/lib/IEPM/Statistics/Distribution/Cumulative

```

```

A   iepm/trunk/lib/IEPM/Statistics/Distribution/Cumulative/Latency.pm
A   iepm/trunk/lib/IEPM/Statistics/DistributionCumulative.pm
A   iepm/trunk/lib/IEPM/Statistics/DistributionCumulativeLatency.pm
A   iepm/trunk/lib/IEPM/Statistics/Distribution.pm
A   iepm/trunk/lib/IEPM/Statistics/DistributionLatency.pm
A   iepm/trunk/lib/IEPM/Stream
A   iepm/trunk/lib/IEPM/Stream/Tools
A   iepm/trunk/lib/IEPM/Stream/Tools/Traceroute
A   iepm/trunk/lib/IEPM/Stream/Tools/Traceroute/HTML.pm
A   iepm/trunk/lib/IEPM/Utilities
A   iepm/trunk/lib/IEPM/Utilities/Number.pm
A   iepm/trunk/lib/IEPM/Utilities/Dumper.pm
A   iepm/trunk/lib/IEPM/Utilities/Time.pm
A   iepm/trunk/lib/IEPM/Utilities/TimeConvert.pm
A   iepm/trunk/lib/IEPM/Utilities/TimeInterval.pm
A   iepm/trunk/lib/IEPM/Topology
A   iepm/trunk/lib/IEPM/Topology/Node.pm
A   iepm/trunk/lib/IEPM/Topology/SubnetChange.pm
A   iepm/trunk/lib/IEPM/Topology/TracerouteComparisonMetaData.pm
A   iepm/trunk/lib/IEPM/Topology/Link.pm
A   iepm/trunk/lib/IEPM/Topology/TracerouteMetaData.pm
A   iepm/trunk/lib/IEPM/Topology/Traceanal
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Row.pm
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Table.pm
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Element.pm
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Table
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Table/HTML.pm
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Element
A   iepm/trunk/lib/IEPM/Topology/Traceanal/Element/Symbolization.pm
A   iepm/trunk/lib/IEPM/CGI.pm
A   iepm/trunk/lib/IEPM/Plot
A   iepm/trunk/lib/IEPM/Plot/CanvasGnuplot.pm
A   iepm/trunk/lib/IEPM/Plot/Data.pm
A   iepm/trunk/lib/IEPM/Plot/Canvas.pm
A   iepm/trunk/lib/IEPM/Plot/DataGnuplot.pm
A   iepm/trunk/bin
A   iepm/trunk/bin/asn.pl
Checked out revision 552.
akbar@iepm-resp $ svn co file:///afs/slac.stanford.edu/g/scs/net/netmon/repo/svn/pinger pinger
A   pinger/trunk
A   pinger/trunk/unit_tests
A   pinger/trunk/lib
A   pinger/trunk/lib/IEPM
A   pinger/trunk/bin
A   pinger/trunk/bin/pingtable.pl
A   pinger/trunk/bin/HostSearcher.pl
A   pinger/trunk/bin/scriptdoc.pl
A   pinger/trunk/bin/prmout_filter.pl
A   pinger/trunk/bin/country_search_list.conf
A   pinger/trunk/bin/script_list.txt
Checked out revision 552.
akbar@iepm-resp $

```

Running the Program

- First you need to cd into the \$HOME/myprojects/topology/trunk/unit_tests/. I am assuming that you created the myprojects in your home directory. In case there was another location, replace \$HOME with that location.
- Next set the PERL5LIB environment variable for the system libraries. Once again if the location of your myprojects directory is different replace \$HOME with that location.
 - setenv PERL5LIB \$HOME/myprojects/topology/trunk/lib:\$HOME/myprojects/iepm/trunk/lib
- Finally you can run the program by simply running the script region_topology.pl

```
akbar@iepm-resp $ ls -rtl
total 82
-rwxr-xr-x  1 akbar  sg      10212 Jul  2 15:55 test_topology.pl*
-rwxr-xr-x  1 akbar  sg      5828  Jul  2 15:55 test_tools-tracerouteASN.pl*
-rwxr-xr-x  1 akbar  sg      5201  Jul  2 15:55 test_iepm-db-topology.pl*
-rw-r--r--  1 akbar  sg      9598  Jul  2 15:55 test_canvasWithASs.pl
-rw-r--r--  1 akbar  sg      9602  Jul  2 15:55 test_canvas.pl
-rw-r--r--  1 akbar  sg      9552  Jul  2 15:55 test_canvas-graphvizWithASN.pl
-rwxr-xr-x  1 akbar  sg      6557  Jul  2 15:55 test_canvas-graphviz.pl*
-rw-r--r--  1 akbar  sg      1515  Jul  2 15:55 regiontopology.conf
-rwxr-xr-x  1 akbar  sg      4908  Jul  2 15:55 region_topology_wrapper.pl*
-rwxr-xr-x  1 akbar  sg     14552  Jul  2 15:55 region_topology.pl*
-rw-r--r--  1 akbar  sg       155  Jul  2 15:55 lookup.txt
akbar@iepm-resp $ region_topology.pl
```

Usage:

