



Anycast and BGP Stability

A Closer Look at DNSMON Data

Daniel.Karrenberg@ripe.net



Other Studies

- Mark Kusters:
"Life and Times of J-Root"
<http://www.nanog.org/mtg-0410/pdf/kusters.pdf>
- Peter Boothe, Randy Bush:
"DNS Anycast Stability, Some Early Results"
<http://rip.psg.com/~randy/050223.anycast-apnic.pdf>
- These observe less stability than we expected.
- Let's take a closer look at DNSMON data !



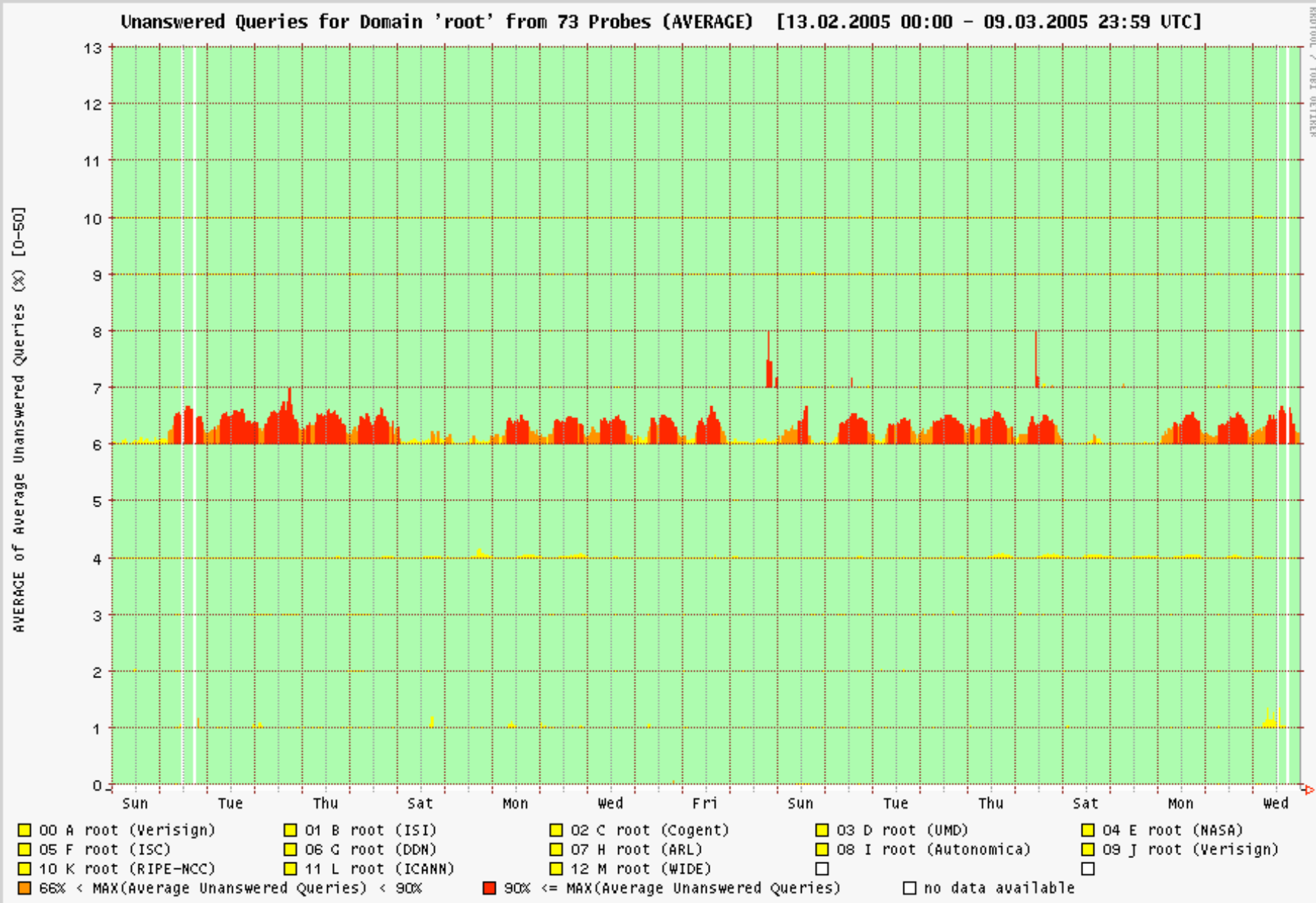
DNSMON

- DNS Server Monitoring
 - Data since April 2003
 - id.server / hostname.bind every 60s (poisson)
 - Less frequently: SOA, version.bind / version.server
 - Loss and delay @ <http://dnsmon.ripe.net>
- DNSMON and Anycasting
 - Measures service independent of anycasting
 - We looked at ID switches in raw data from time-to-time
 - Found nothing peculiar, thus did not look more closely



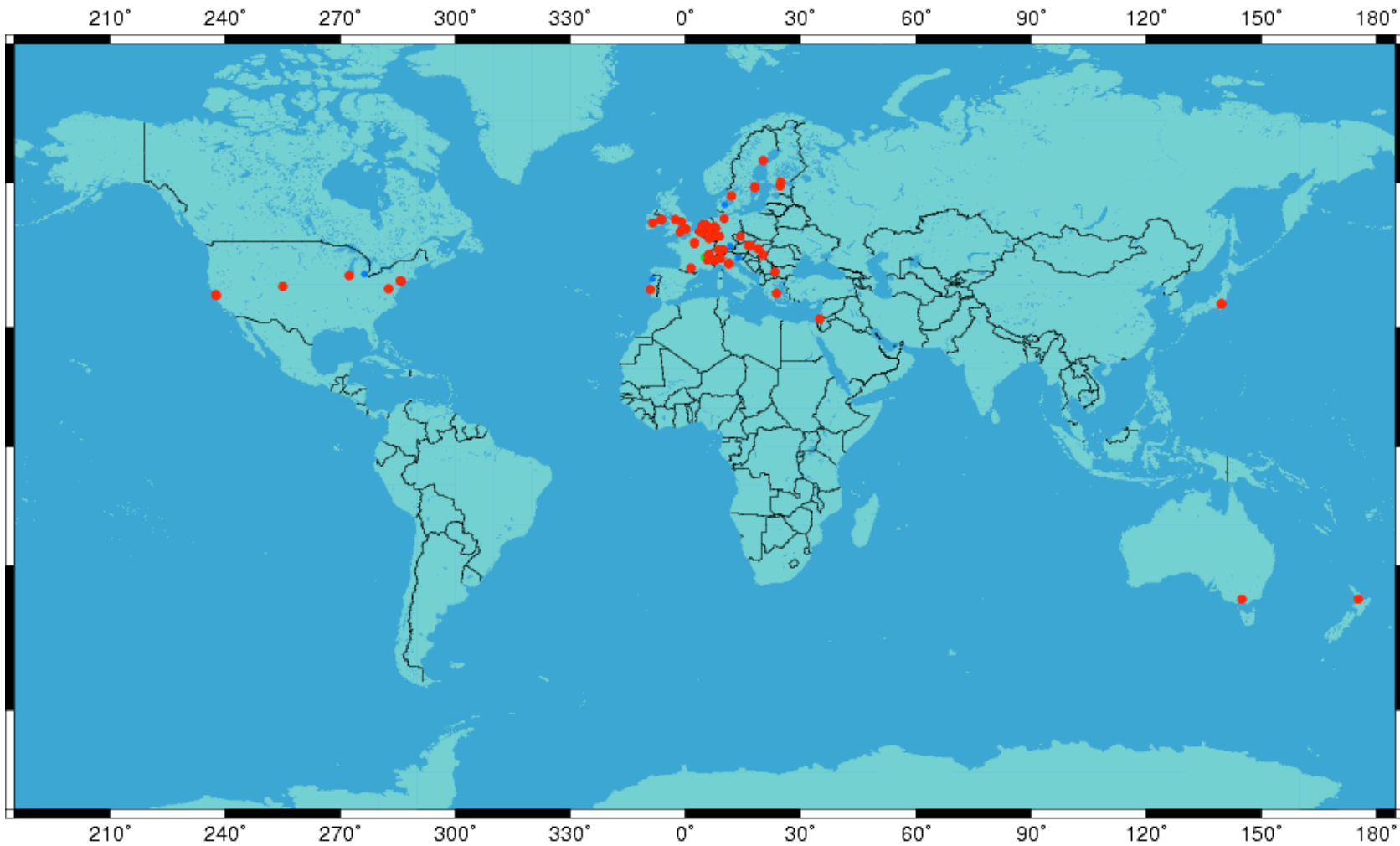
The Data Set

- DNSMON Server ID Queries
 - Sun Feb 13 00:00:00 - Wed Mar 9 23:59:59 2005 UTC
 - Chosen because most readily available
 - Probes
 - 77 test-boxes
 - in 59 BGP ASes
 - Targets
 - Anycasted DNS Root Servers : C, F, I, J, K, M
 - Volume
 - ~ 35k queries per probe per host
 - ~ 16M data points





Probe Locations: Biased





Data Set Example

Probe	UTC	Target	RTT	Try	Query	Answer
tt68.ripe.net	1108267231	j.root-servers.net	0.004215	1	hostname.bind	jns2-zgtld.j.root-servers.net
tt68.ripe.net	1108267244	f.root-servers.net	0.007993	1	hostname.bind	sfo2b.f.root-servers.org
tt68.ripe.net	1108267259	k.root-servers.net	0.131091	1	id.server	k1.linx
tt68.ripe.net	1108267266	c.root-servers.net	0.022524	1	hostname.bind	lax1a.c.root-servers.org
tt68.ripe.net	1108267276	i.root-servers.net	0.101645	1	hostname.bind	s1.tok
tt68.ripe.net	1108267281	m.root-servers.net	0.101400	1	hostname.bind	M-n4
tt68.ripe.net	1108267291	j.root-servers.net	0.004008	1	hostname.bind	jns3-zgtld.j.root-servers.net
tt68.ripe.net	1108267298	f.root-servers.net	0.008080	1	hostname.bind	sfo2b.f.root-servers.org
tt68.ripe.net	1108267319	m.root-servers.net	0.101348	1	hostname.bind	M-n4
tt68.ripe.net	1108267326	k.root-servers.net	0.131029	1	id.server	k1.linx



Data Reduction

- Extract ID changes
- Eliminate inconsistent ID naming
- Eliminate intra-instance switches
 - Same location, different box
 - Done for C, F, J

```
s/([^.]+)\...*/\1/ if $t eq 'c.root-servers.net';  
s/([0-9])\...*/\1x/ if $t eq 'f.root-servers.net';  
s/jns.-([^.]+)\...*/jnsx-\1/ if $t eq 'j.root-servers.net';
```




Reduced Data Set

UTC time, rec-type, probe, AS, target, #query, #switch, #loss, #unkonwn, delta-t switch, avg RTT, lds

UTC time	rec-type	probe	AS	target	#query	#switch	#loss	#unkonwn	delta-t	switch	avg RTT	lds
Feb 13 00:00:02	A	tt111	AS8422	f	0	0	0	0	0	0	0.000000	sfo2x
Feb 17 13:20:49	*	tt111	AS8422	f	6482	1	2	0	393647	0.167233	sfo2x	-> paolx
Feb 17 13:21:41	*	tt111	AS8422	f	6483	2	2	0	52	0.170097	paolx	-> sfo2x
Feb 17 13:33:26	*	tt111	AS8422	f	6495	3	2	0	705	0.169988	sfo2x	-> paolx
Feb 17 13:37:38	*	tt111	AS8422	f	6499	4	2	0	252	0.165463	paolx	-> sfo2x
Mar 9 12:12:58	*	tt111	AS8422	f	34824	5	12	0	1722920	0.165365	sfo2x	-> paolx
Mar 9 13:00:34	*	tt111	AS8422	f	34871	6	13	0	2856	0.169342	paolx	-> sfo2x
Mar 9 23:58:37	Z	tt111	AS8422	f	35524	6	13	0	39483	0.166880	sfo2x	
					sfo2x(4) paolx(3)		switches:6(0.02%)		hours:3(0.50%)			



Global Results

16,327,366 queries

14,124 drops (0.087%)

99.9% Happy Packets !

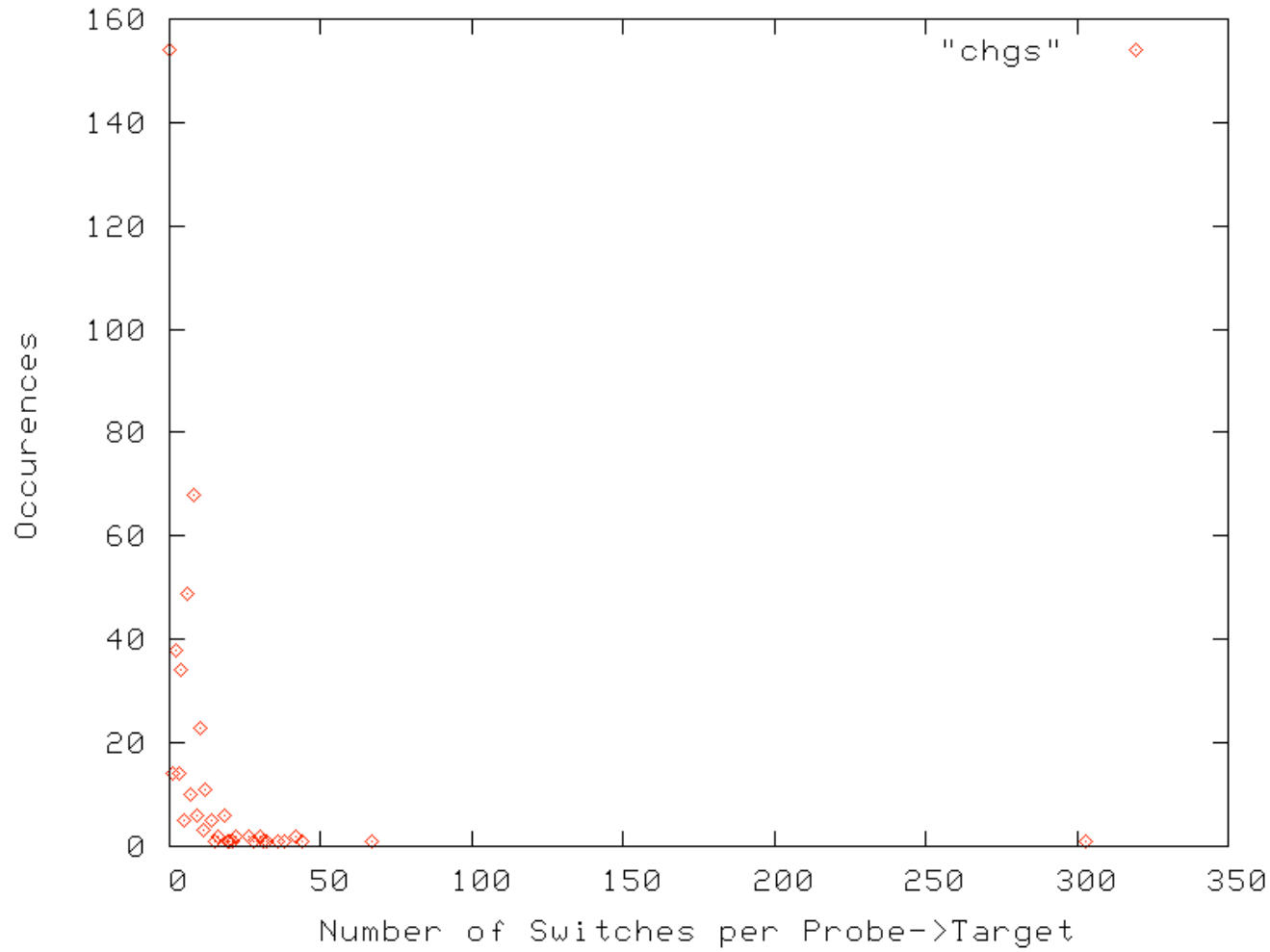
2,754 switches (0.017%)

~ 1 in 10000 packets

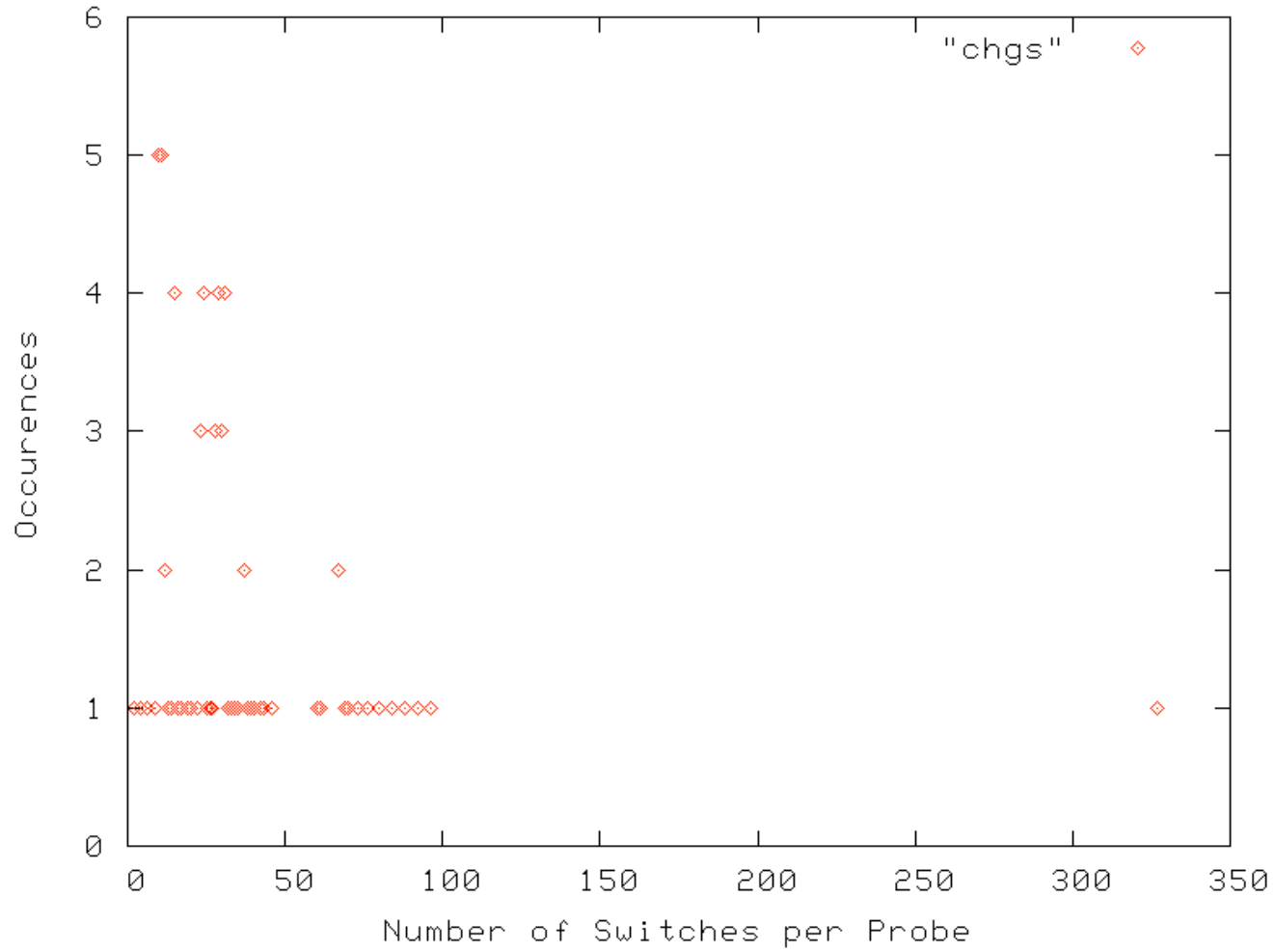
0 unknown IDs

- File available on request
- Regular report on dnsmon site possible

Distribution of the Switches



Switches per Probe





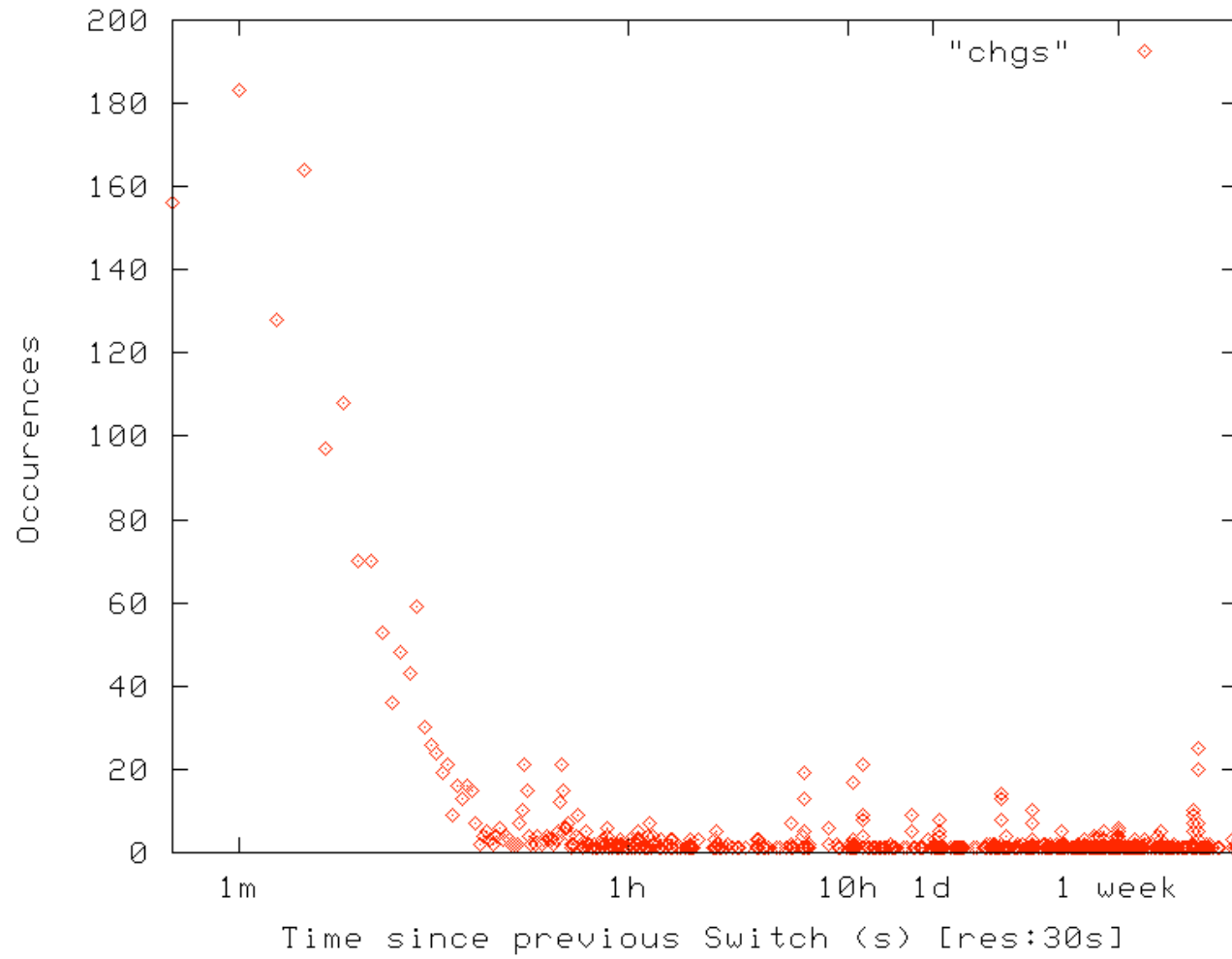
First Impression

- Agrees with cursory observations
- A little badness, no real ugliness or worse
- No real hope for the results others see
 - Not enough switches per probe->target
 - Resolution is lower

- But hang on, if we lump it all together ...