```
region_topology.pl --region |--group |--country |--file |--hostfilelist --detail [--
tracerouteserver] [--colorlinks] [--endnodes]
```

Arguments:

--region	Name of region in PingER to which the traceroutes must be done for generating graph
--group	Name of group in PingER to which the traceroutes must be done for generating graph
--country	Name of country in PingER to which the traceroutes must be done for generating graph
--file	Path of file containing a list of raw traceroutes, which to use for drawing the graph
--hostfilelist	Path of file containing a list of hosts to do traceroutes, for drawing the graph
--detail	Detail level of the graph (values include 'asn' or 'country' or 'all' and is mandatory)
--tracerouteserver	Complete address of traceroute server from which to do traceroutes. Default is localhost
--colorlinks	Whether to color links according to traceroute ('on' or 'off')
--endnodes	Whether to display end nodes in the graph ('on' or 'off')
--basepath	Base path where to generate the graphs
--from	Name of the site containing the tracerouteserver (optional, e.g. SLAC)

Examples:

```
region_topology.pl --group 'SUB.SAHARA' --detail 'country'
region_topology.pl --region 'Latin America' --detail 'all'
region_topology.pl --file 'rawtraceroutes.txt' --detail 'all'
region_topology.pl --region 'Latin America' --detail 'all' --tracerouteserver 'http://www.tenet.
ac.za/cgi-bin/traceroute.pl'
region_topology.pl --hostfilelist 'beacons.txt' --detail 'country'
region_topology.pl --basepath '/afs/slac/u/sg/akbar/graphs'
```

If you run the script with --help or without any options it gives you the usage. The following options need to be provided:

- The target nodes for the topology graph may be specified to the program in a variety of ways. Only one of the following must be specified.
 - A region in PingER which case you must provide the **--region** switch followed by the exact name of the region in PingER.
 - A country in PingER in which case you must provide the **--country** switch followed by the exact name of the country in PingER.
 - A file containing a list of target host names followed by ip addresses seperated by commas (e.g. beacons.txt). This is specified using the **--hostfilelist** switch.
 - A file containing a list of raw traceroutes to a set of target nodes. This can be provided by the **--file** switch followed by the path of the file containing the traceroutes. This option is provided because the program automatically saves raw traceroutes in a text file and in case the graph is to be redrawn, the raw traceroutes can be provided to the program so that the time consuming task of actually performing the traceroutes can be avoided.
- The source node for the graph is by default the current machine. In case you want to change that you must specify the complete address of the traceroute server using the switch **--tracerouteserver**. An example is given in the usage above. In case you are using a tracerouteserver you should also provide the name of the tracerouteserver (e.g. TENET) using the **--from** switch. This helps in naming the graph files.
- The detail of the graph i.e. whether to show the intermediates nodes or only list the countries must be provided. It can be specified using the **--detail** switch followed by either 'country' or 'all'
- The **--endnodes** switch is optional and indicates whether you want to draw the endnodes or not. It has values of 'on' or 'off' and the default value is 'off'
- The **--basepath** switch is also optional and is used for providing the directory where you want to generate the graphs and rawtraceroute files. The default is the current directory. The file will have the name \$basepath/graph-for-\$file where \$file is the name associated with region|group|country.

To assist in debugging problems the script also produces an annotated file:

- In case you specify --region, --country or --group the name of the file generated will be annotated traceroutes-\$FROM-to-\$REGIONNAME.txt. Here the \$REGIONNAME is the name of the region, country or group in pinger that you specified and \$FROM will have a value if you specify it using --from field. The same will be the case for graphs and rawtraceroutes.
- In case you specify --file or --hostfilelist the name of the file generated will be annotated traceroutes-\$FROM-to-\$FILENAME.txt. Here the \$FILENAME is the name of the file specified in the --file or --hostfilelist and \$FROM once again has a value if you specify the --from field.
- The annotated traceroute currently works if all the nodes are in the linked list of objects. If you specify --detail as 'country' then the objects are collapsed to only country objects and the relevant information is not there.

Creating the gif file from the dot file

To create the gif file from the dot file use