Time Since Previous Switch for the Whole Dataset





Something Worth Looking At

- There must be periods of high switching activity
- Agrees with other studies
- Not as bad as they see, why?
- Location of DNSMON probes biased
 - Within ISP infrastructure
 - “Quality conscious ISPs” more likely to have probes
 - Not on the network fringes
- Let’s look at the badness and correlate with BGP
 - <http://www.ris.ripe.net/bgplay/>



AS1853 to k.root-servers.net

Summary:

Mar 9 23:59:55 35530 queries 36 switches 8 drops

k2.ams-ix(19) k2.linx(18)

switches:36(0.10%) "errored hours":8(1.33%)



AS1853 to k.root-servers.net

```
Feb 22 04:39:43 13065 1 0 794376 0.034115 k2.ams-ix -> k2.linx
Feb 22 04:49:06 13074 2 0 563 0.032744 k2.linx -> k2.ams-ix
Feb 22 04:53:07 13078 3 0 241 0.023697 k2.ams-ix -> k2.linx
Feb 22 05:03:10 13088 4 0 603 0.032999 k2.linx -> k2.ams-ix
Feb 22 05:04:13 13089 5 0 63 0.034307 k2.ams-ix -> k2.linx
Feb 22 05:13:17 13098 6 0 544 0.032885 k2.linx -> k2.ams-ix
Feb 22 05:15:21 13100 7 1 124 0.034429 k2.ams-ix -> k2.linx
Feb 22 05:18:23 13103 8 1 182 0.029720 k2.linx -> k2.ams-ix
Feb 22 05:20:49 13105 9 1 146 0.027436 k2.ams-ix -> k2.linx
Feb 22 05:25:50 13110 10 1 301 0.031518 k2.linx -> k2.ams-ix
Feb 22 05:27:48 13112 11 1 118 0.027449 k2.ams-ix -> k2.linx
Feb 22 05:31:31 13116 12 1 223 0.030807 k2.linx -> k2.ams-ix
Feb 22 05:33:24 13118 13 2 113 0.034222 k2.ams-ix -> k2.linx
Feb 22 05:36:57 13122 14 2 213 0.031018 k2.linx -> k2.ams-ix
Feb 22 05:37:58 13123 15 2 61 0.034384 k2.ams-ix -> k2.linx

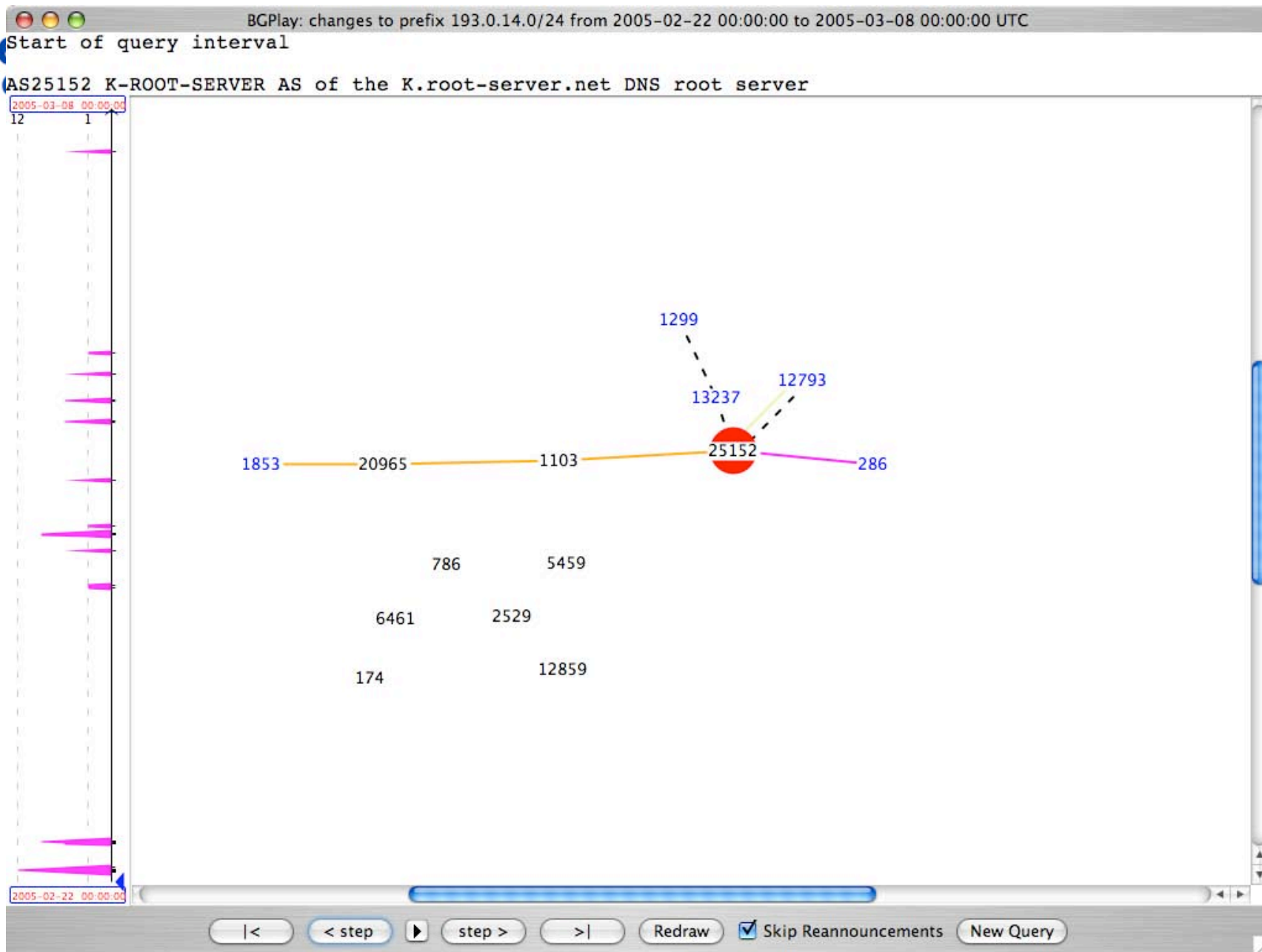
Feb 22 06:56:04 13201 16 2 4686 0.034235 k2.linx -> k2.ams-ix

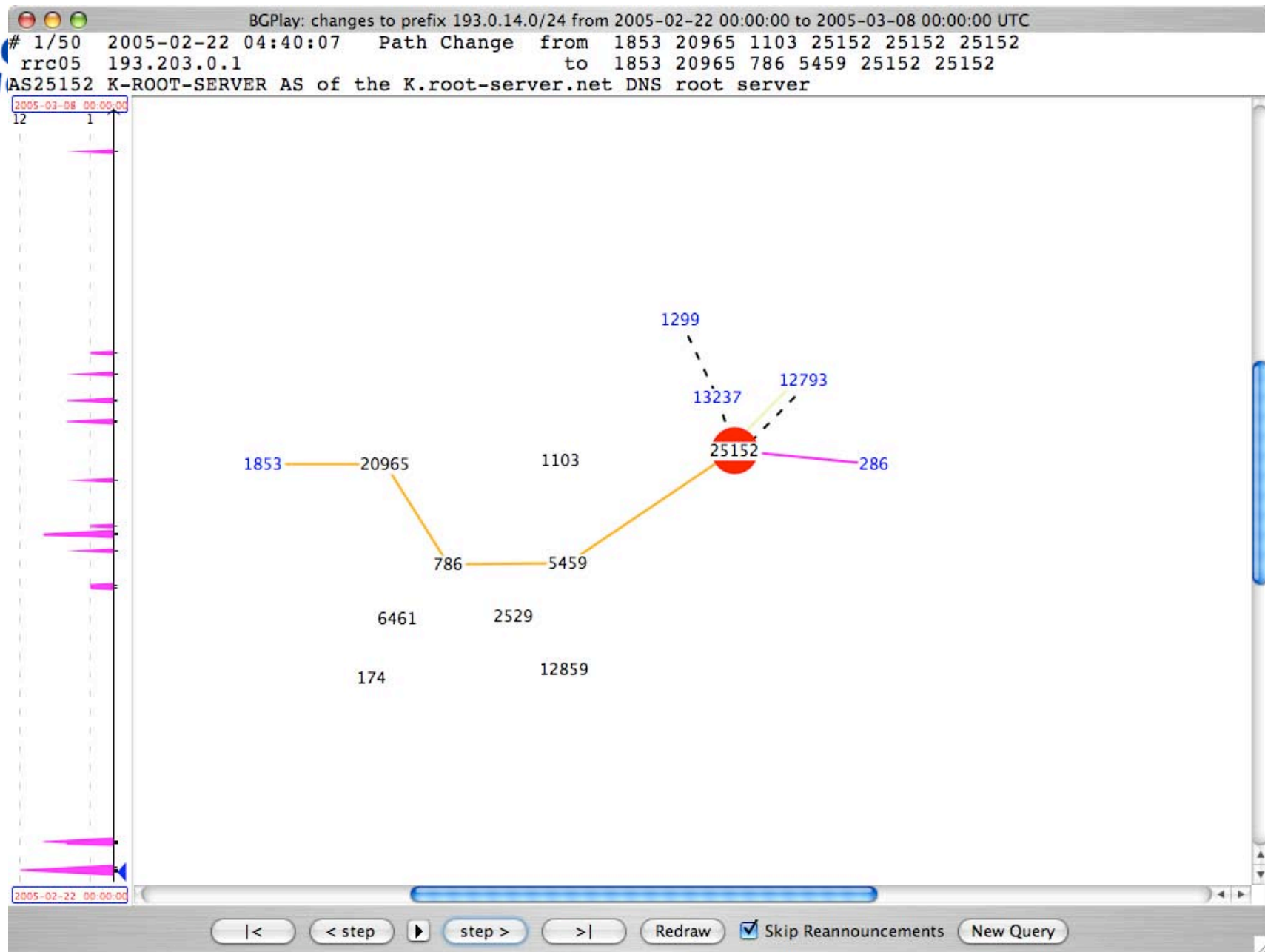
Feb 22 17:03:54 13800 17 2 36470 0.020362 k2.ams-ix -> k2.linx
Feb 22 17:04:52 13801 18 2 58 0.020316 k2.linx -> k2.ams-ix
```

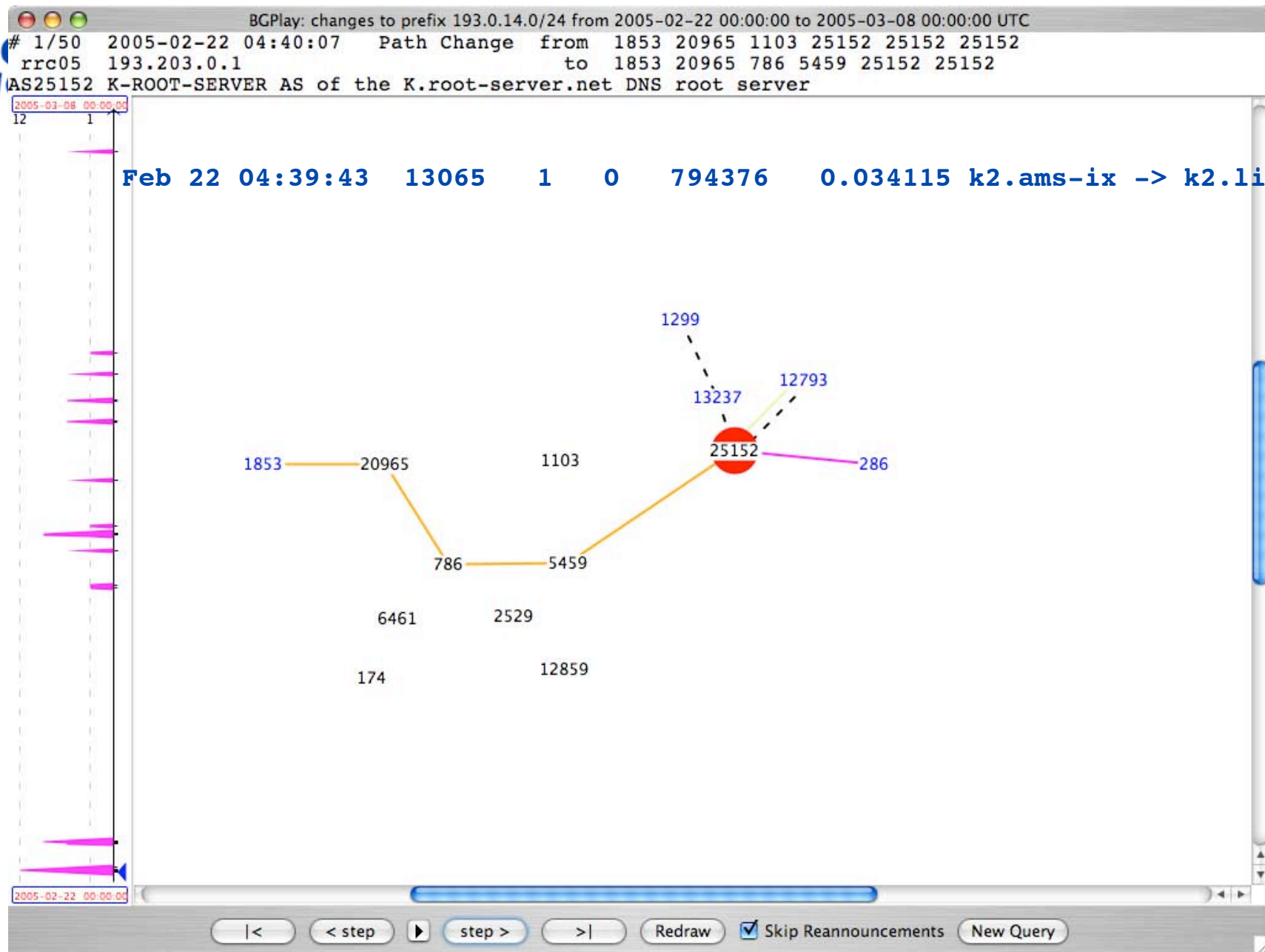


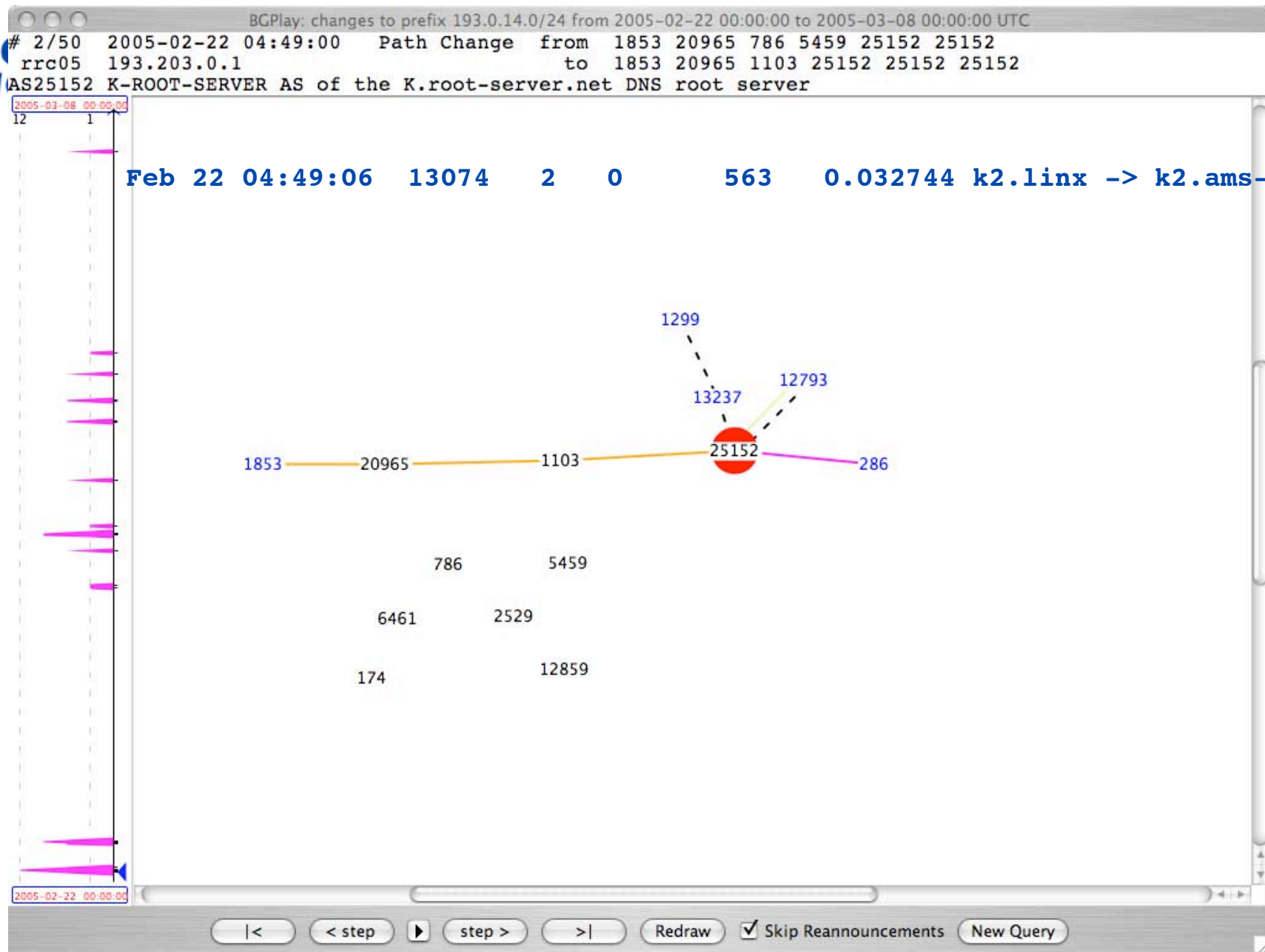
AS1853 to k.root-servers.net

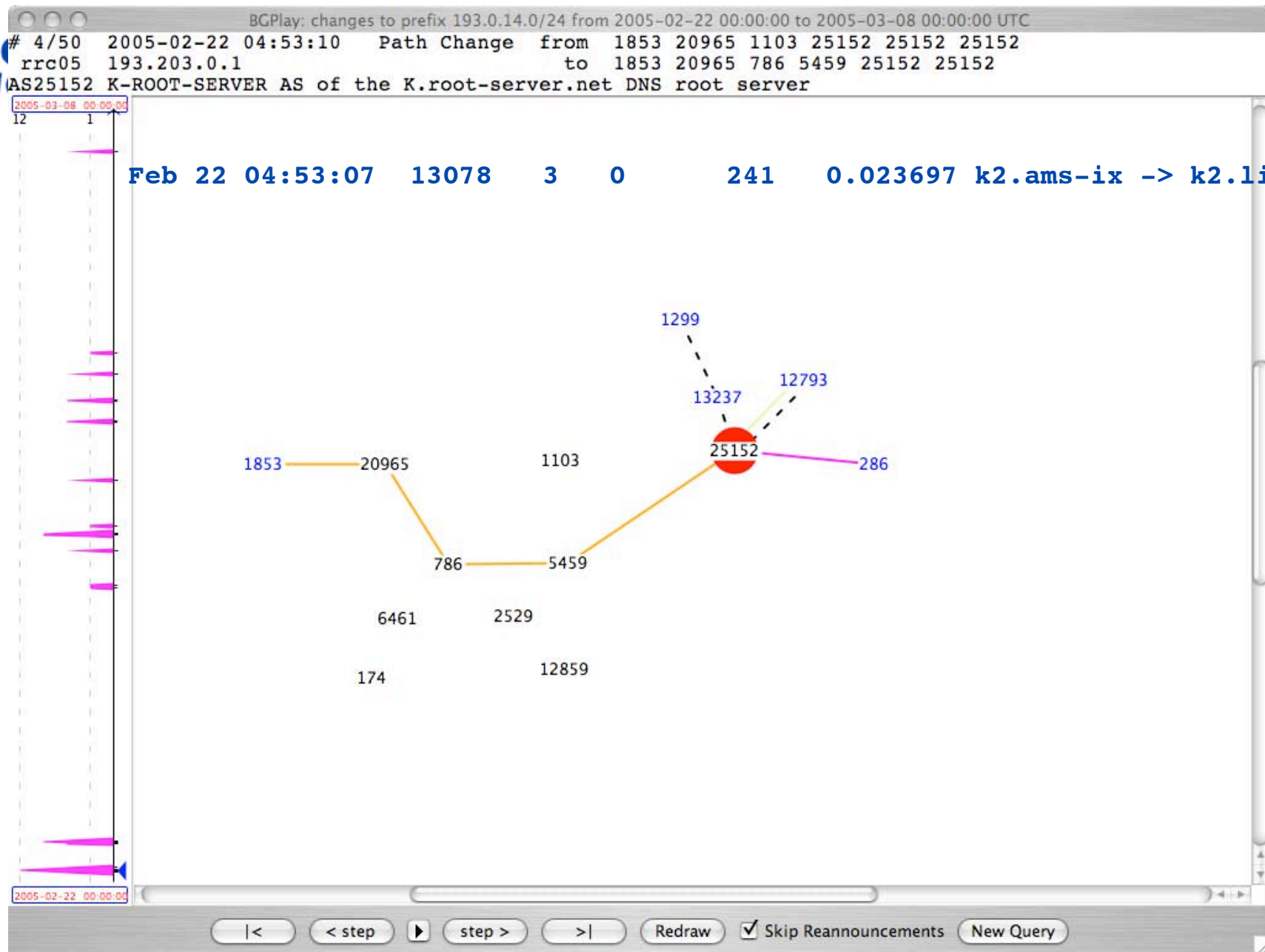
Feb 22 17:37:05	13833	19	2	1933	0.020593	k2.ams-ix -> k2.linx
Feb 22 17:40:03	13836	20	2	178	0.029839	k2.linx -> k2.ams-ix
Feb 22 17:50:47	13847	21	2	644	0.021476	k2.ams-ix -> k2.linx
Feb 22 17:52:54	13849	22	2	127	0.027259	k2.linx -> k2.ams-ix
Feb 22 18:04:21	13859	23	2	687	0.021546	k2.ams-ix -> k2.linx
Feb 22 18:06:37	13861	24	2	136	0.027392	k2.linx -> k2.ams-ix
Feb 28 07:04:58	21733	25	4	478701	0.022852	k2.ams-ix -> k2.linx
Feb 28 07:11:03	21739	26	7	365	0.034523	k2.linx -> k2.ams-ix
Feb 28 07:14:51	21742	27	7	228	0.034898	k2.ams-ix -> k2.linx
Feb 28 07:19:39	21747	28	7	288	0.035498	k2.linx -> k2.ams-ix
Feb 28 07:28:12	21755	29	8	513	0.034860	k2.ams-ix -> k2.linx
Feb 28 07:39:47	21766	30	8	695	0.035535	k2.linx -> k2.ams-ix
Feb 28 07:57:35	21784	31	8	1068	0.034521	k2.ams-ix -> k2.linx
Feb 28 07:59:20	21786	32	8	105	0.034709	k2.linx -> k2.ams-ix
Mar 1 06:51:41	23146	33	8	82341	0.034695	k2.ams-ix -> k2.linx
Mar 1 06:53:28	23148	34	8	107	0.079817	k2.linx -> k2.ams-ix
Mar 7 06:02:06	31629	35	8	515318	0.034554	k2.ams-ix -> k2.linx
Mar 7 06:05:27	31632	36	8	201	0.034243	k2.linx -> k2.ams-ix

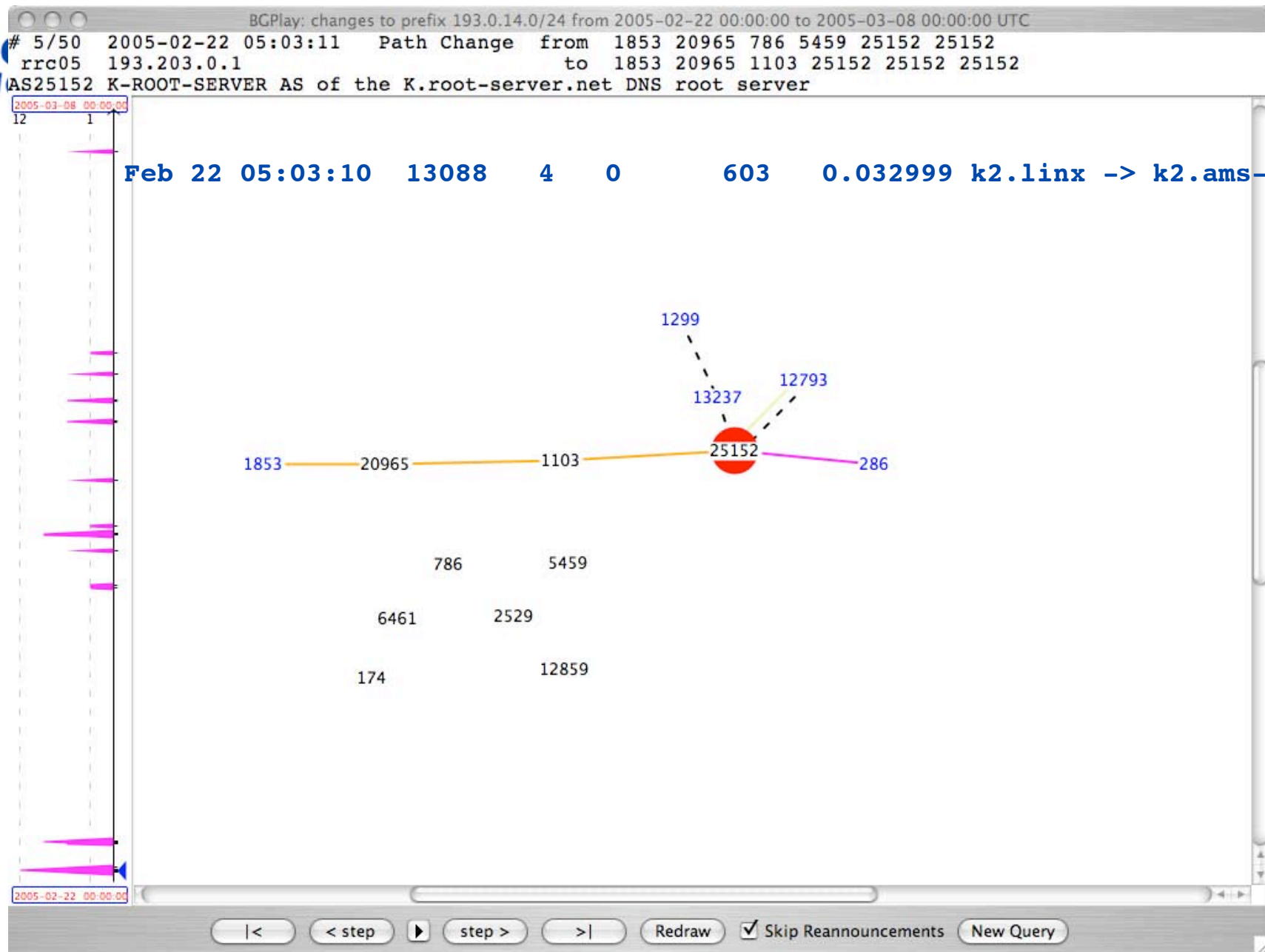


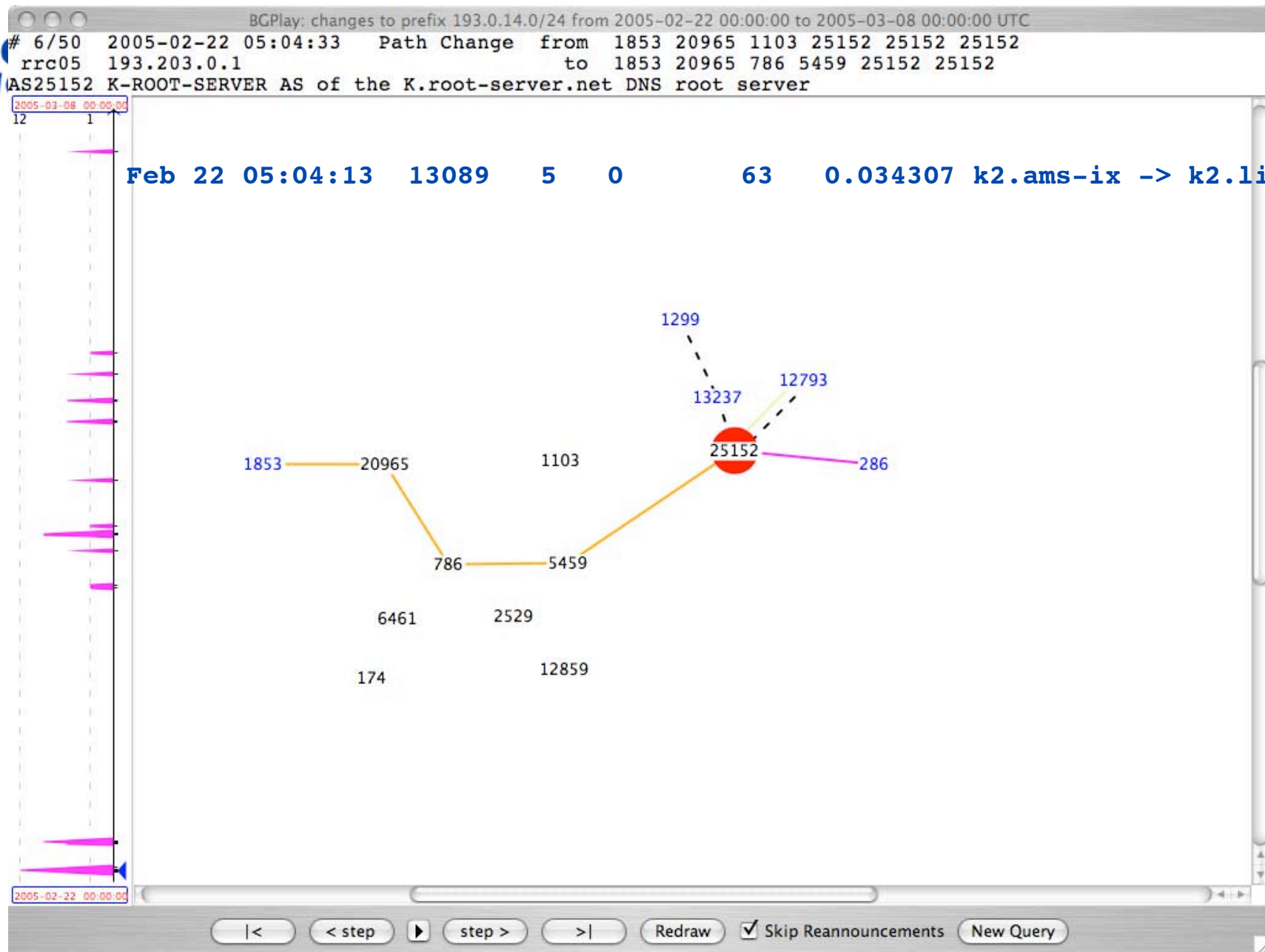


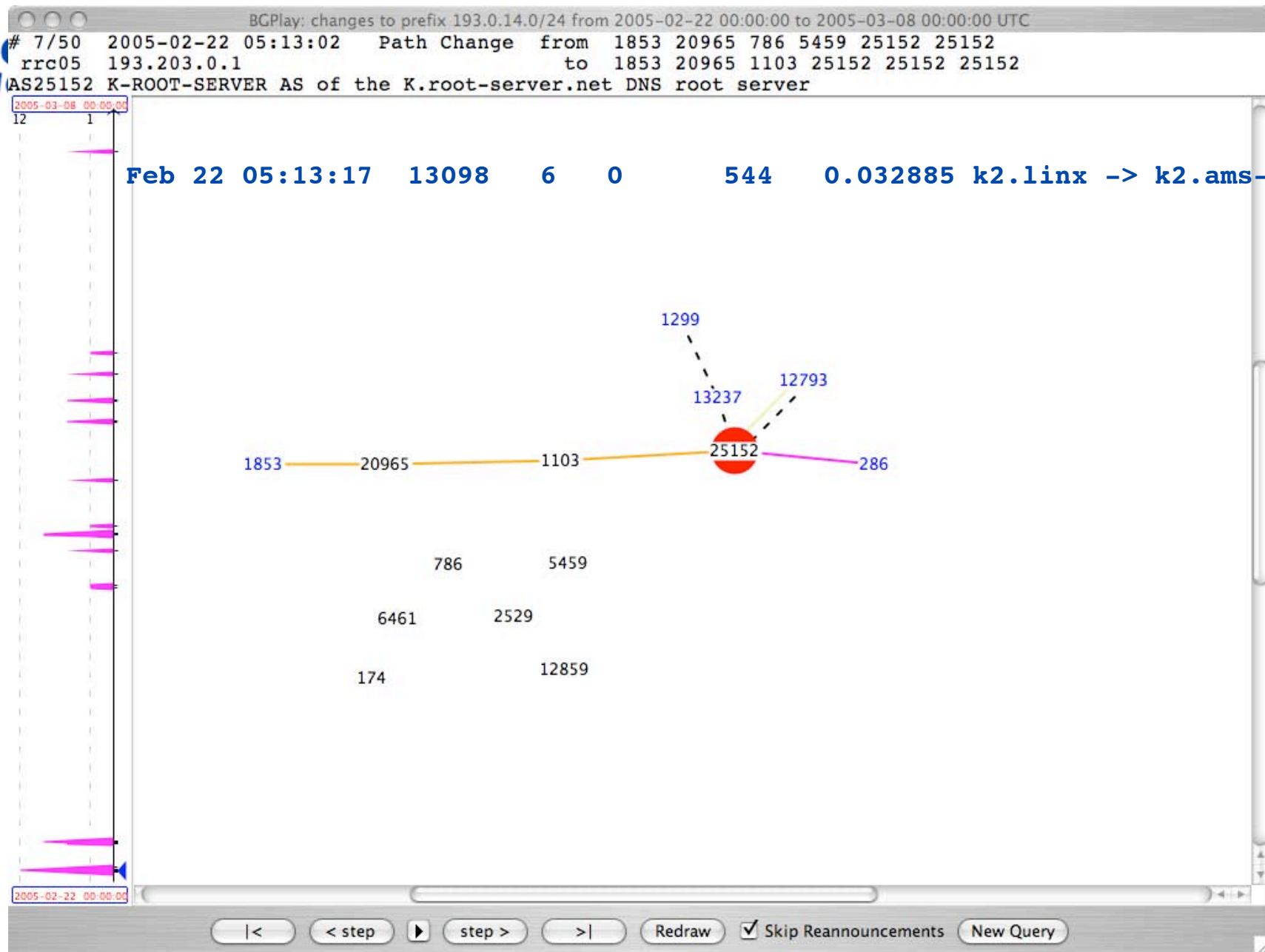


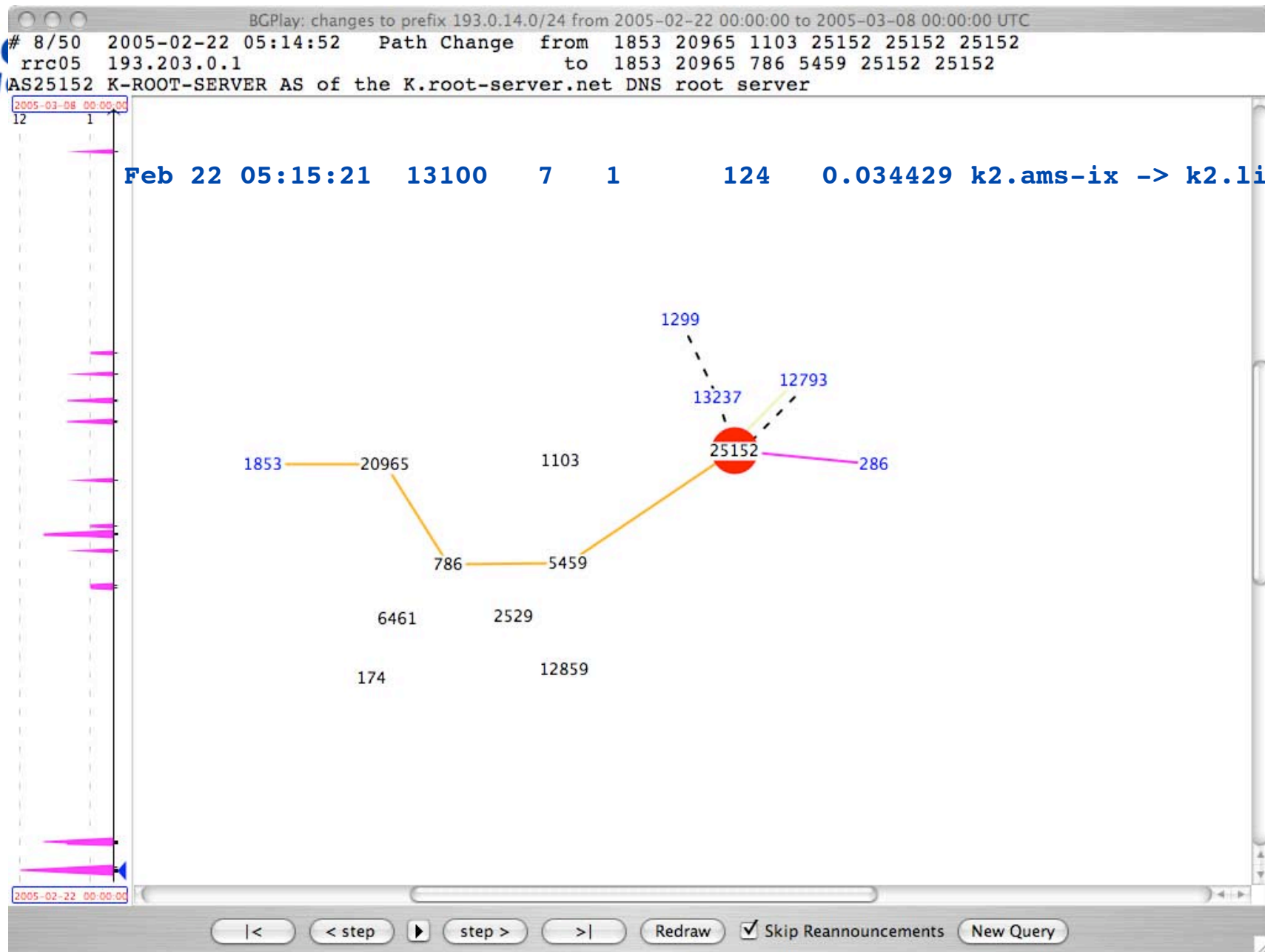


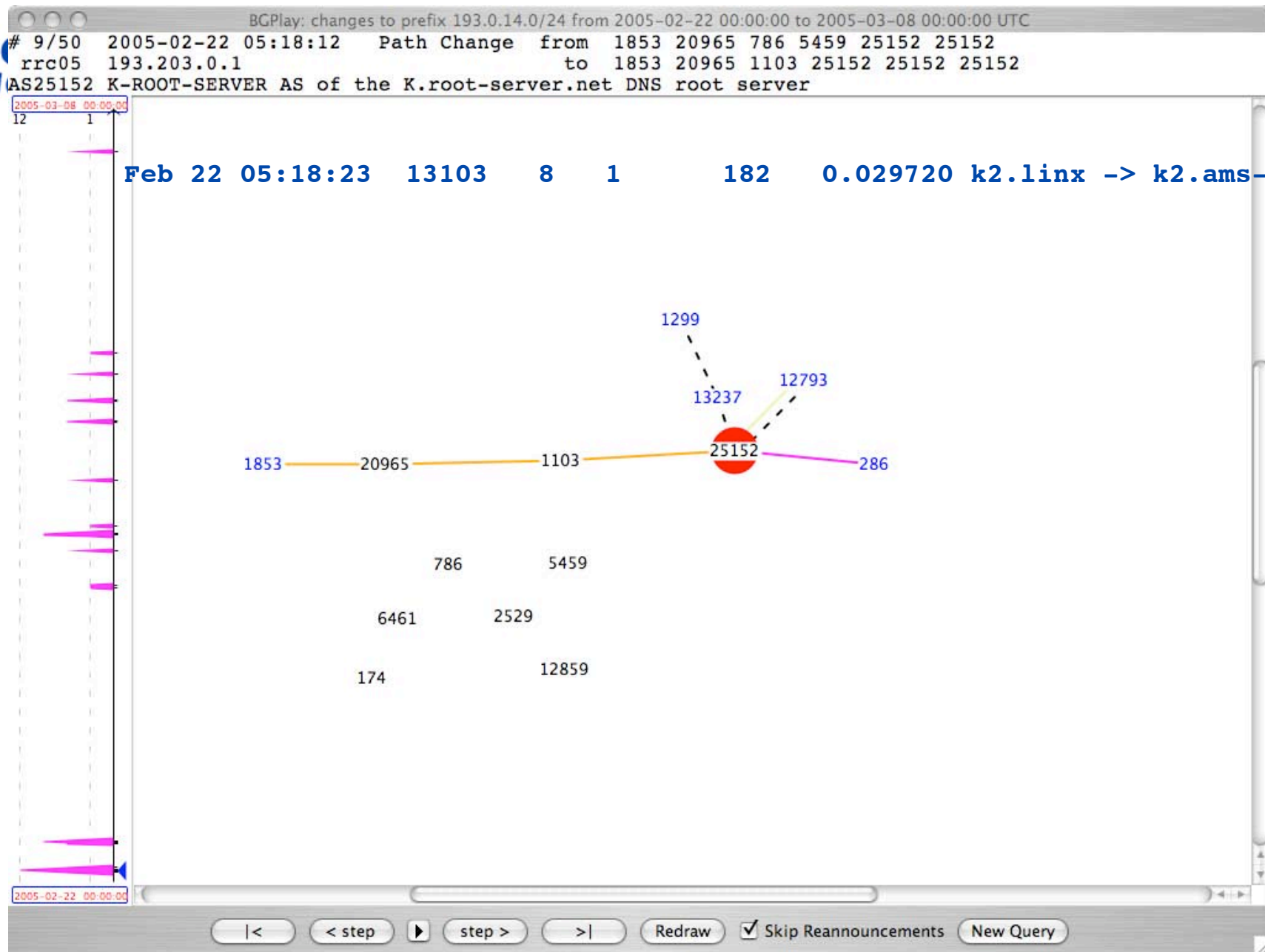












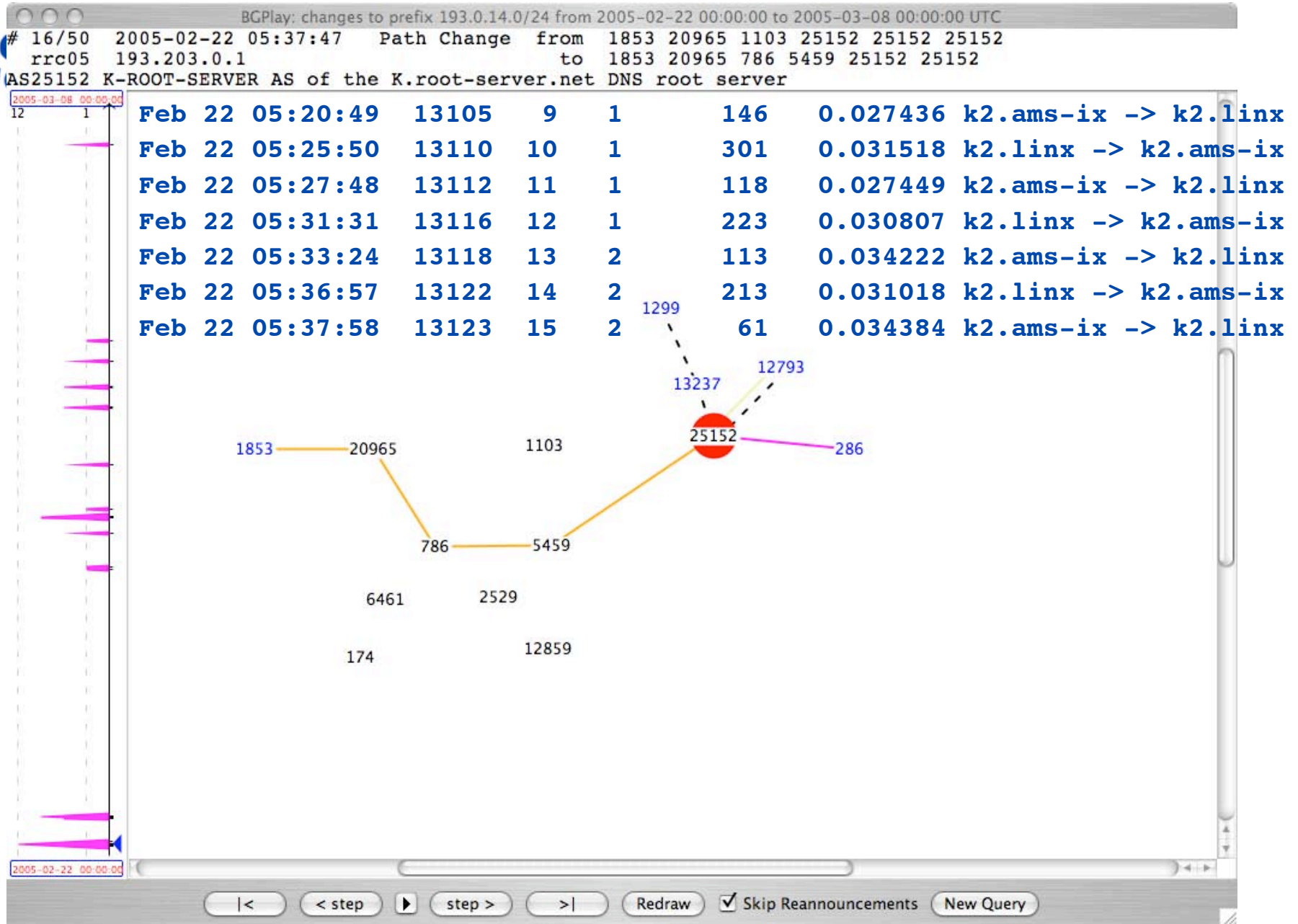


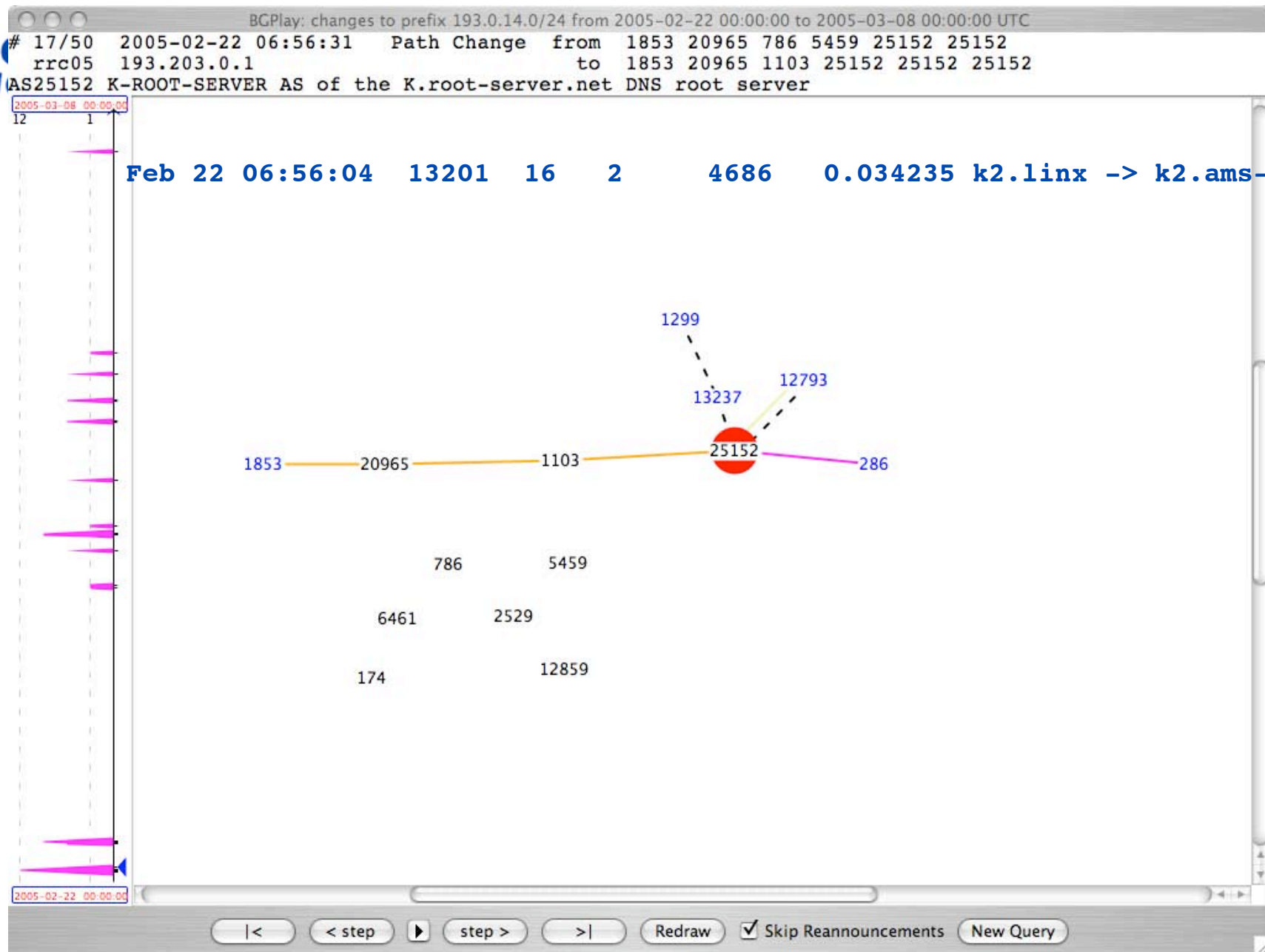
Observations

- 1 BGP path change per instance switch
 - Suggests we are not missing much because of lower probing frequency
- Cause of path change probably instability on preferred path
 - Concurrent BGP updates for other prefixes suggest that
- A little packet loss