```
36cottrell@pinger:~/bin/projects/topology/trunk/unit_tests>dot -Tgif -o ~cottrell/public_html/rviz.gif tempdot-SaoPaolo-to-Latin_merica.txt
Error: Could not find/open font : Times in /usr/X11R6/lib/X11/fonts/TrueType:/usr/X11R6/lib/X11/fonts/truetype:/usr/X11R6/lib/X11/fonts/TTF:/usr/share/fonts/TrueType:/usr/share/fonts/truetype:/usr/openwin/lib/X11/fonts/TrueType:/usr/X11R6/lib/X11/fonts/Type1:/usr/common/share/fonts/ttf
```

This will read input file tempdot-SaoPaolo-to-Latin_merica.txt and create output gif file ~cottrell/public_html/rviz.gif. Ignore the error message. In the above case you will be able to view the output at <http://www.slac.stanford.edu/~cottrell/rviz.gif>

Output

By default the topology gif file is placed in the current working directory with the name graph-<from>to<region|group|country>.gif. Where <from> is the value in the --from option, <region|group|country> is the value in the --region | --group or --country option. For example if called with:

```
region_topology.pl --region Africa --detail country --endnodes on --tracerouteserver 'http://www.unikin.cd/cgi-bin/traceroute.pl' --from 'DRC'
```

then the topology gif file has the name: graph-DRC-to-Africa.gif. If the --basepath option is specified then rather than placing the file in the current working directory it is placed in the directory specified by basepath. For example if called with:

```
region_topology.pl --region Africa --detail country --tracerouteserver 'http://www.unikin.cd/cgi-bin/traceroute.pl' --from 'DRC' --basepath '~cottrell/public_html'
```

then the file will appear at ~cottrell/public_html/graph-DRC-to-Africa.gif. The output of this file is a graph with end nodes in rectangles, routers in ellipses and directed edges between the nodes.

It also creates three other files (for example):

1. tempdot-DRC-to-Africa.txt, this is the dot file that provides the node relationships. It is of the form:

```

digraph G {
    nodesep = 0.10;
    ranksep = 0.25;
    fontsize = 9;
    node [style=filled,height=0.15,width=0.3,fontsize=9];
    subgraph cluster_ {
        label = " ("";
        style = "dashed";
        n0xa567b44 [label=""];
    }
    subgraph cluster_NI {
        label = "Nicaragua (NI)";
        style = "dashed";
        nNI [label="Nicaragua"];
    }
    subgraph cluster_EC {
        label = "Ecuador (EC)";
        style = "dashed";
        nEC [label="Ecuador"];
        n200_31_30_47 [label="gye.impsat.net.ec", shape="polygon", sides="4"];
        n200_125_133_7 [label="* www.gastromonde.edu.ec", shape="polygon", sides="4"];
        n201_217_87_3 [label="3.201-217-87.uio.satnet.net", shape="polygon", sides="4"];
    }
    }... nBR -> n143_108_25_100;
    nBR -> n143_108_31_3;
    nBR -> nUS;
    nUS -> nUY;...}

```

2. rawtraceroutes-DRC-to-Africa.txt, this is the raw traceroute data measu4red, for example a series of traceroutes of the form:

```

%%
%Traceroute No 1
%%
traceroute to 212.88.97.44 (212.88.97.44), 30 hops max, 38 byte packets
 1  gate (172.16.1.1) 0.224 ms 0.187 ms 0.114 ms
 2  213.255.195.225 (213.255.195.225) 9.986 ms 1.067 ms 0.985 ms
 3  FE8-0-0.gw1.dcm.sky-vision.net (217.194.158.12) 563.816 ms 710.425 ms 760.908 ms
 4  gw-uk.sky-vision.net (217.194.158.210) 570.907 ms 569.249 ms 551.291 ms
 5  skyvision-01692-ldn-b2.c.telia.net (213.248.75.202) 653.103 ms 661.445 ms 644.227 ms
 6  * ldn-b2-geth10-0-336.telia.net (213.248.75.201) 647.579 ms *
 7  ldn-b2-link.telia.net (80.91.250.213) 645.400 ms 630.215 ms 640.610 ms
 8  ldn-bb2-link.telia.net (80.91.250.229) 650.851 ms 657.959 ms 660.839 ms
 9  ldn-b4-link.telia.net (80.91.254.22) 672.093 ms 639.208 ms 640.740 ms
10  teleglobe-114869-ldn-b4.telia.net (213.248.74.2) 640.735 ms 670.570 ms 637.608 ms
11  195.219.195.13 (195.219.195.13) 645.608 ms 680.312 ms 690.821 ms
    MPLS Label=2789 CoS=5 TTL=1 S=0
12  195.219.195.10 (195.219.195.10) 650.858 ms 672.188 ms
__END__

```

3. annotatedtraceroutes-DRC-to-Africa.txt, which contain annotation information from the individual traceroutes, for example:

```

%%Traceroute No. 1
%%
COUNTRY's: (total 4)
-----
Country:
Name:
-----
1. gate (172.16.1.1)
-----
Country:      CG
Name:   Congo
-----
2. 213.255.195.225 (213.255.195.225)
-----
Country:      GB
Name:   United Kingdom
-----
3. FE8-0-0.gw1.dcm.sky-vision.net (217.194.158.12)
4. gw-uk.sky-vision.net (217.194.158.210)
5. 195.219.195.13 (195.219.195.13)
6. 195.219.195.10 (195.219.195.10)
-----
Country:      EU
Name:   Europe
-----
7. skyvision-01692-ldn-b2.c.telia.net (213.248.75.202)
8. * ldn-b2-geth10-0-336.telia.net (213.248.75.201)
9. ldn-b2-link.telia.net (80.91.250.213)
10. ldn-bb2-link.telia.net (80.91.250.229)
11. ldn-b4-link.telia.net (80.91.254.22)
12. teleglobe-114869-ldn-b4.telia.net (213.248.74.2)
-----
ACTUAL TARGET NODE = 212.88.97.44 , TARGET COUNTRY = UG
-----
COUNTRYLINKS: (total 4)
NAME:    -> NAME:    CG
NAME:    CG -> NAME:    GB
NAME:    GB -> NAME:    EU
NAME:    EU -> NAME:    GB

```

Problems

Accurately locating the routers can be difficult. We use [Geo IP Tool](#) as the default. We also looked carefully at [hostip.info](#), however it provides locations for fewer hosts and is not more accurate for our purposes. Both systems use databases and are not very complete/accurate for router interfaces. We are also developing [TULIP](#) to provide Geolocation using ping RTTs. It is a very valuable check on suspicious locations provided by the database methods. When we find the locations of hosts for which database methods provide the wrong countries, we provide a simple [lookup table](#) with the preferred countries.

Improvements/Bugs etc.

1. Correct traceroute number in raw file, currently it is always 1 [Done 8/30/07].
2. Correct nodes (routers mainly) identified like n0x9435b80, tried deleting in dot file but more complex (9/15/07)
3. Some routers identified by ip address but should be name
4. Fix the legend or whatever it is [Done 8/30/07]
5. Many end countries have no node [Done 8/30/07]
6. Add unique traceroute number to end node
7. Some end nodes are ellipses, should only be rectangles: Akbar
8. For Africa there are some end countries such as CH, NO that are not in Africa [Done 8/30/07]
9. Can we include the RTT in the end node [Drop probably not very useful and a lot of work?]
10. Can we color the links by the RTT [Drop 8/30/07]?
11. Some countries (e.g. SN for Africa) have no name
12. Why for Kinshasa router server in DRC (CD) is it labelled CG (Congo) [Understood]?
13. Add as a comment the landmark, userid, executing host, name & path of executing script, date to raw and annotated traceroute data (Les is working on 9/1/07): Les
14. Need to allow replacement with Hostip.info see for example <http://www.maxmind.com/app/geolitecountry> and <http://www.maxmind.com/app/geolitecity> (Akbar will do 8/9/07): Akbar.
15. Look at how to disambiguate horizontal lines:
 - a. Write script to remove the subgraph clusters while retaining the country names.

- b. Play with the ranksep (which increases the vertical separation)
 - c. Randomly color edges: Akbar
16. Need to update the documentation,
- a. ~~Indicate how to interpret the topology graph [Done Les 9/1/07]~~
 - b. ~~The information on debugging seems to need attention [Done Les 9/15/07].~~
 - c. ~~Reduce the width needed [Done Les 8/23/07]~~
17. ~~Does not correctly show end country, in the example below the route is identified from BR to UY but it should be BR to PE. Since the traceroute terminates before Peru it does not find the end country [Done 8/30/07].~~

```
traceroute to 200.37.45.39 (200.37.45.39), 30 hops max, 38 byte packets
 1  200.143.197.129 (200.143.197.129)  0.892 ms  0.783 ms  0.759 ms
 2  ge-1-0-0-r1-rj.bkb.rnp.br (200.143.252.181)  3.058 ms  0.590 ms  0.706 ms
 3  so-0-1-0-r1-sp.bkb.rnp.br (200.143.252.21)  6.326 ms  6.250 ms  21.632 ms
 4  rnp-whren-stm.ampath.net (198.32.252.221)  146.245 ms  143.362 ms  143.321 ms
 5  nlr.ampath.net (198.32.252.161)  177.944 ms  178.762 ms  181.123 ms
 6  216.24.184.90 (216.24.184.90)  341.576 ms  341.484 ms  342.770 ms
 7  saopaulo-buenosaires.core.redclara.net (200.0.204.37)  369.670 ms  369.690 ms  370.923 ms
 8  buenosaires-santiago.core.redclara.net (200.0.204.29)  389.121 ms  389.505 ms  389.805 ms
 9  raap-pe-sant.peer.redclara.net (200.0.204.158)  480.197 ms  484.125 ms  479.740 ms 10  * * *
11  * * *
12  *
__END__
```