- This keeps churning (not all path changes shown)
 - It settles down on the alternate path ...
 - Then switches back to the preferred path after another hour or so

- Note the equal length of the paths (by prepending)

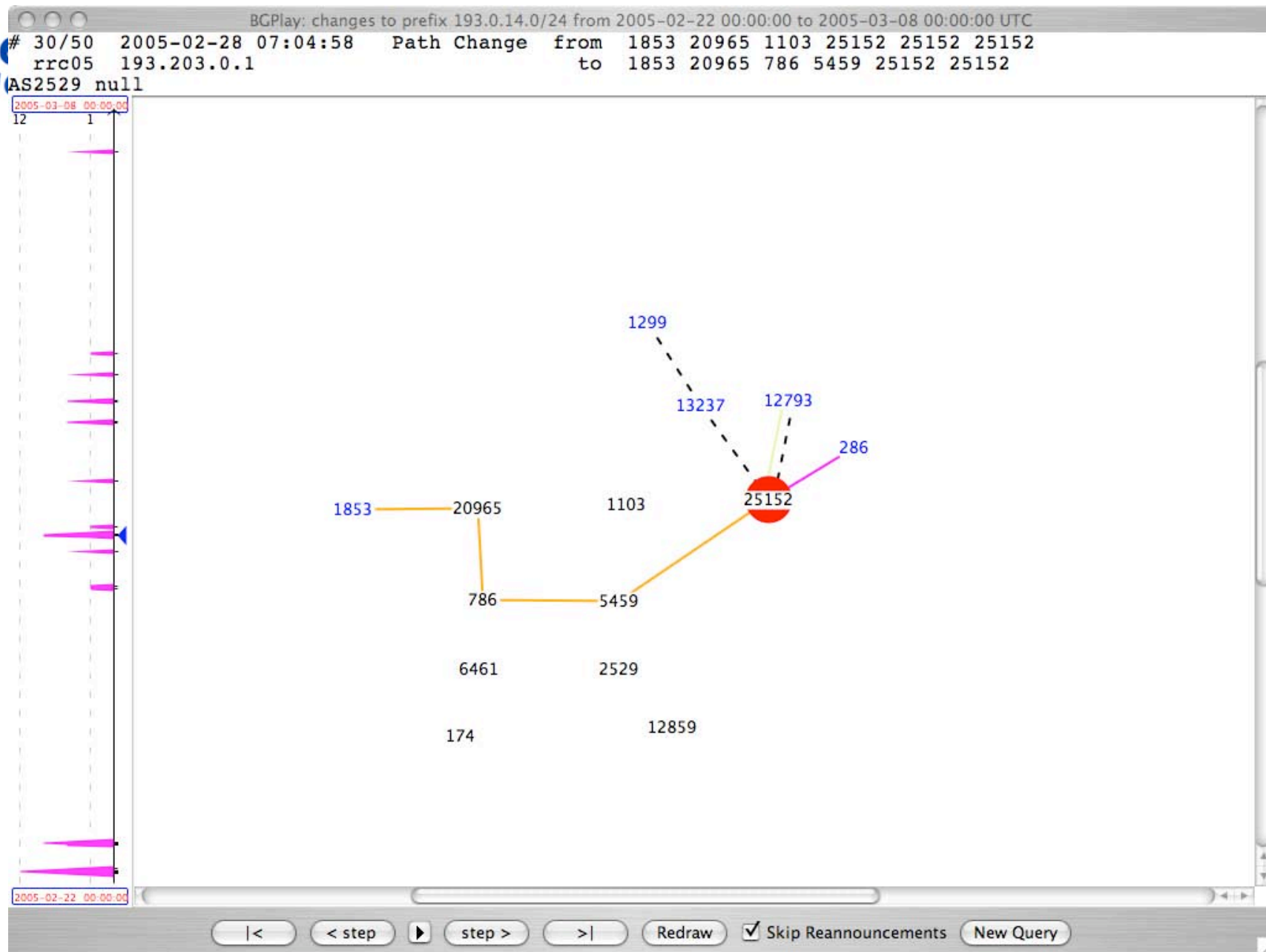


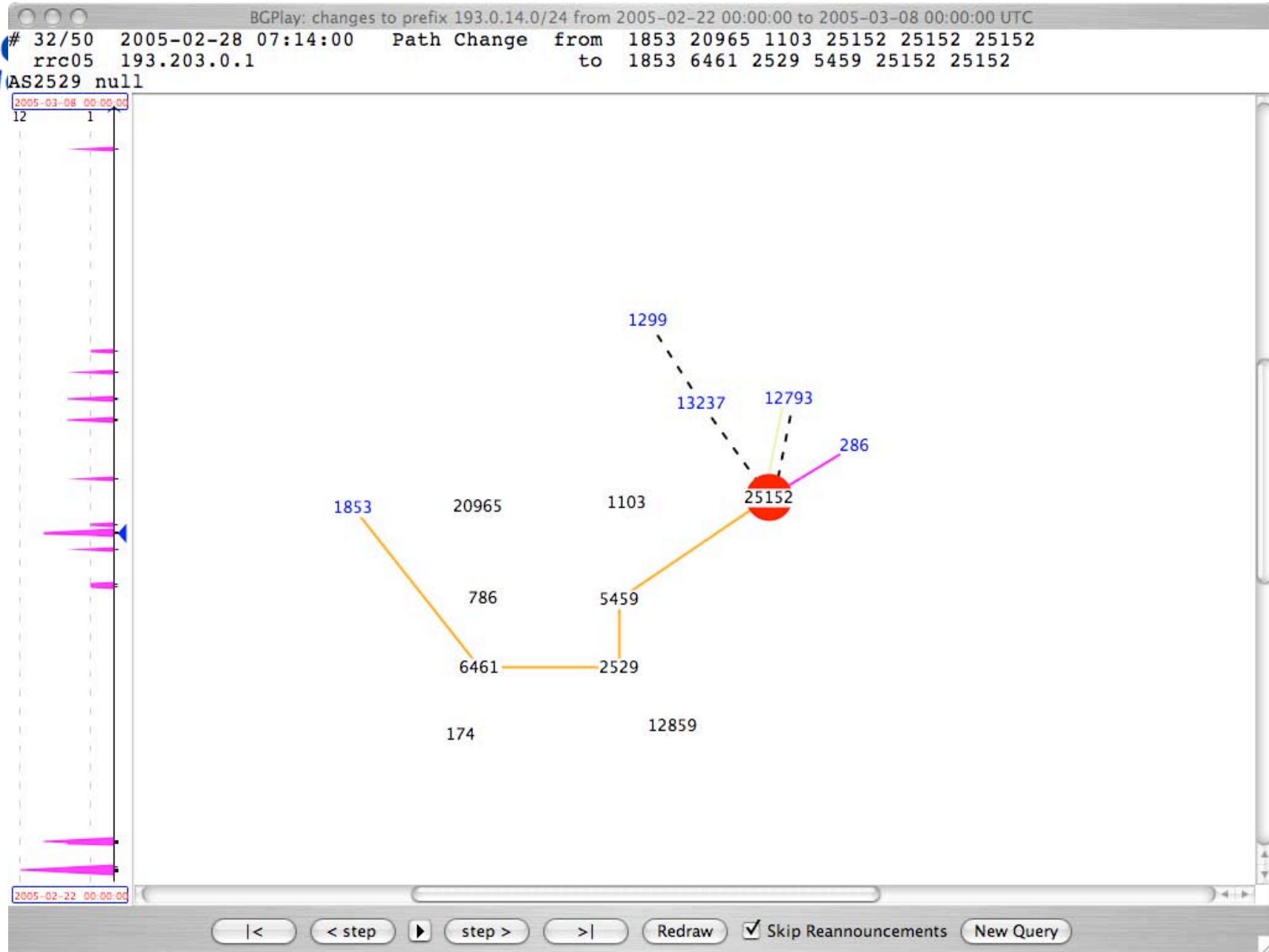


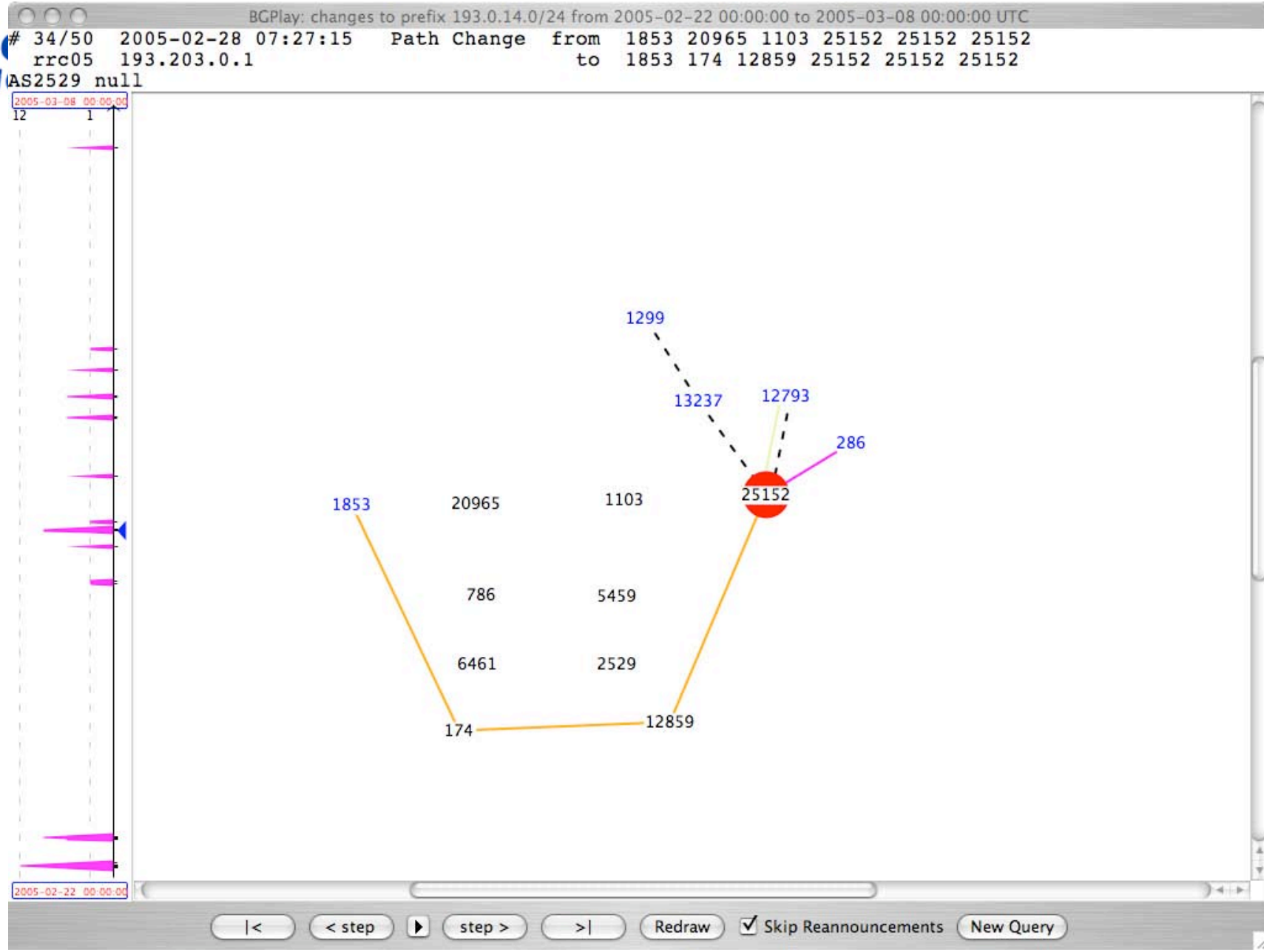


More ...

- Other paths observed
- DNSMON data not repeated in graphs
 - but timing and IDs correspond closely to all switches
 - If you want to see the whole movie, use bgplay yourself
- Note the path lengths again!



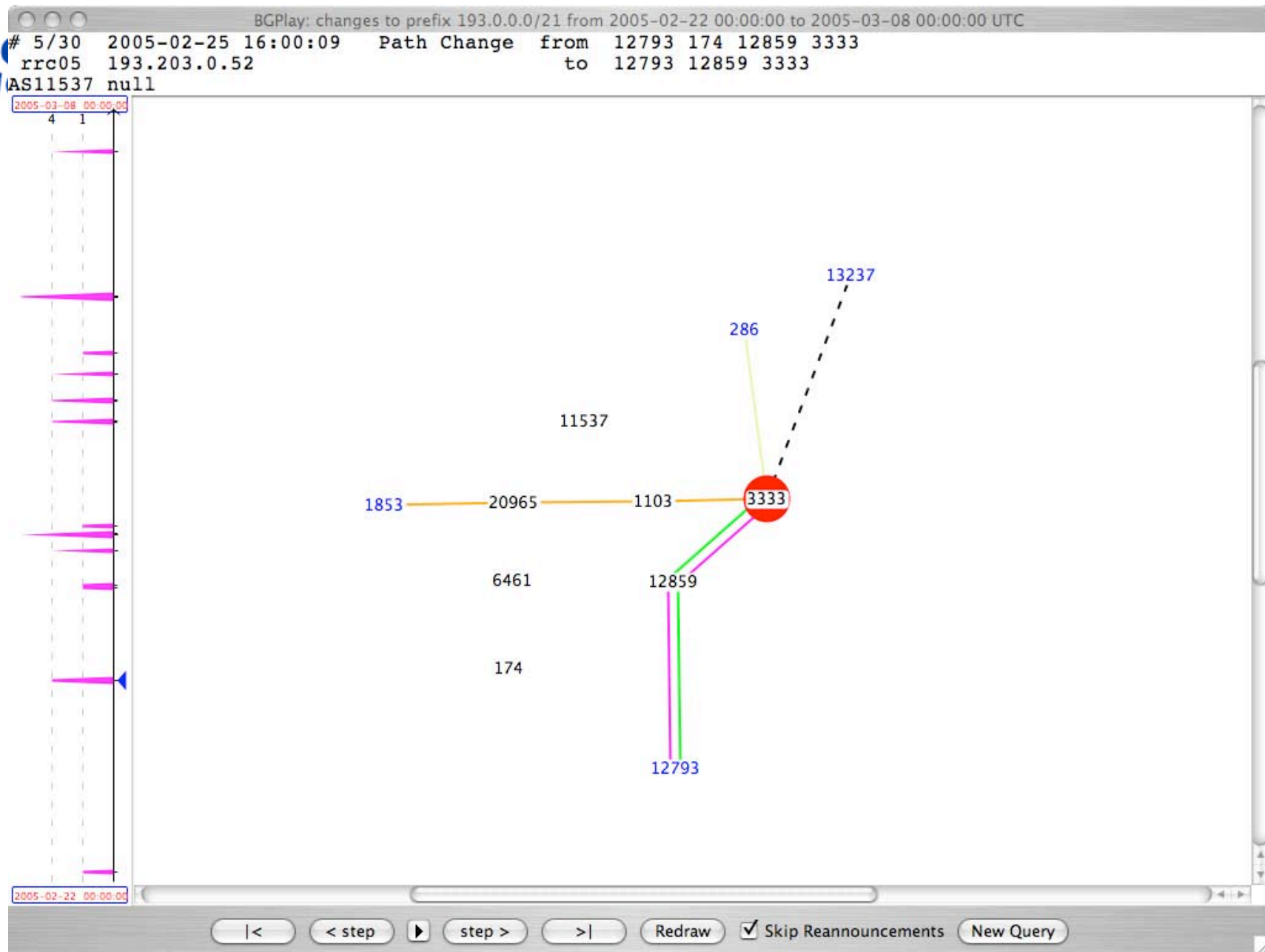


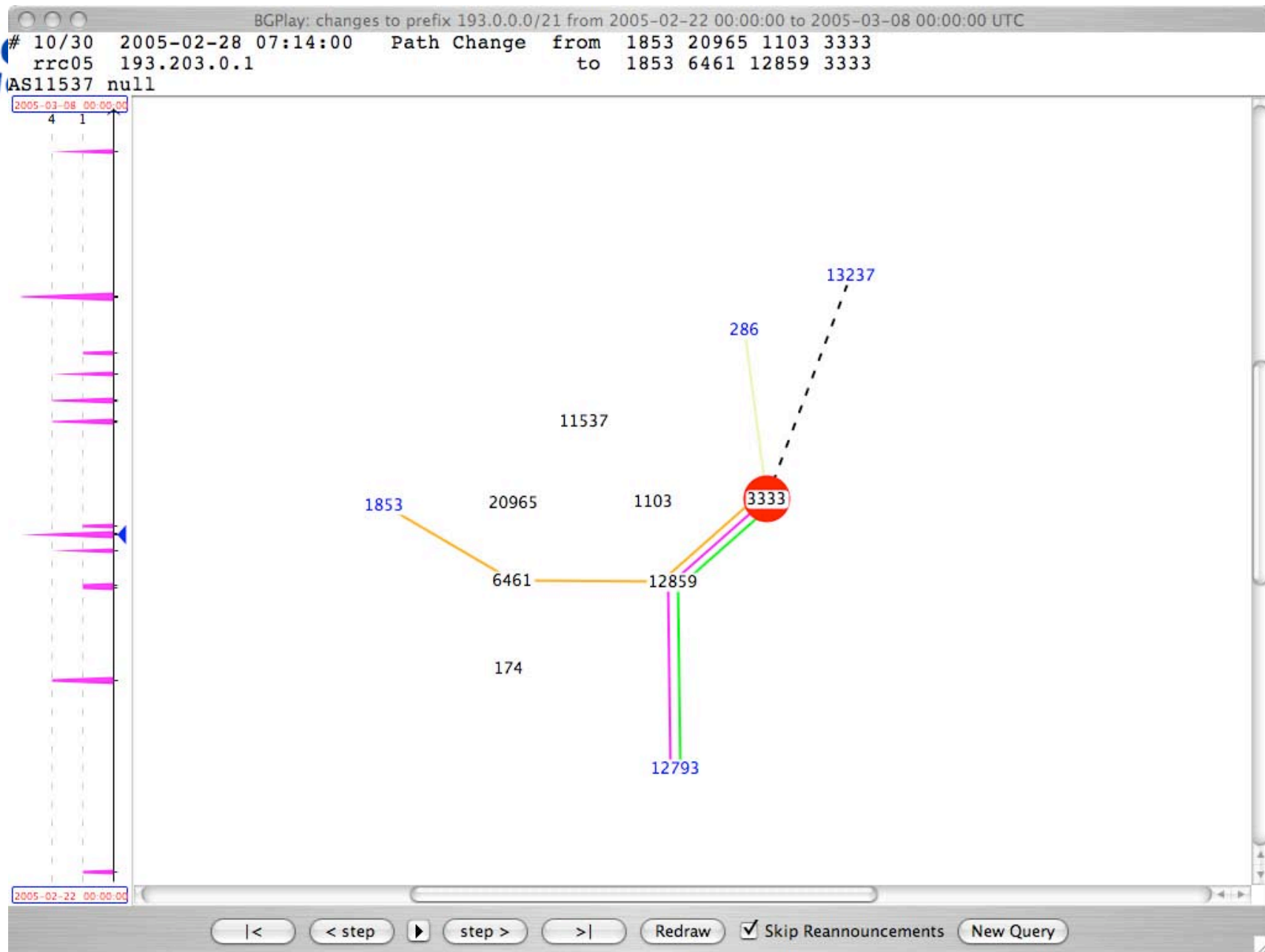


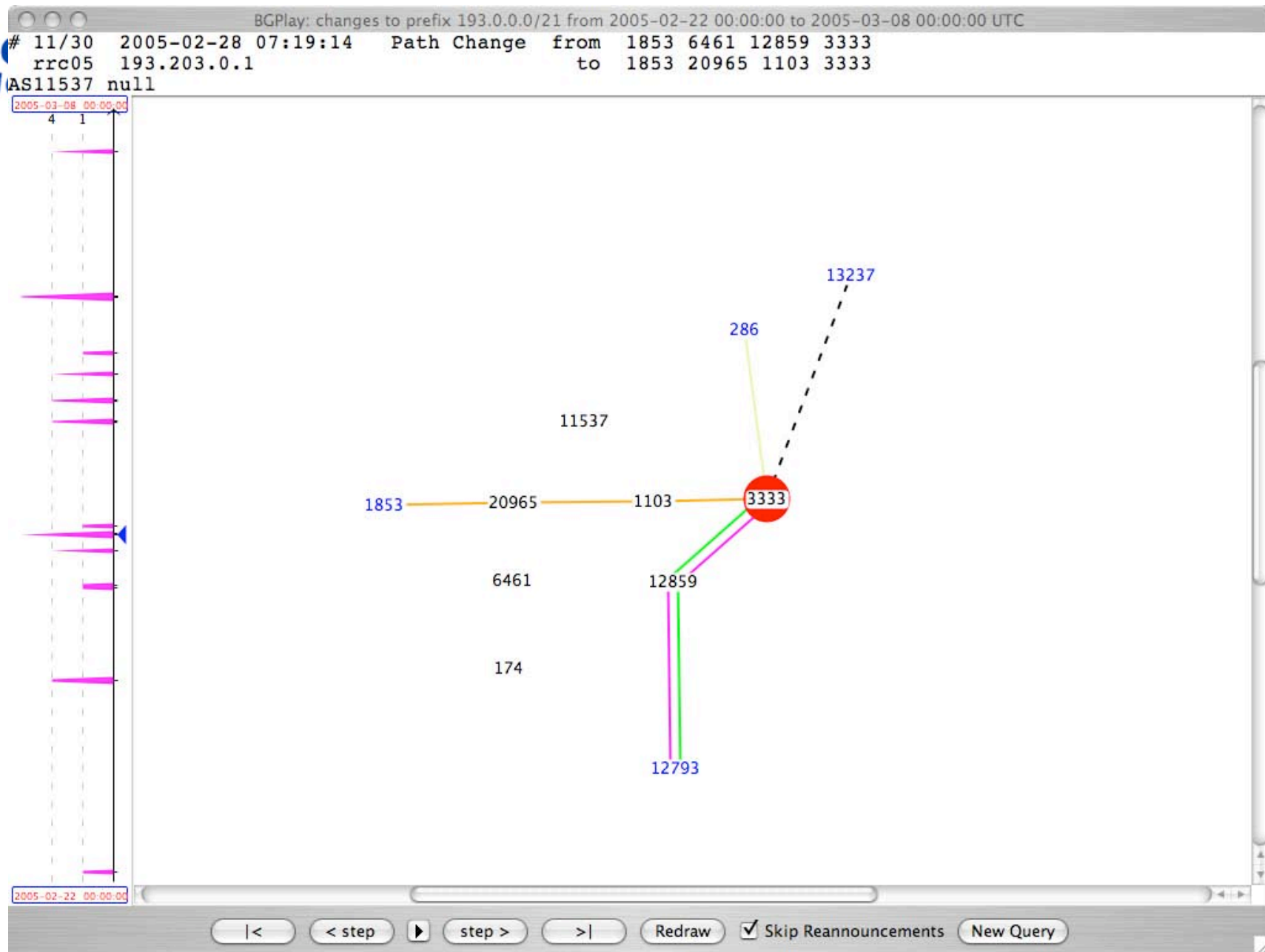


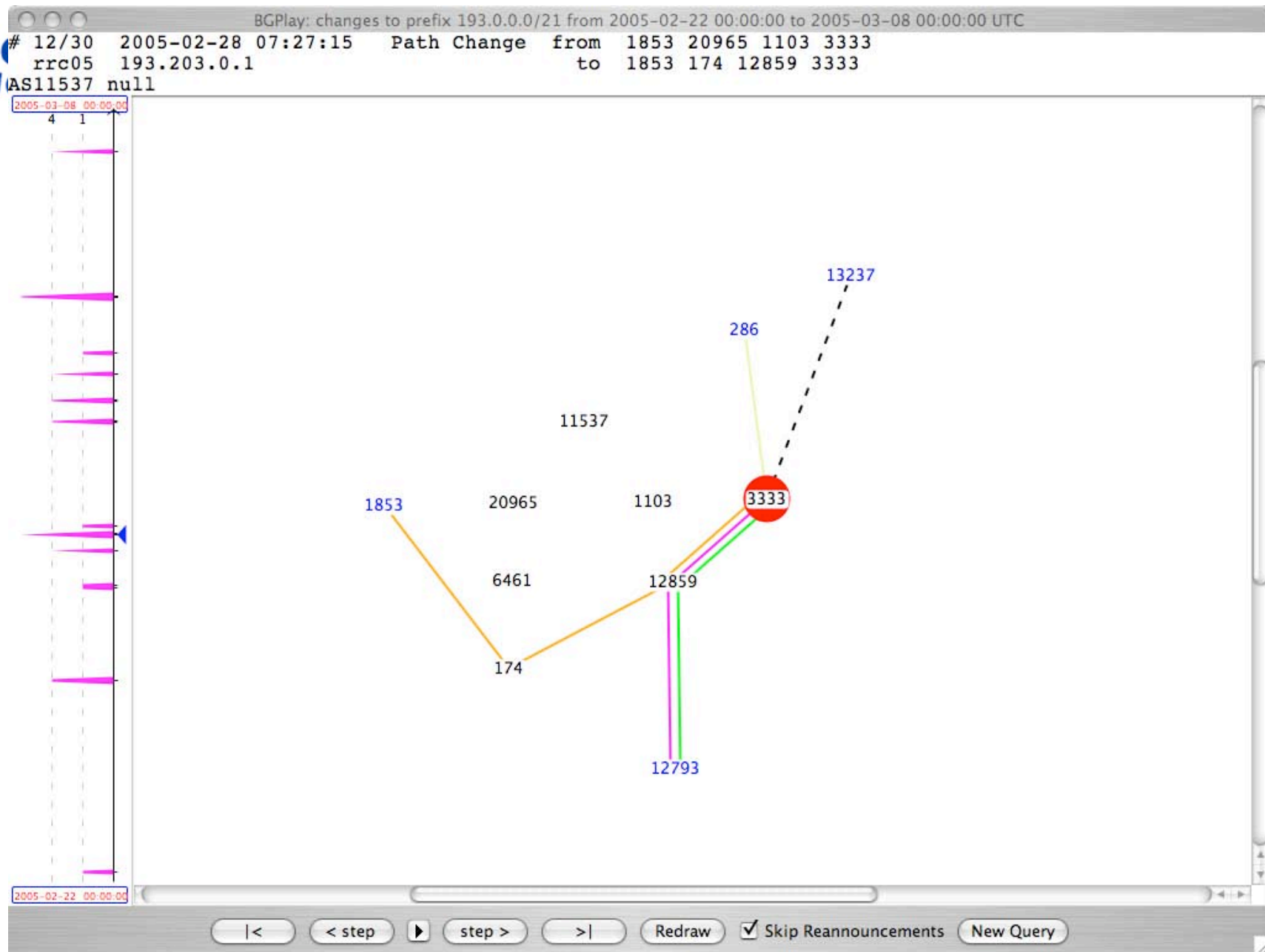
Is this unique to anycast?

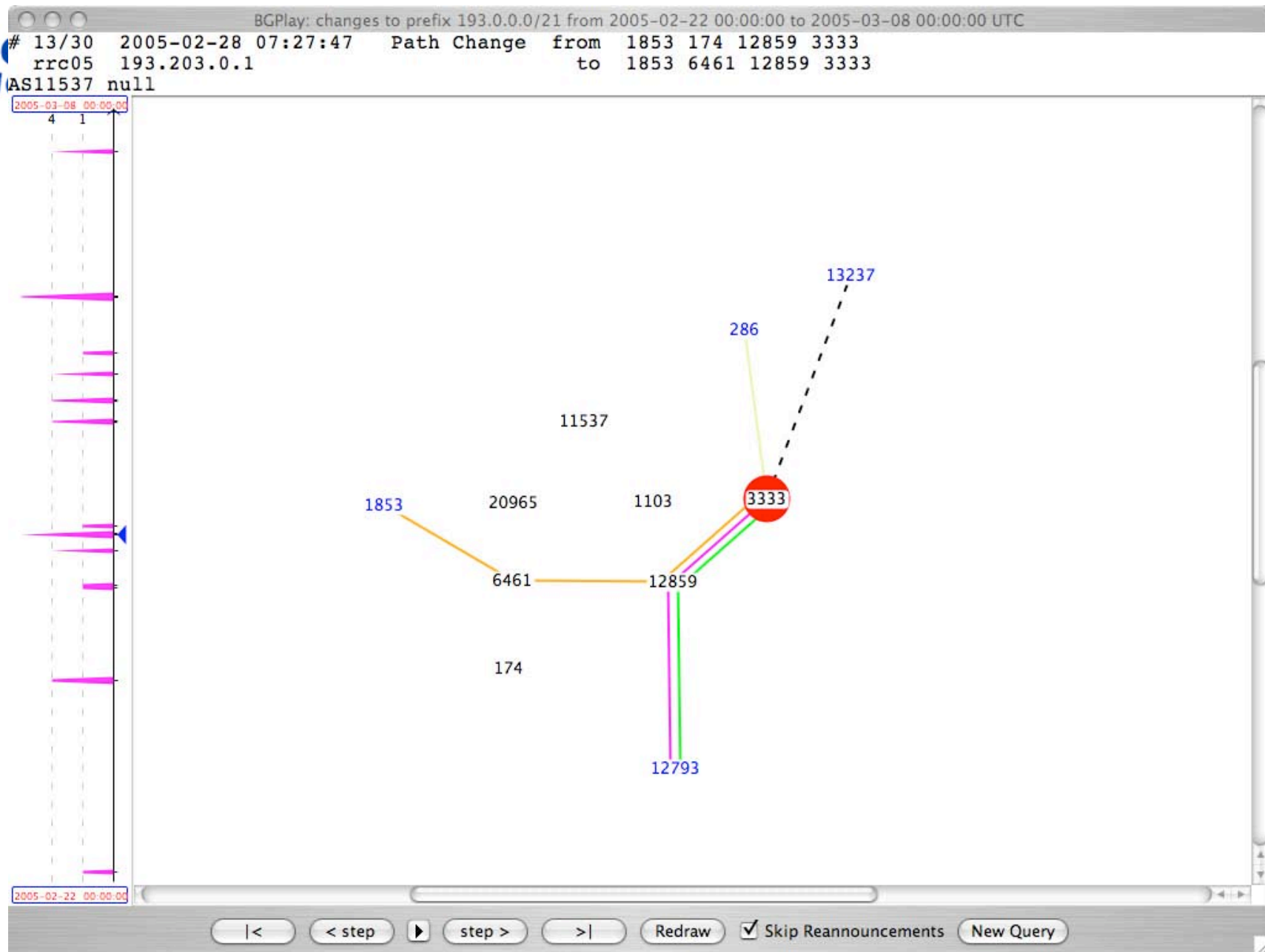
- No, unicast prefixes also have
 - Equal path lengths
 - Instabilities
- Watch this ...
 - Same time frame as last few slides
 - Prefix is now a unicast /21

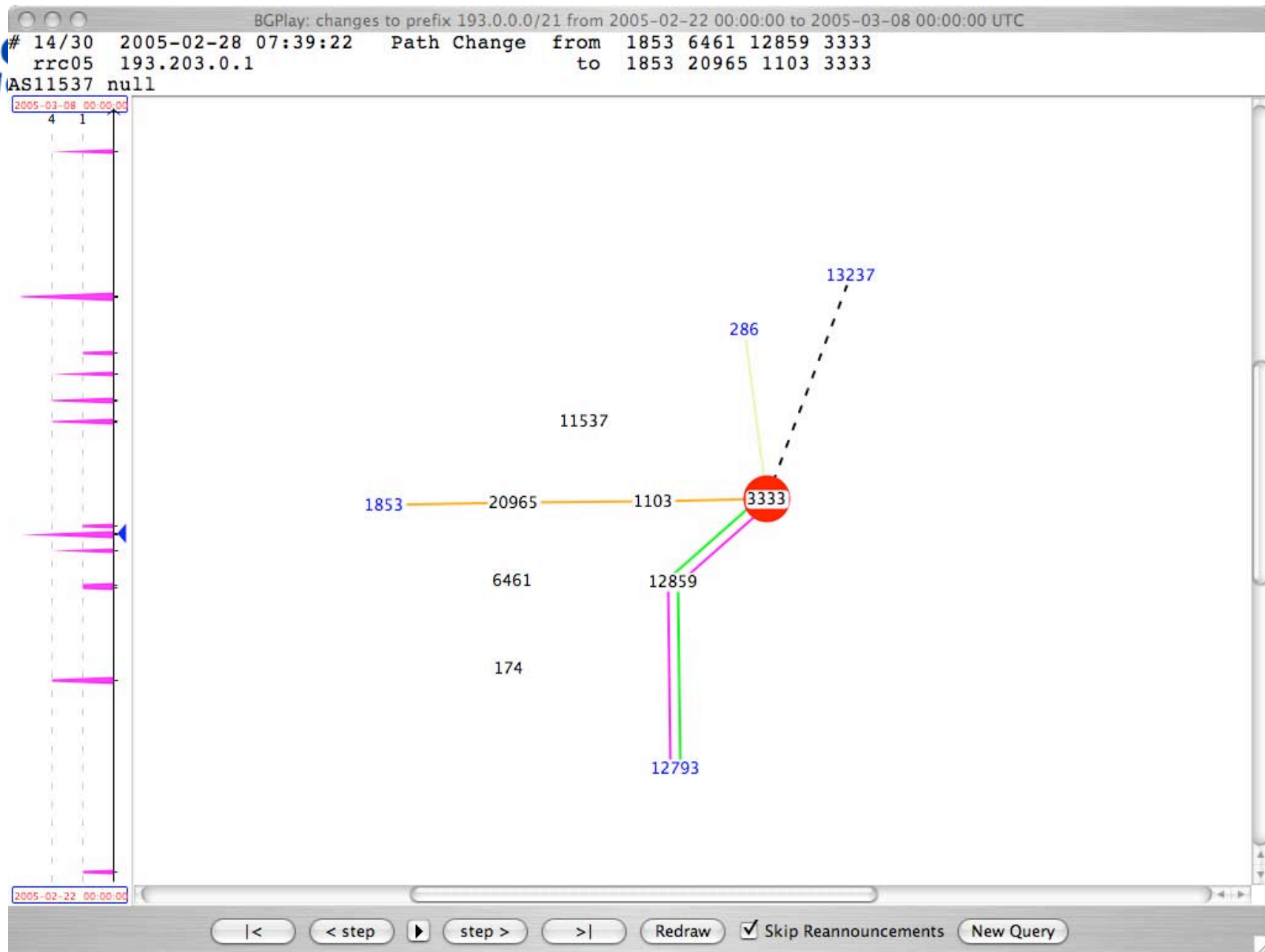








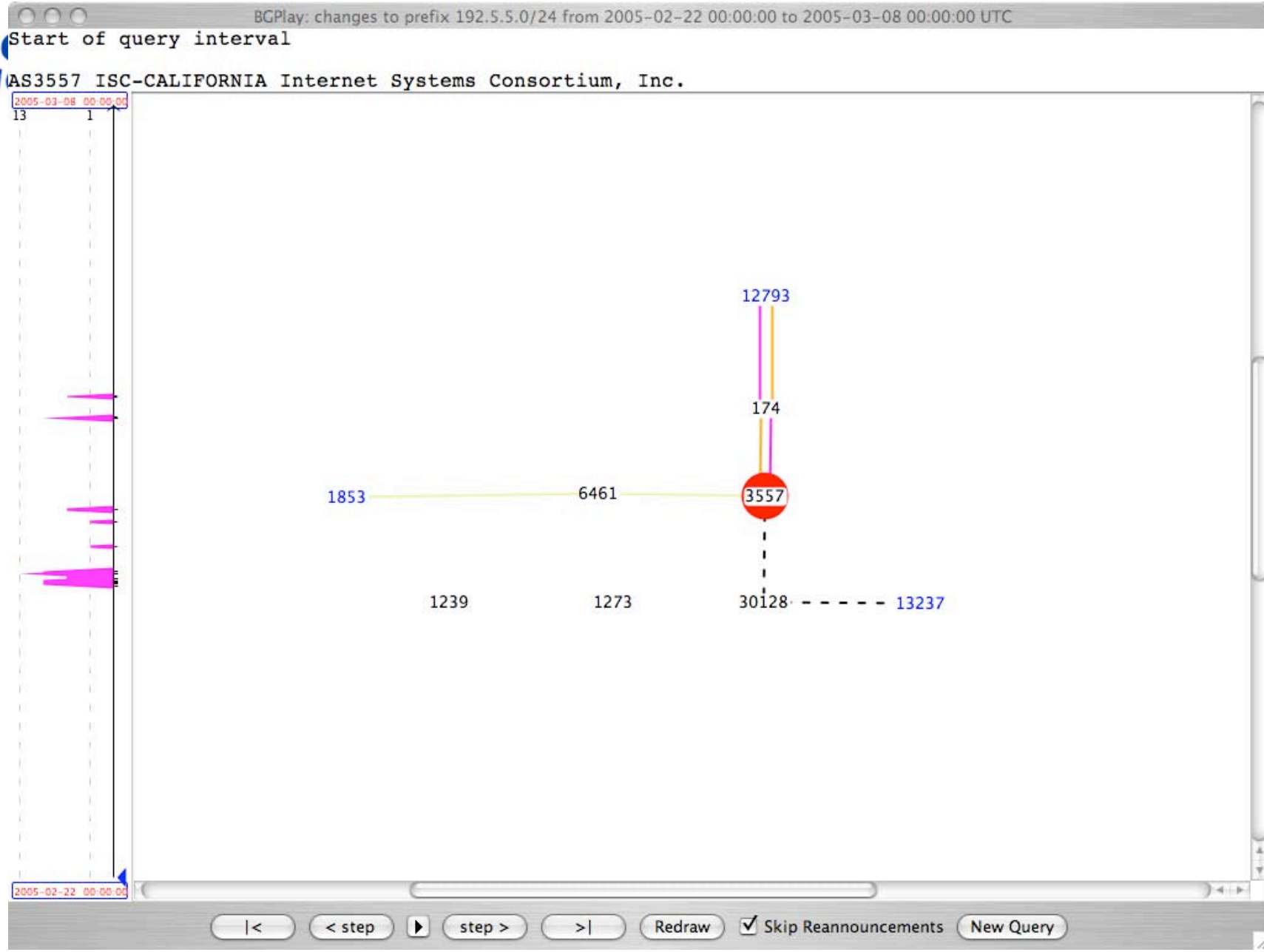


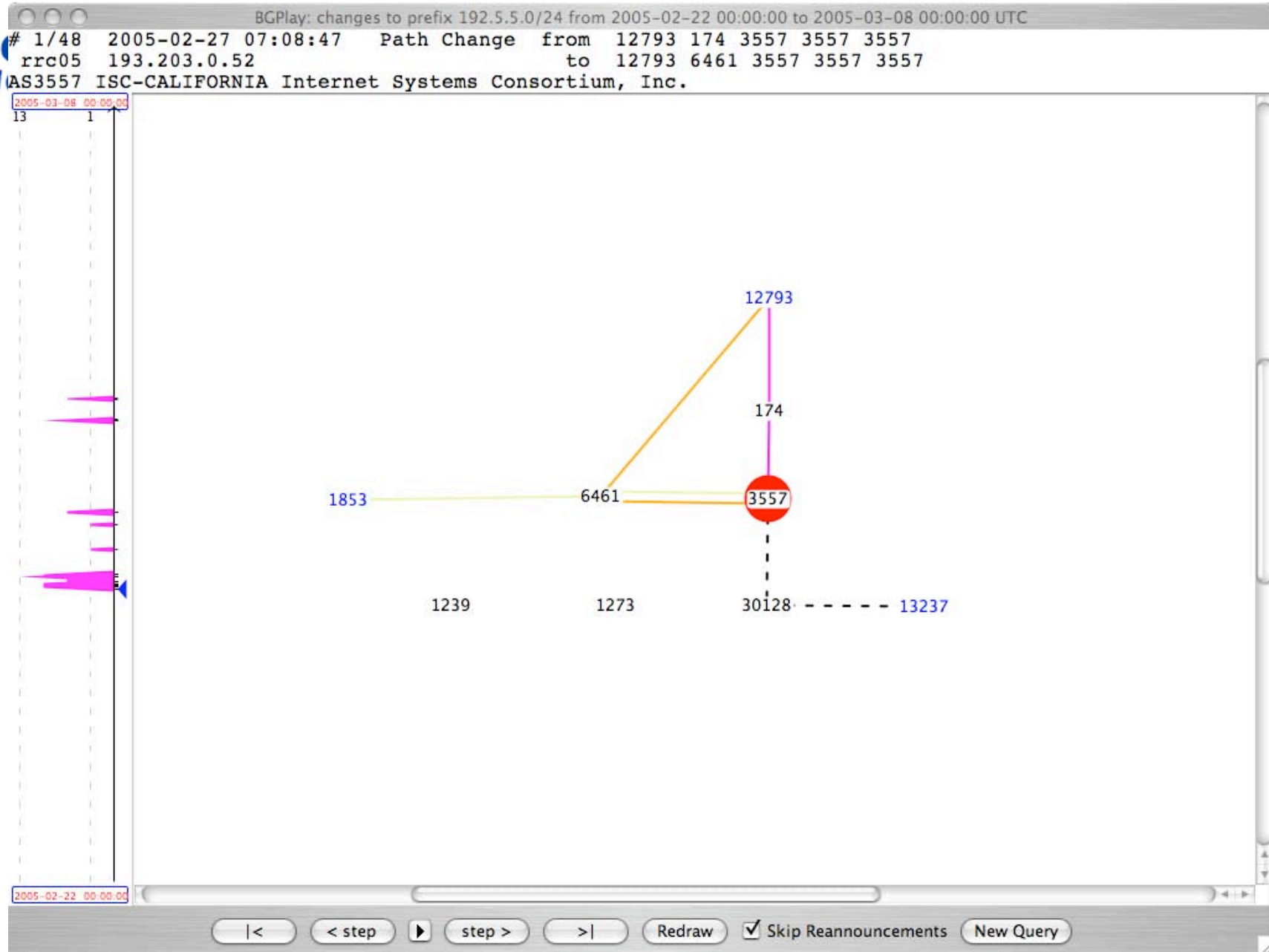


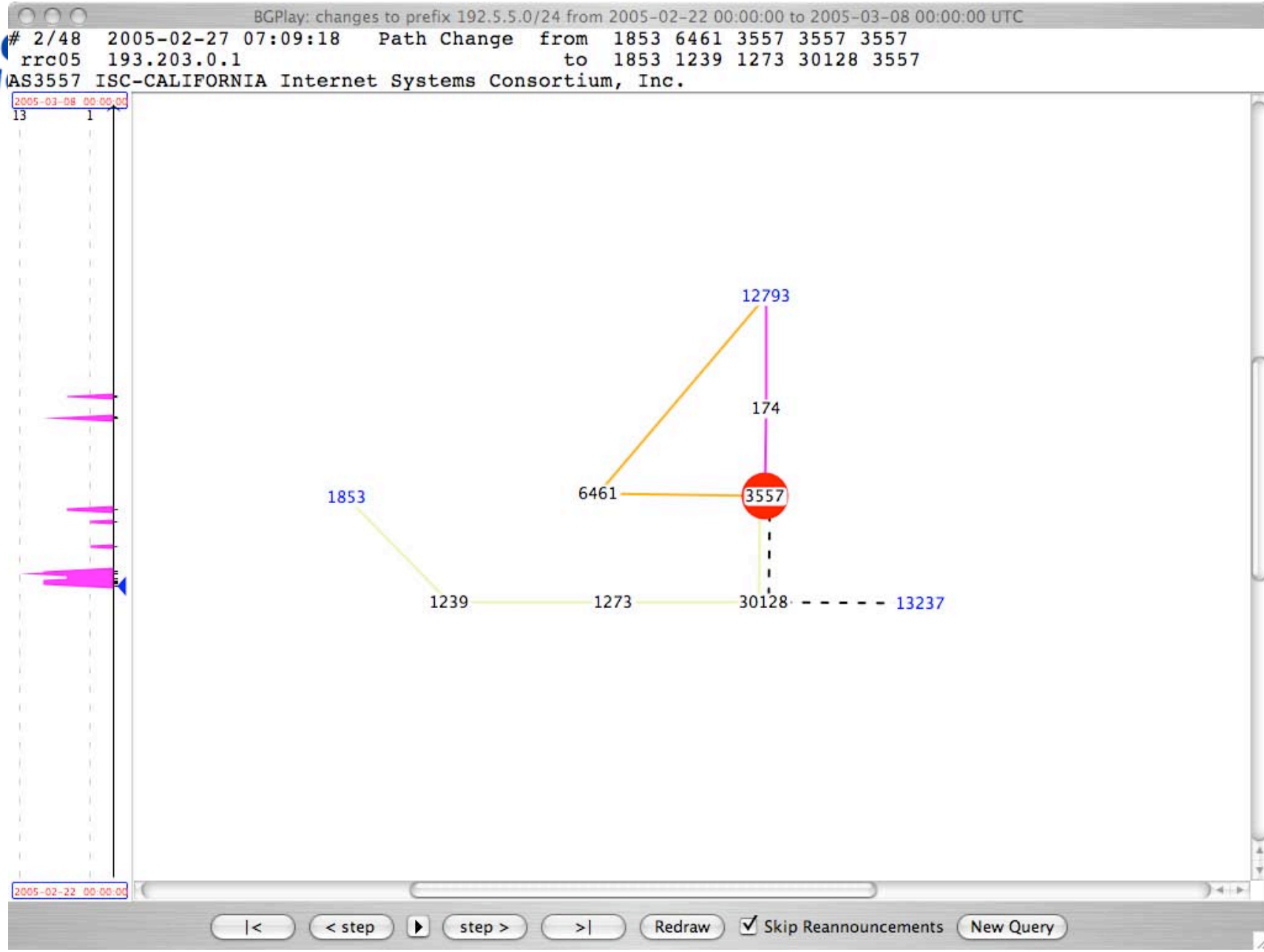


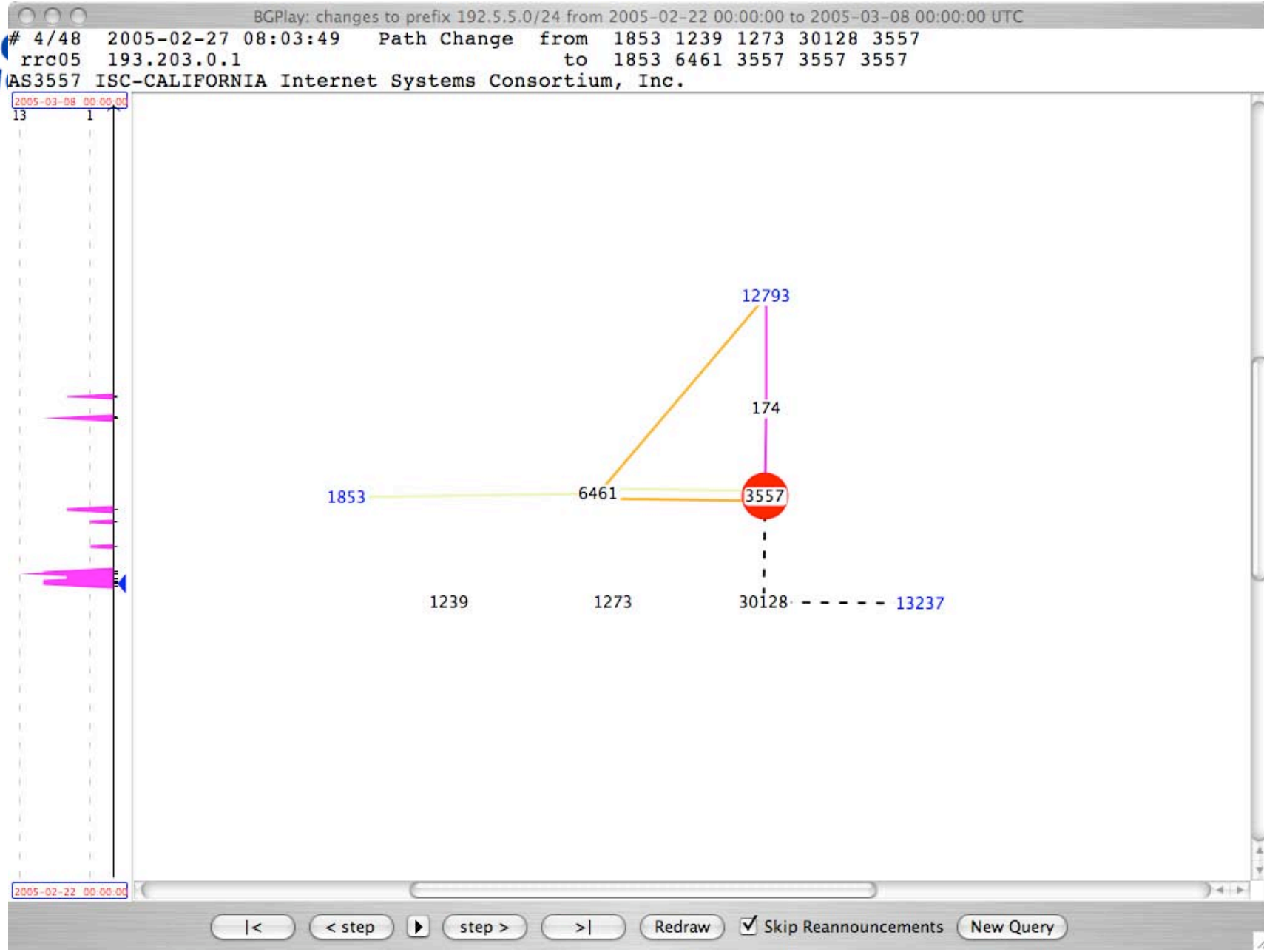
AS1853 to f.root-servers.net

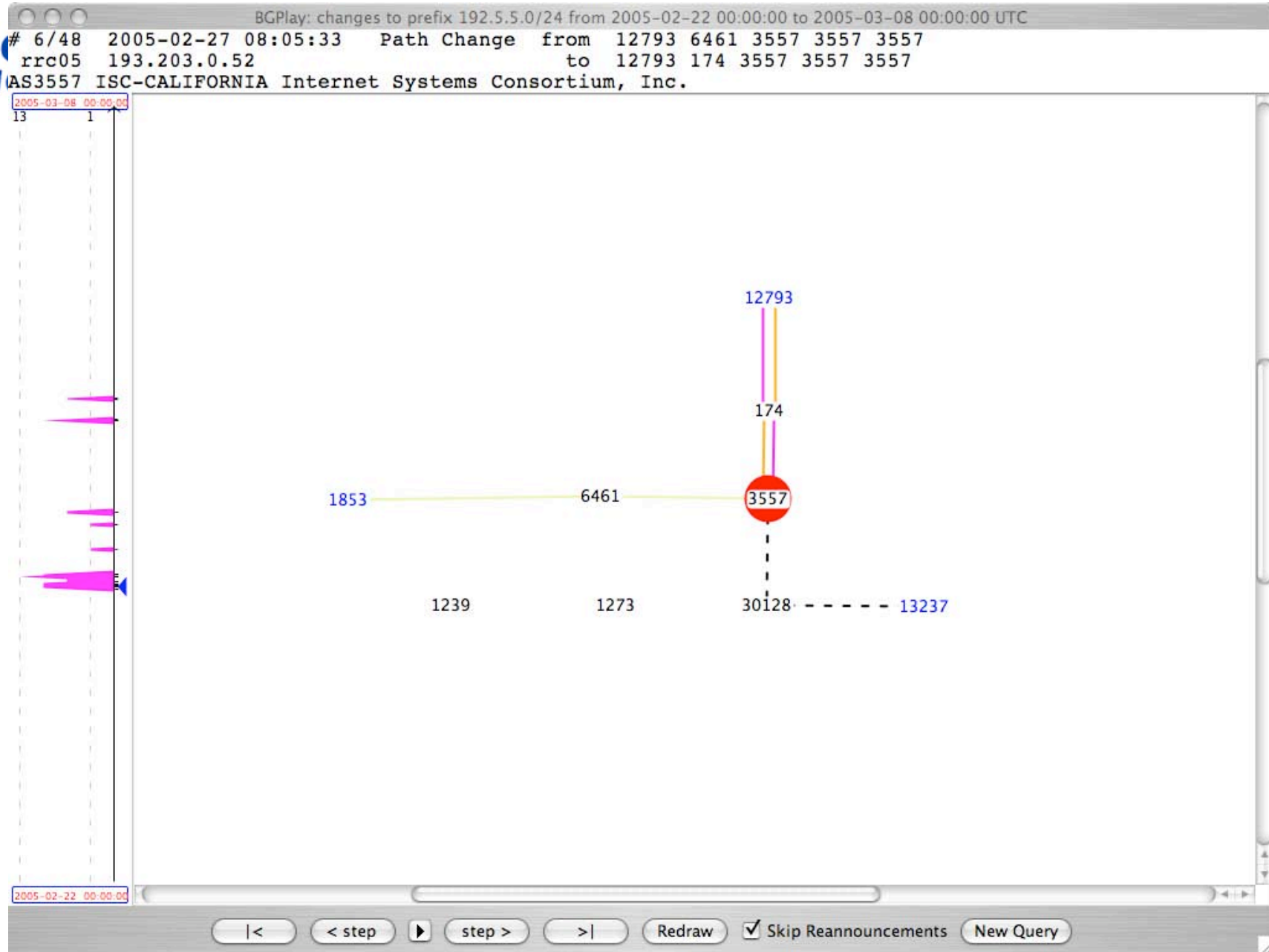
- Last example
- Involves two global nodes of F
 - pao1 and sfo2
 - Both connected via AS6461 and AS174
- Plus one local node
 - muc1
 - Connected via AS30128
- This shows the shorter (/24) prefix only!

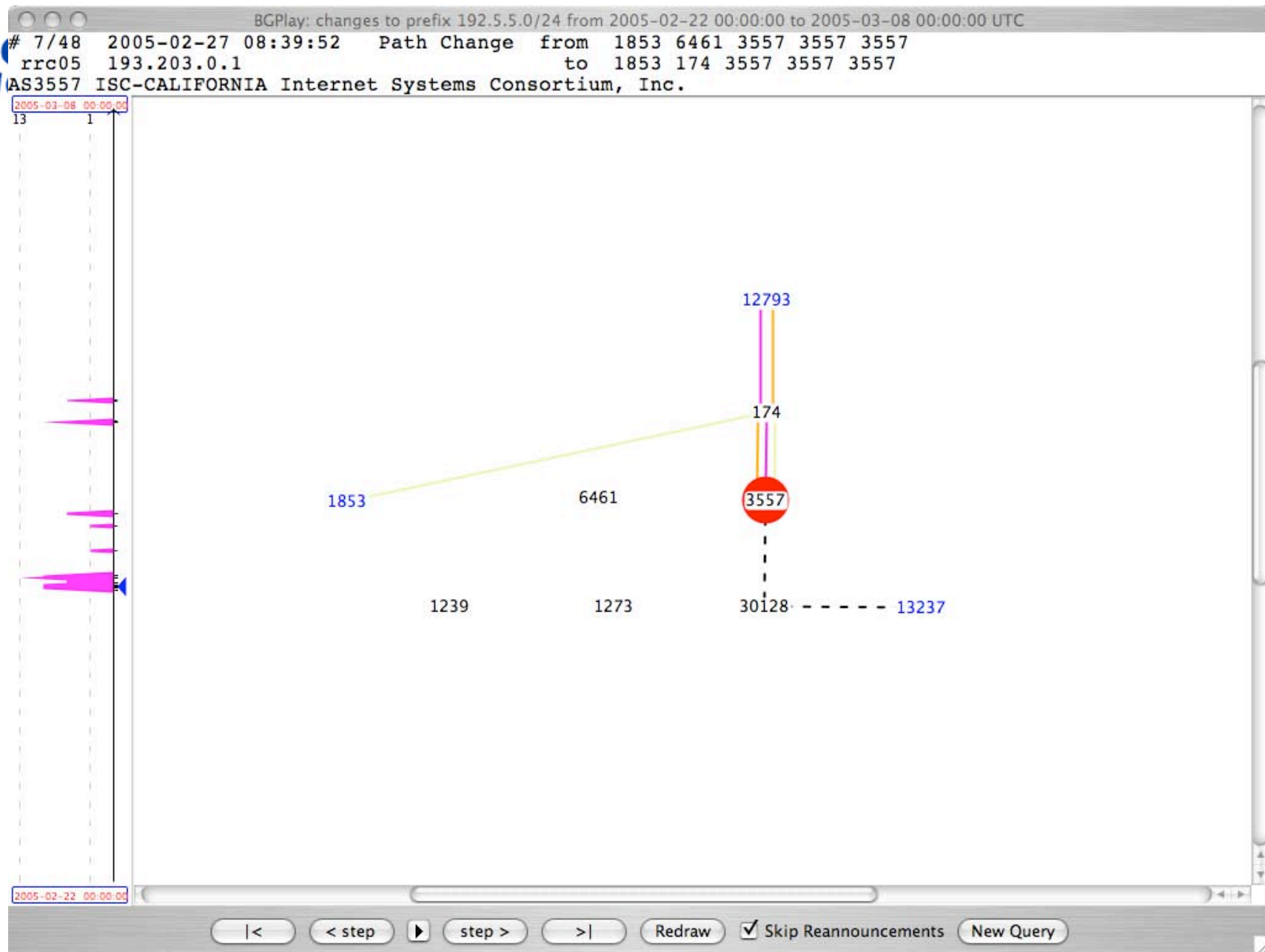


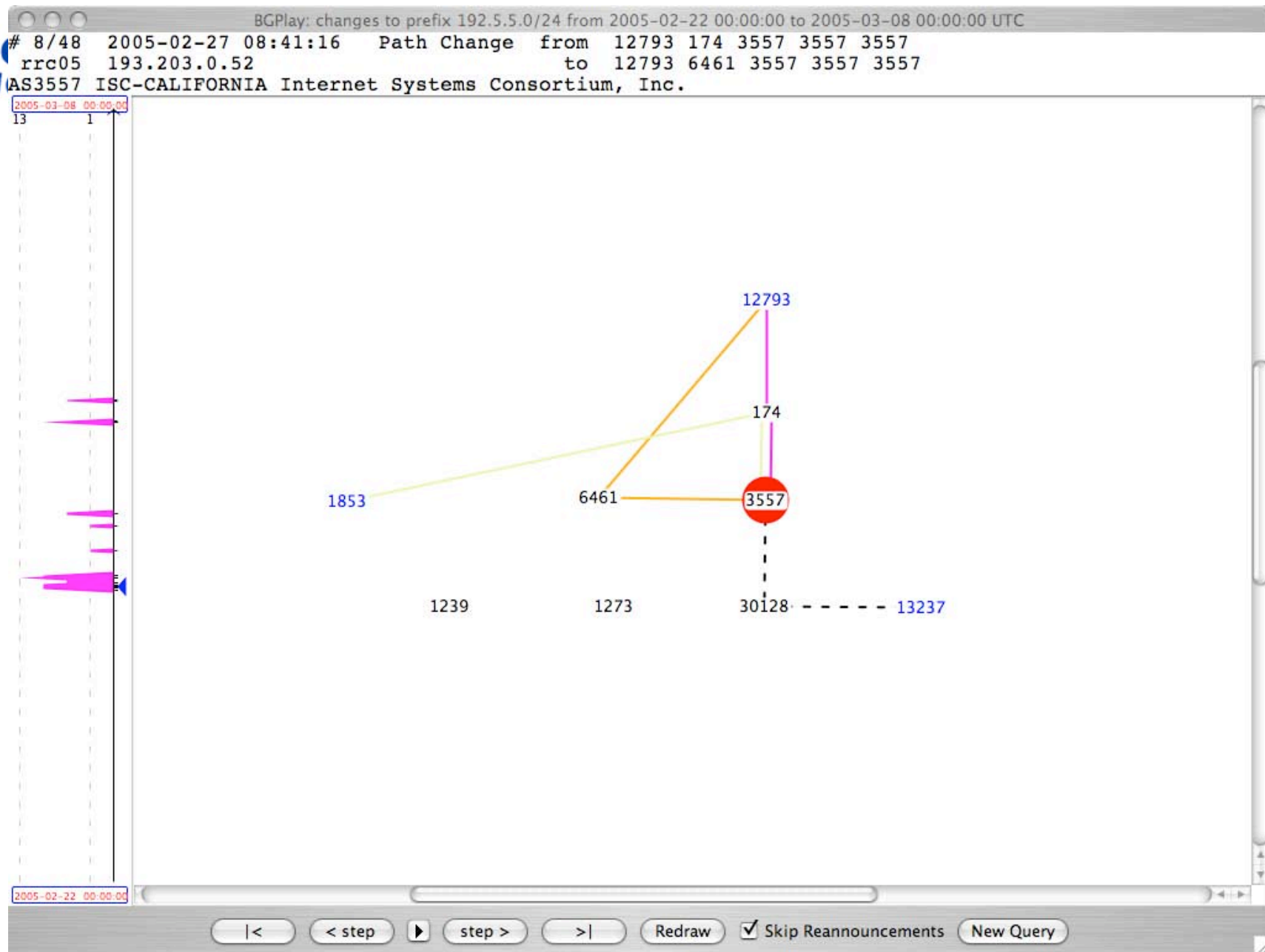


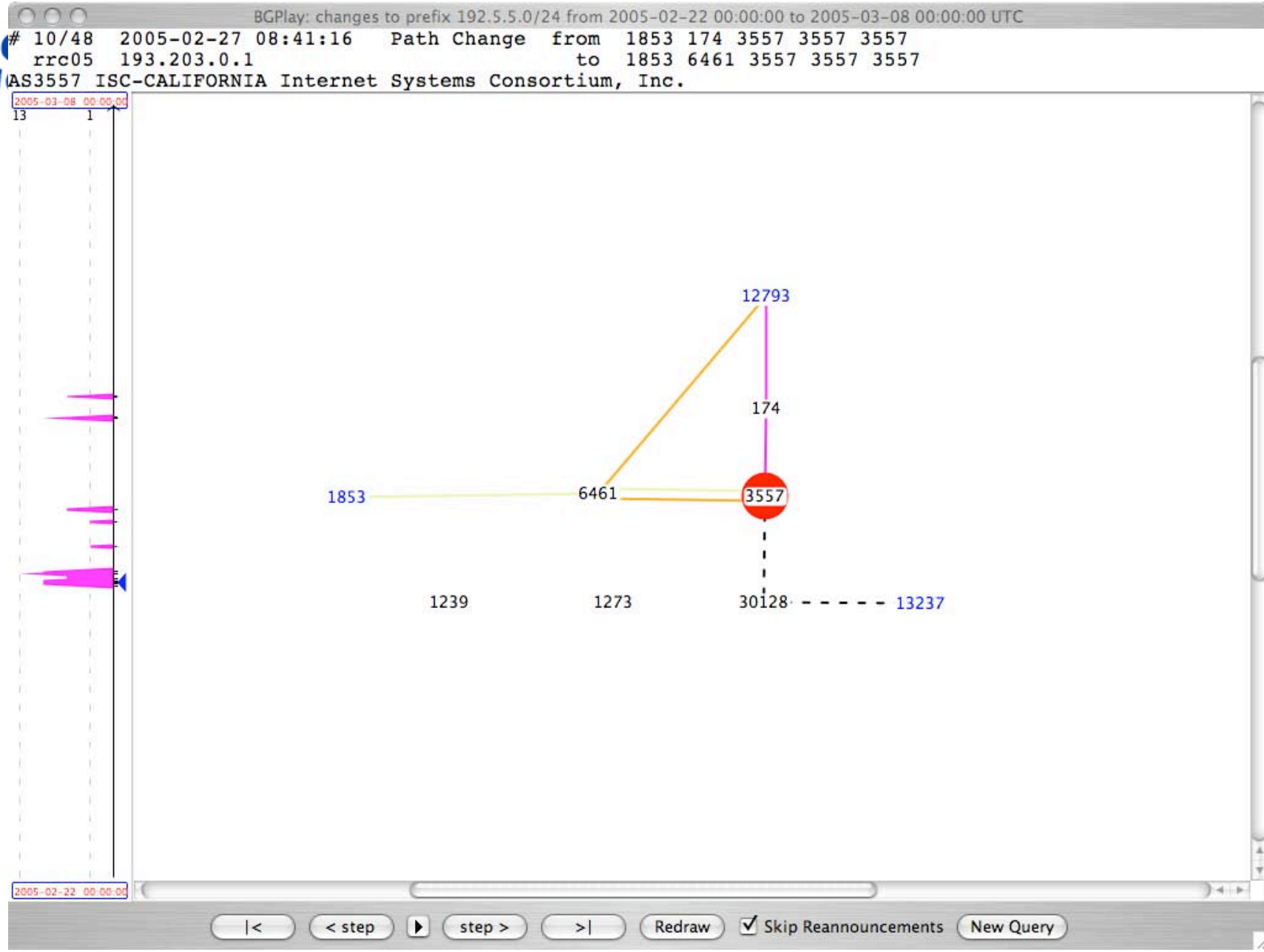


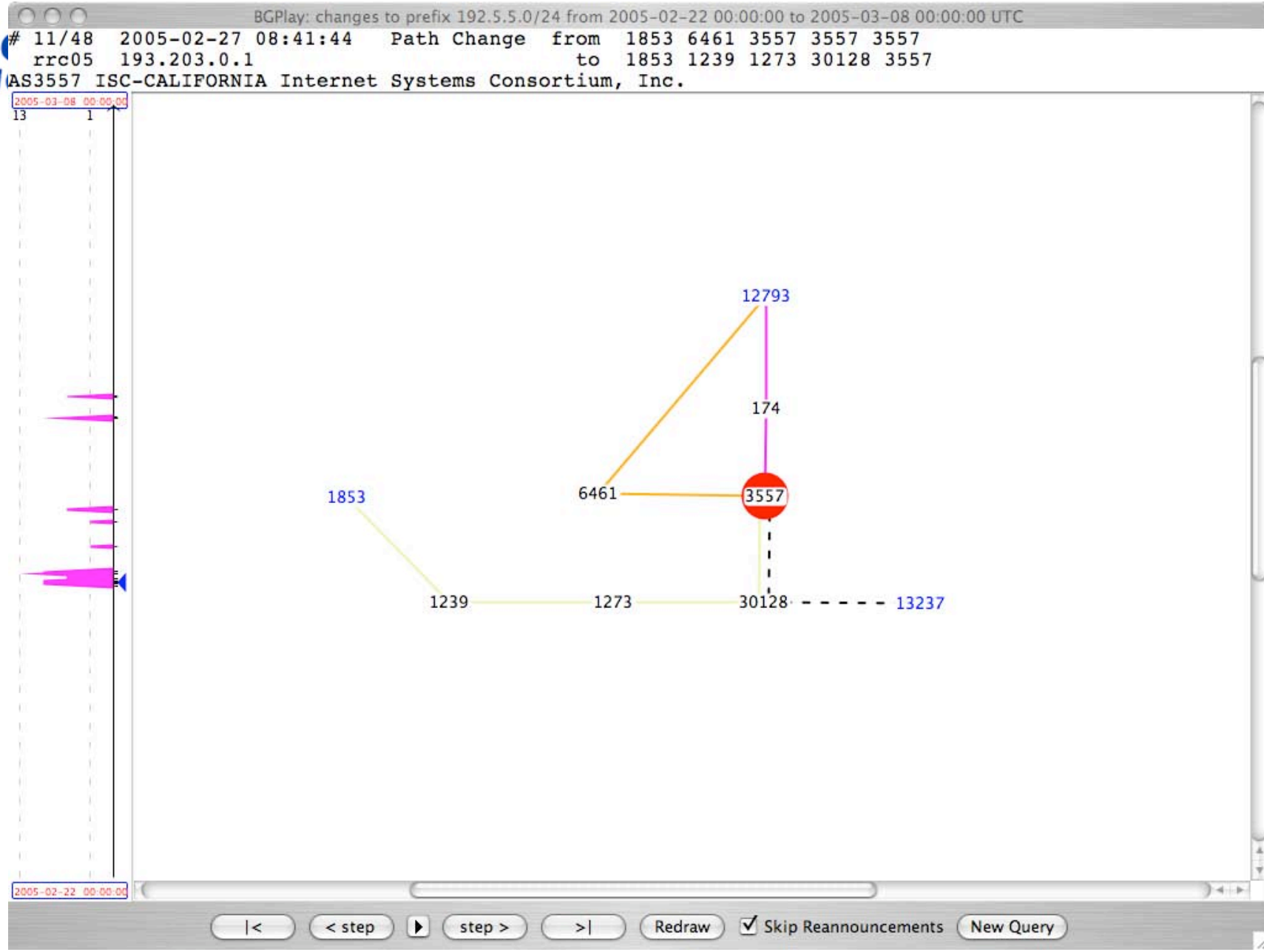


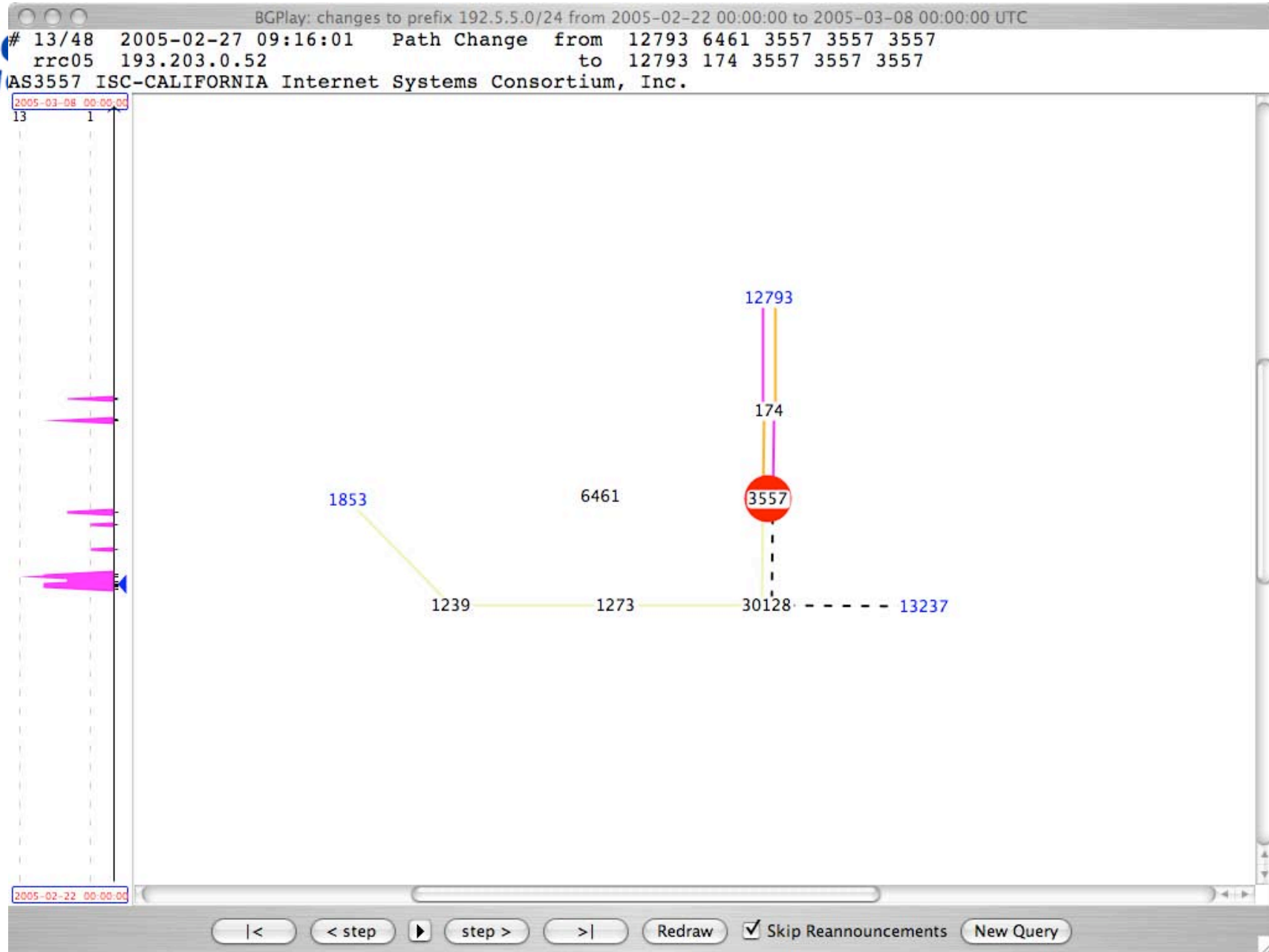


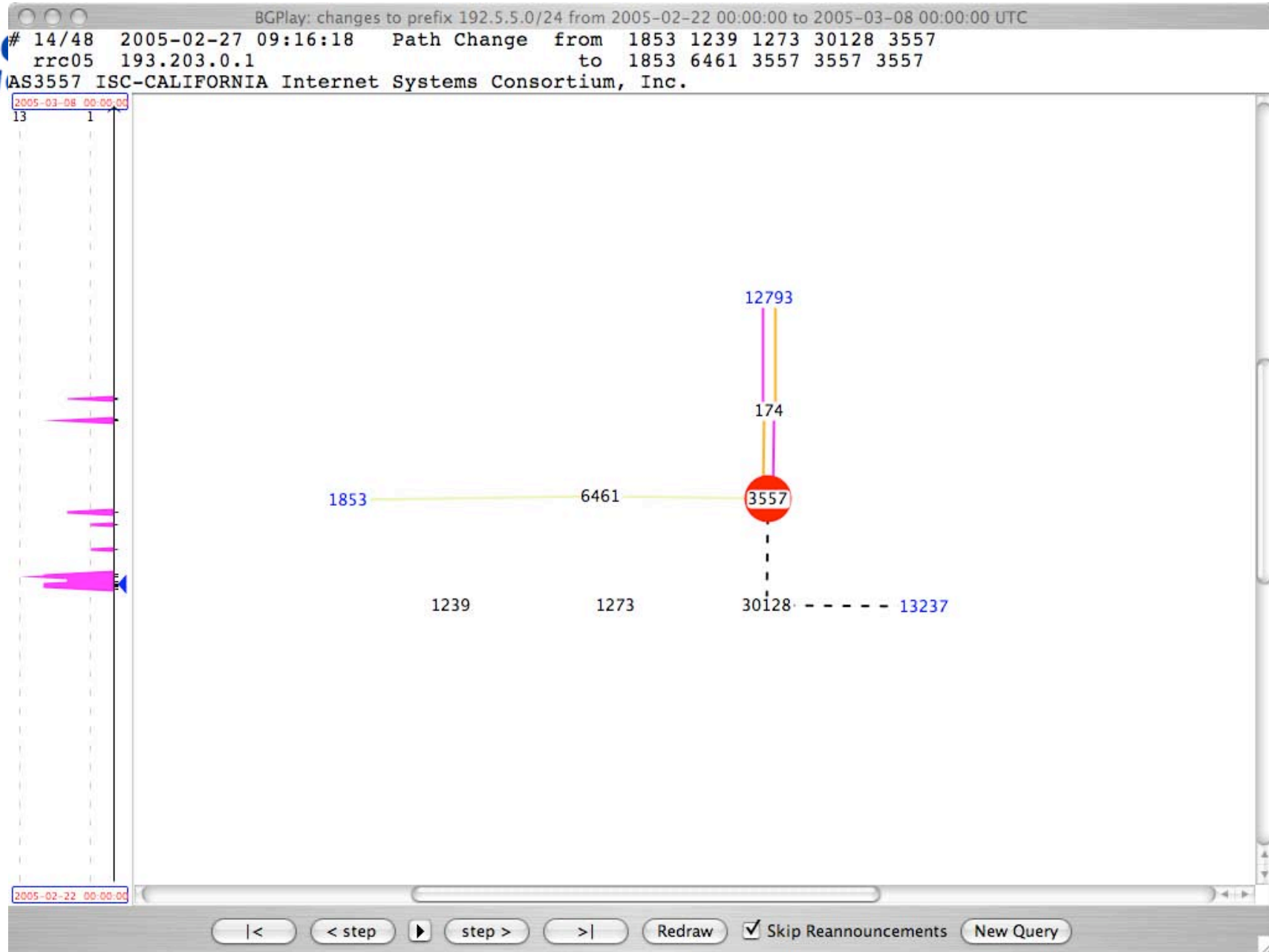


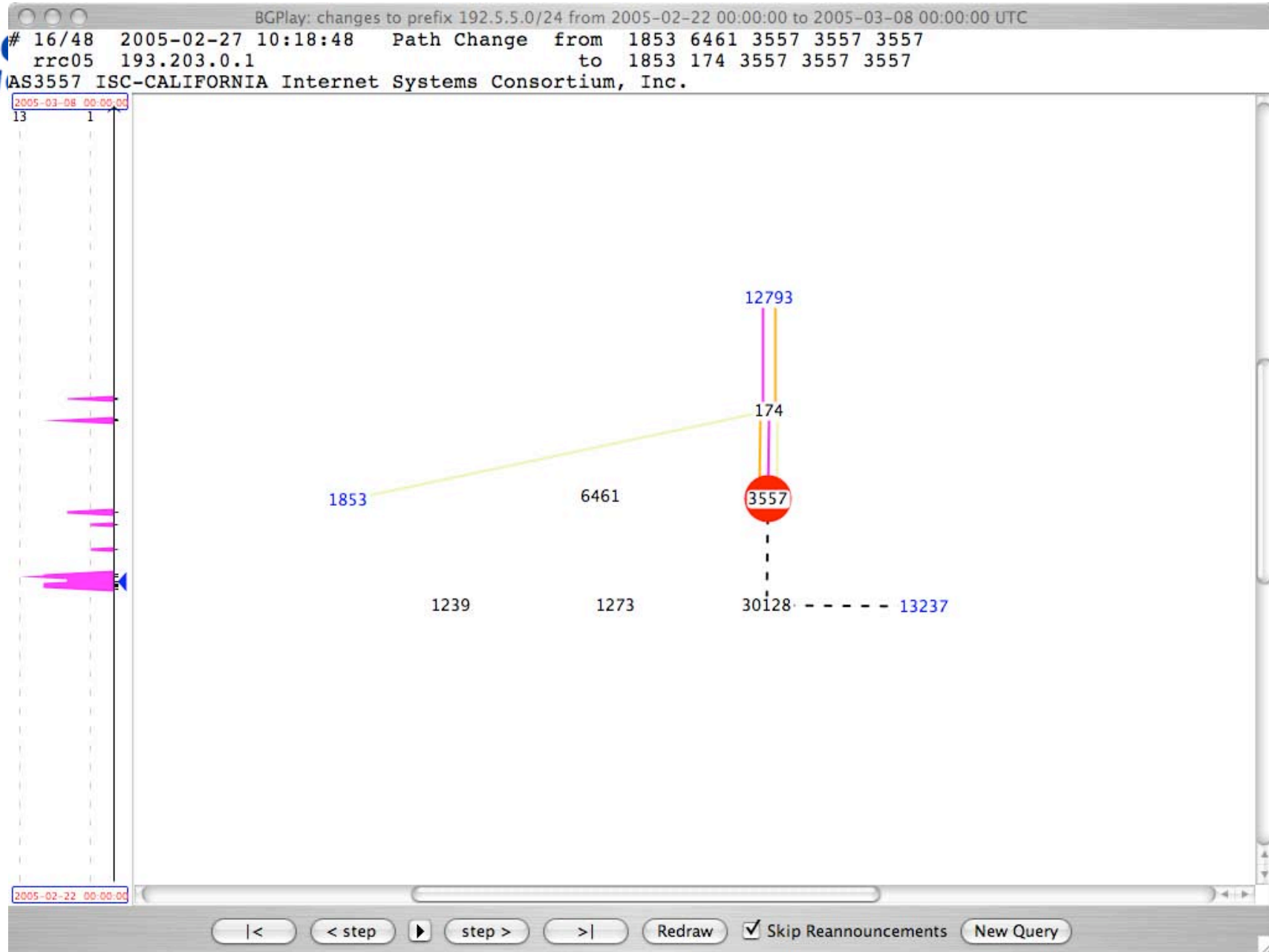


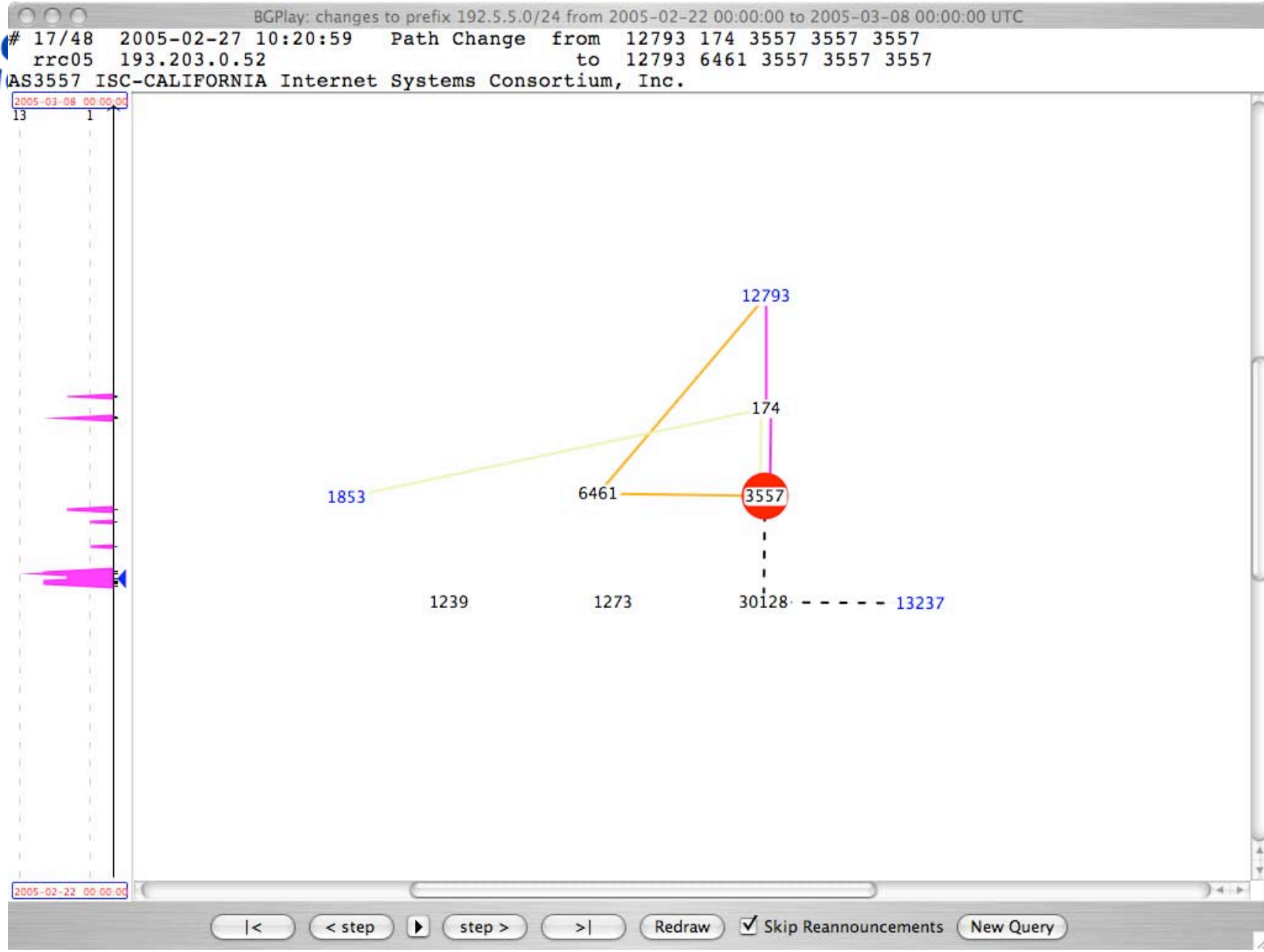


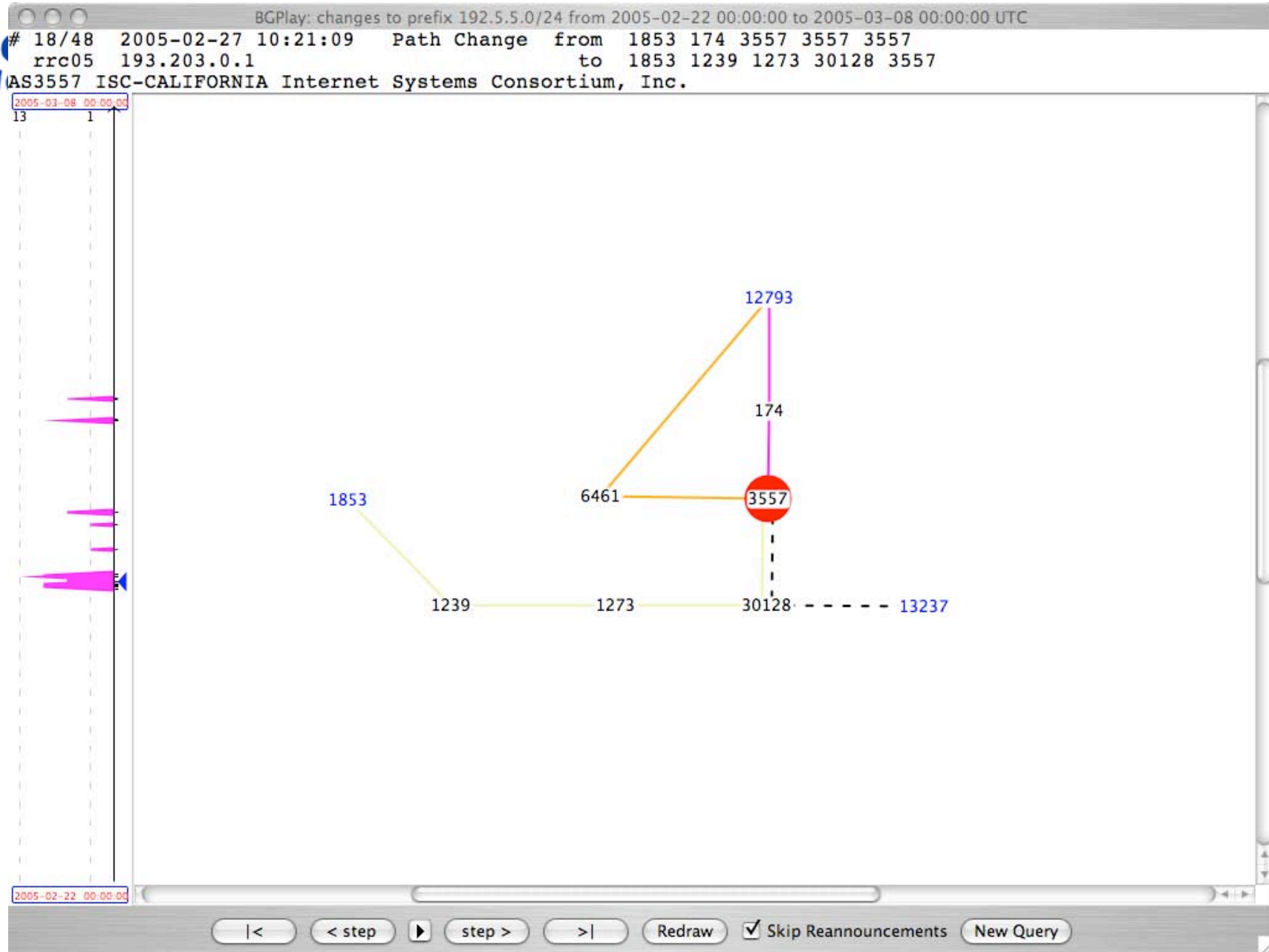


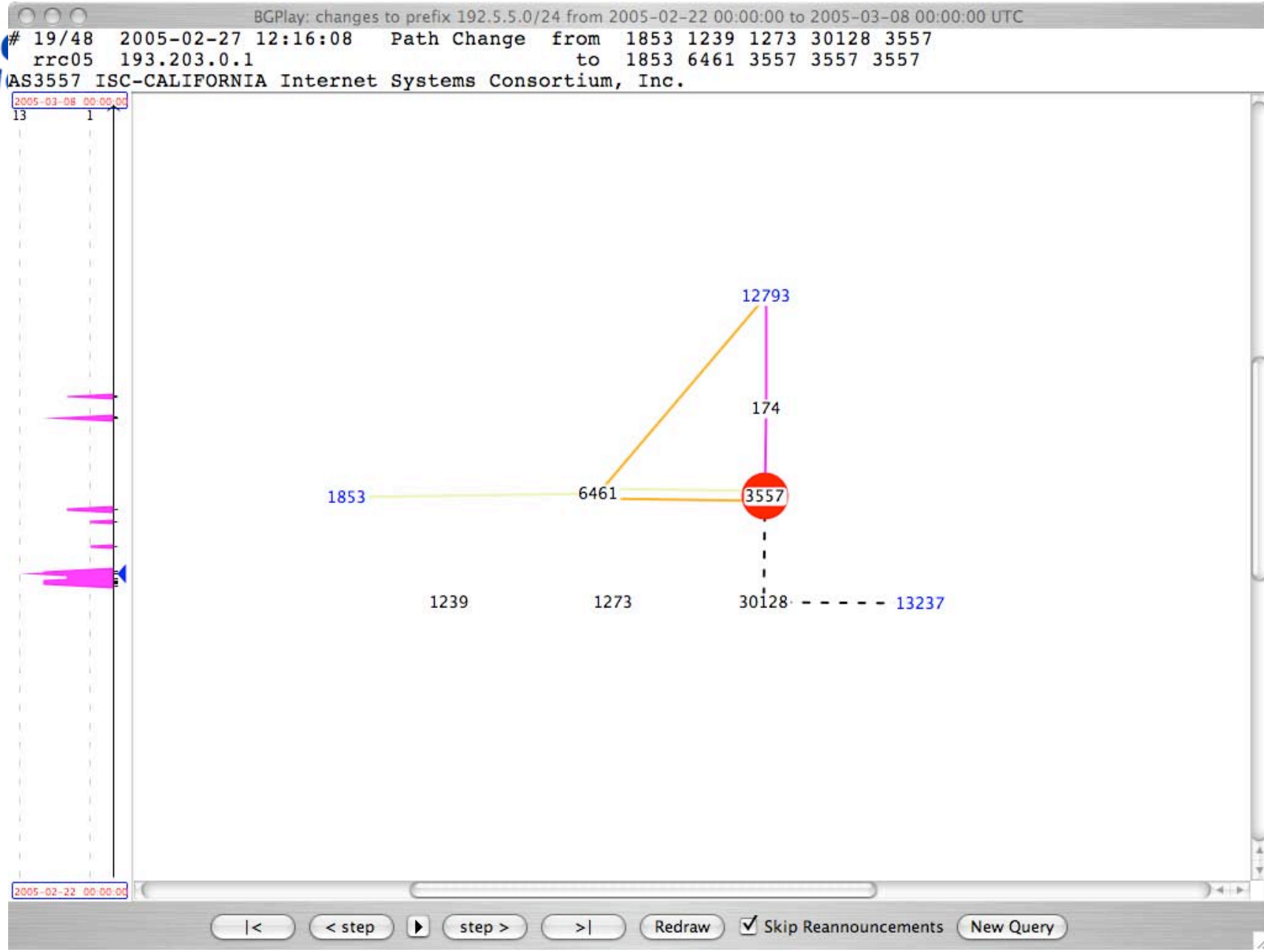


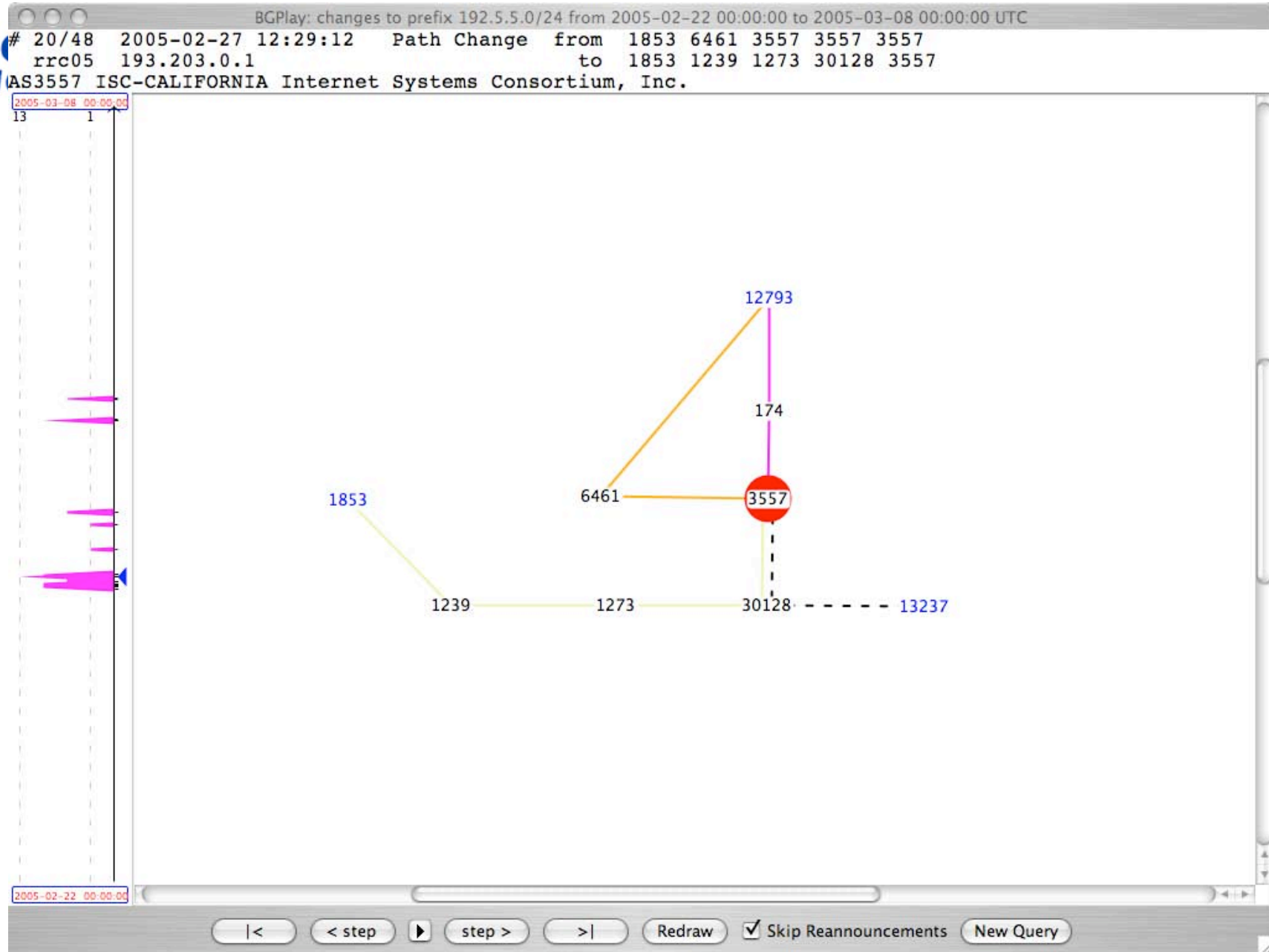


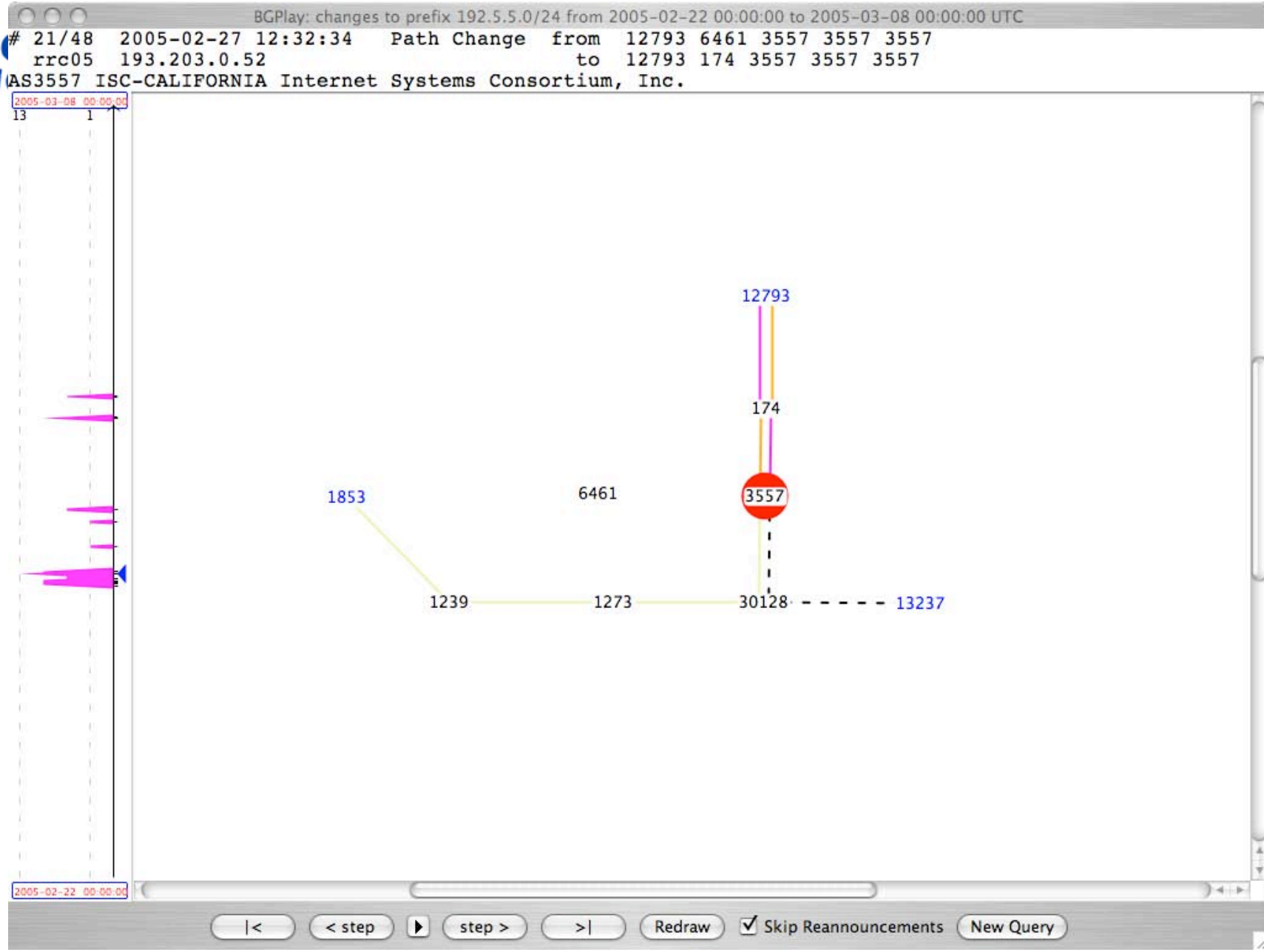


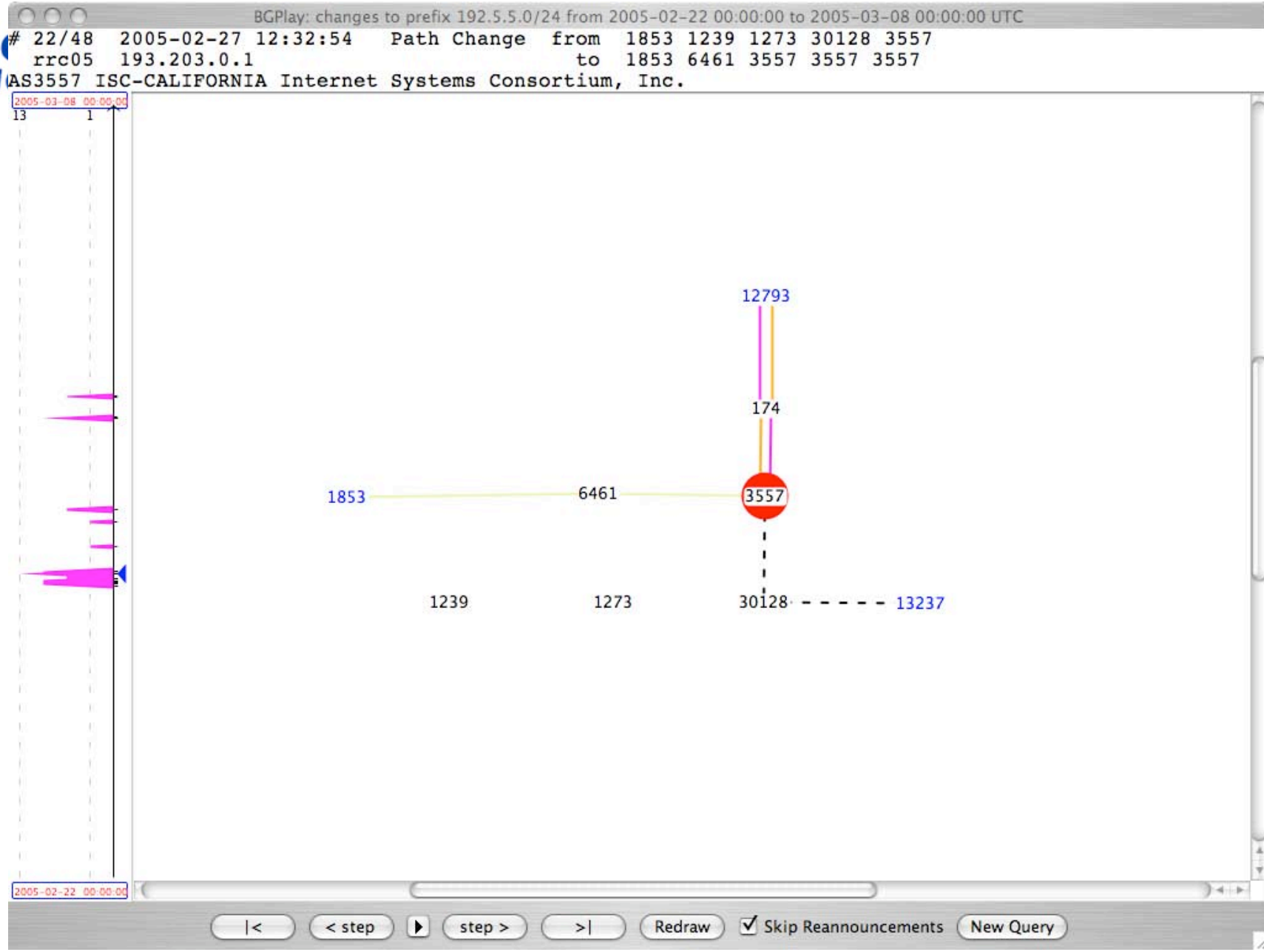


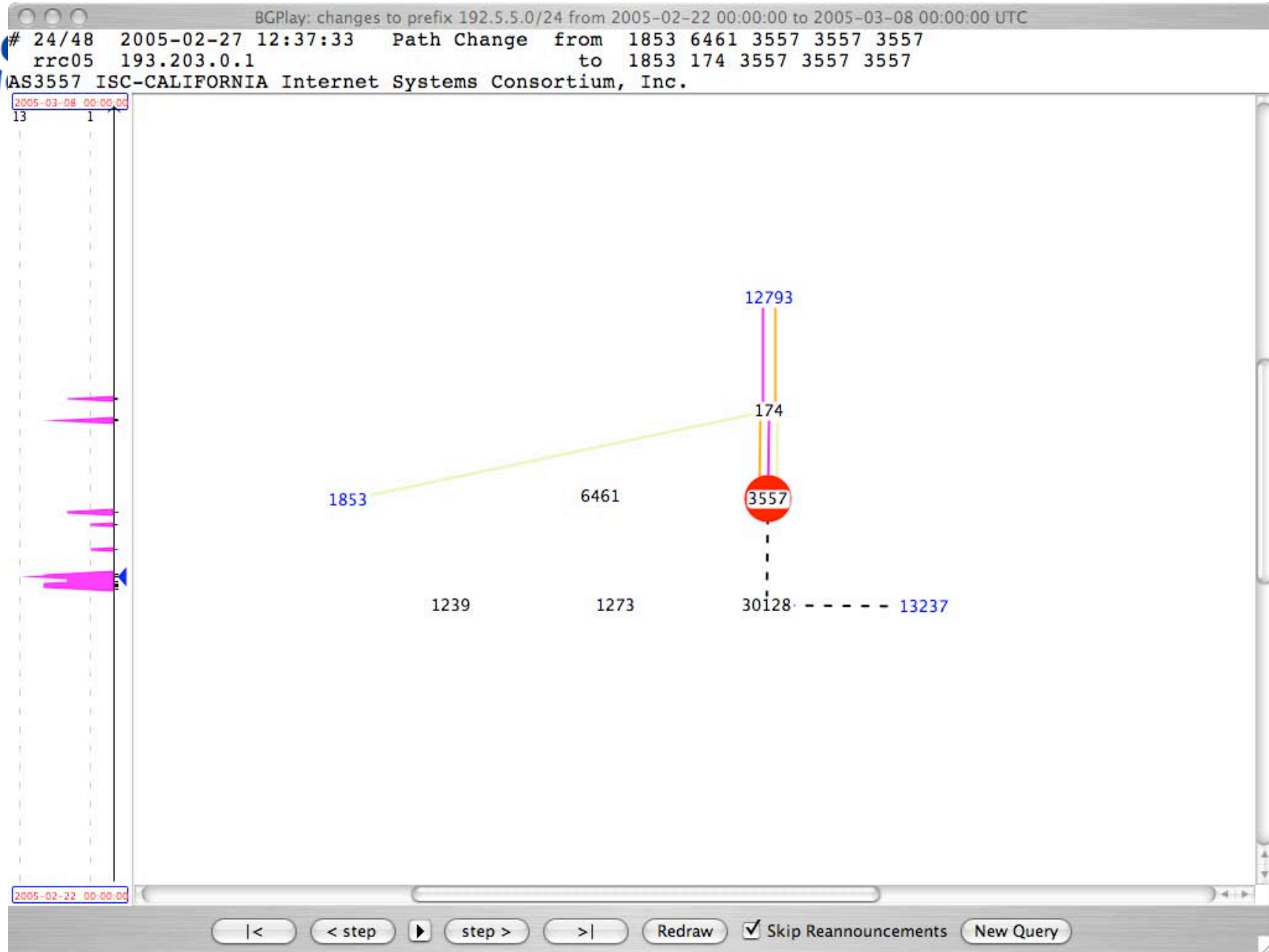


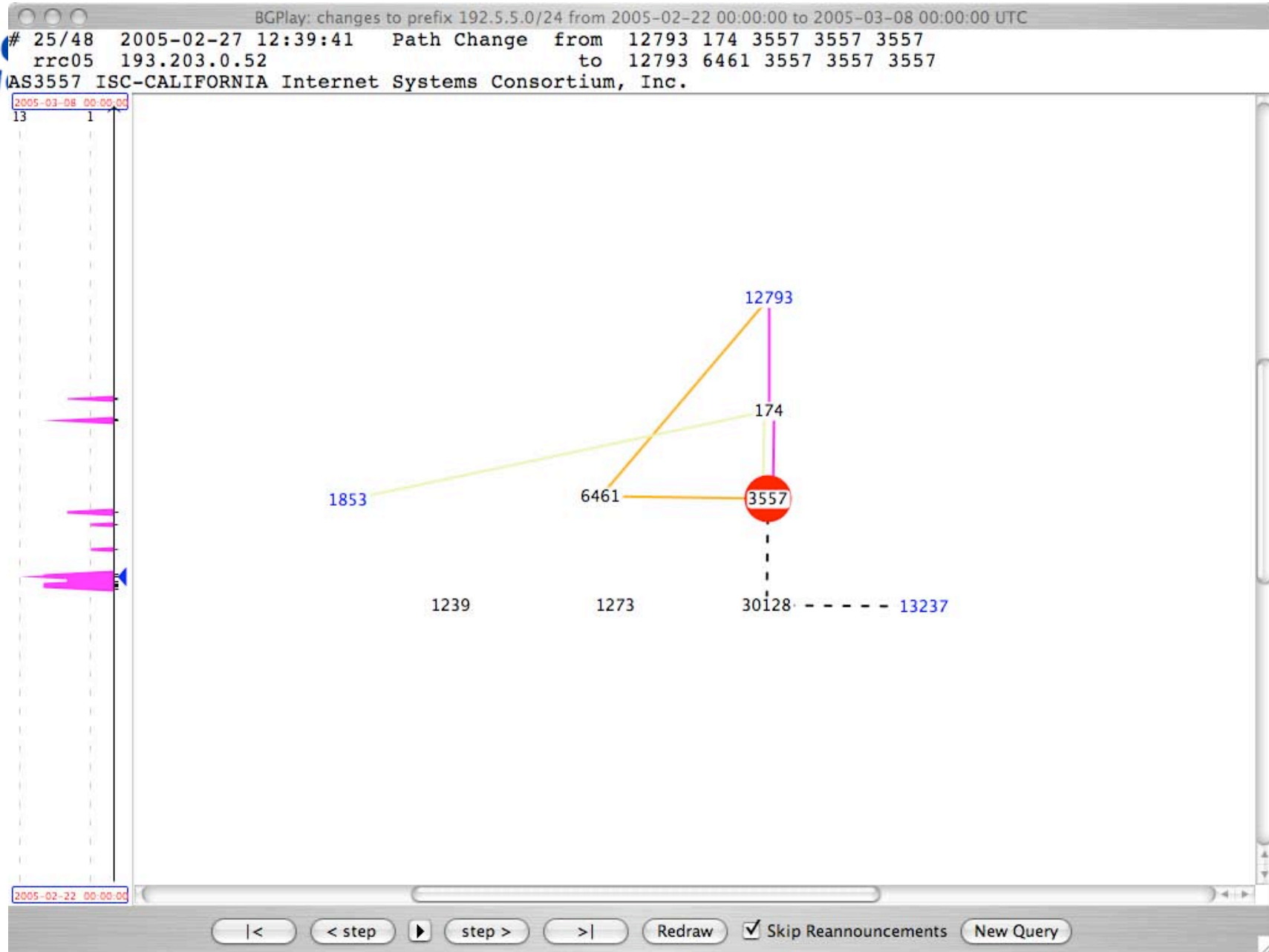


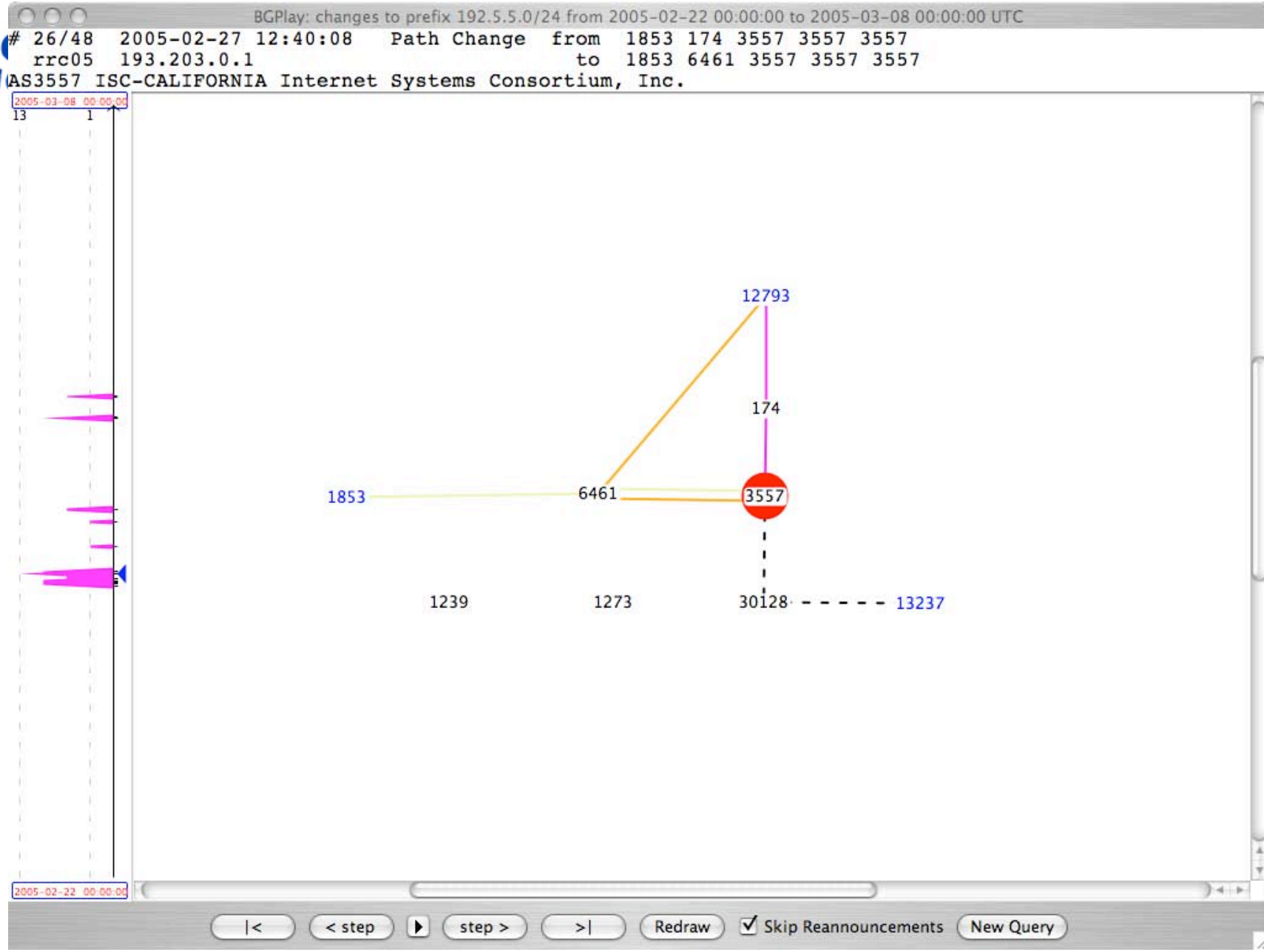


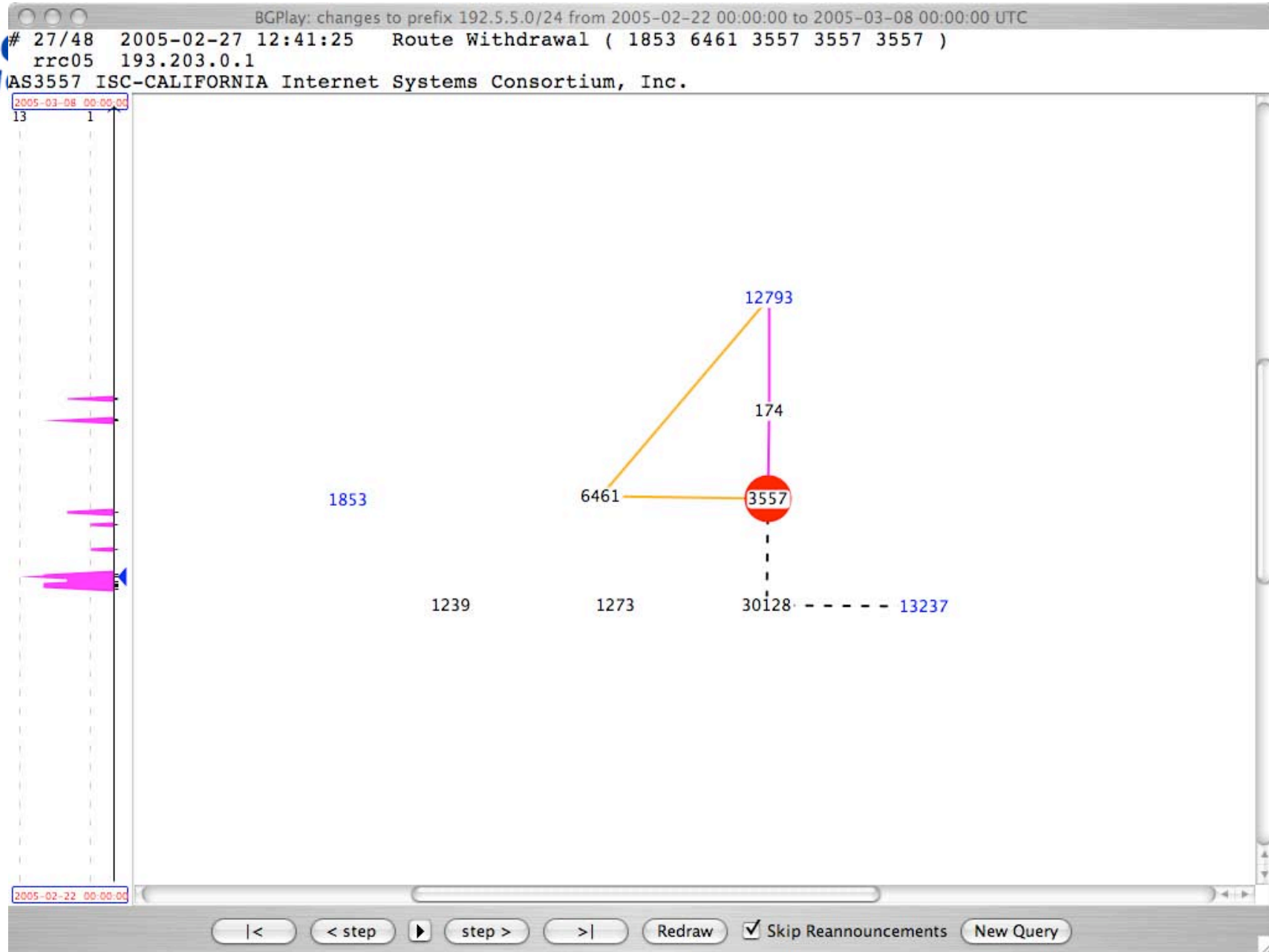


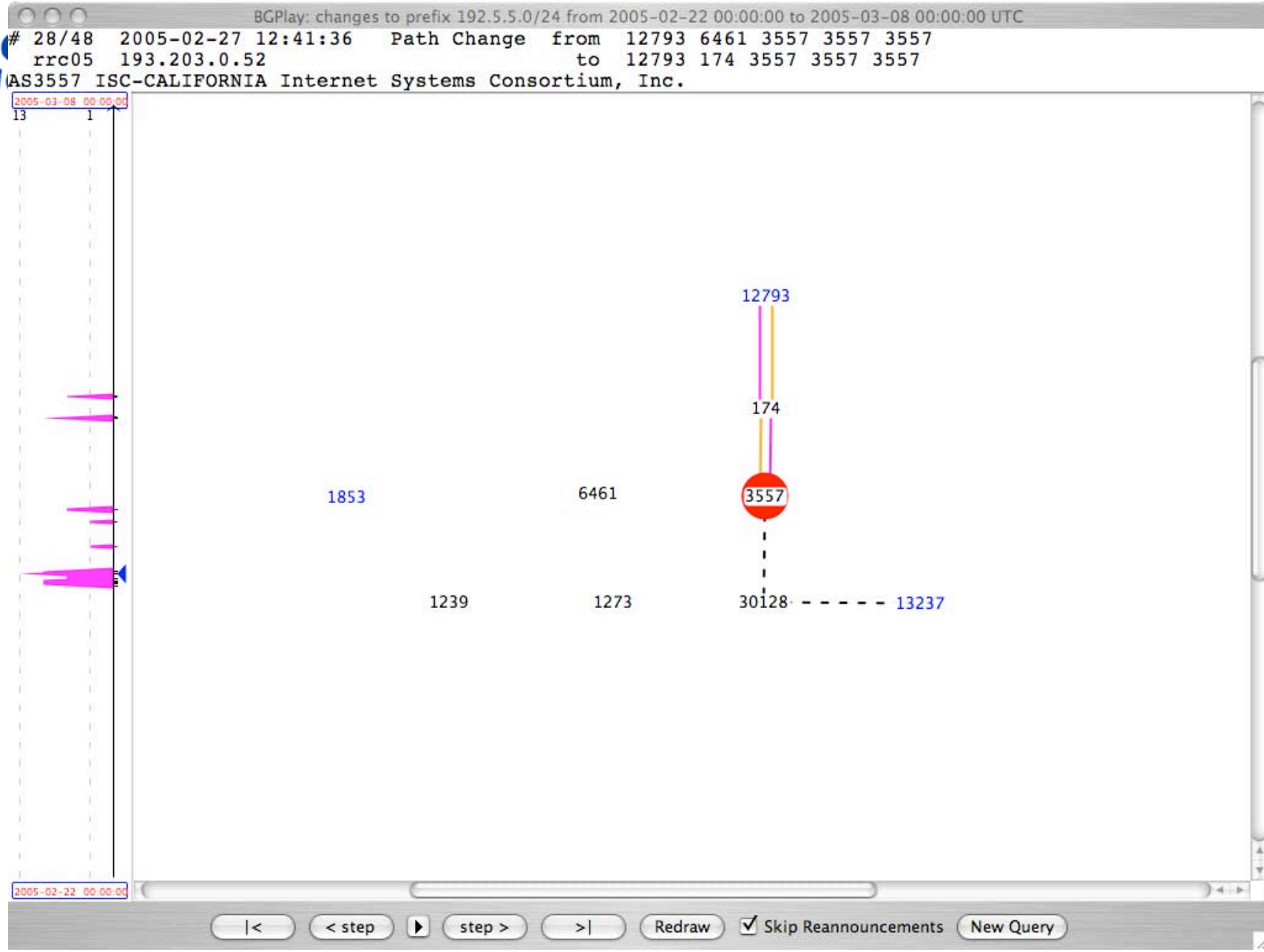


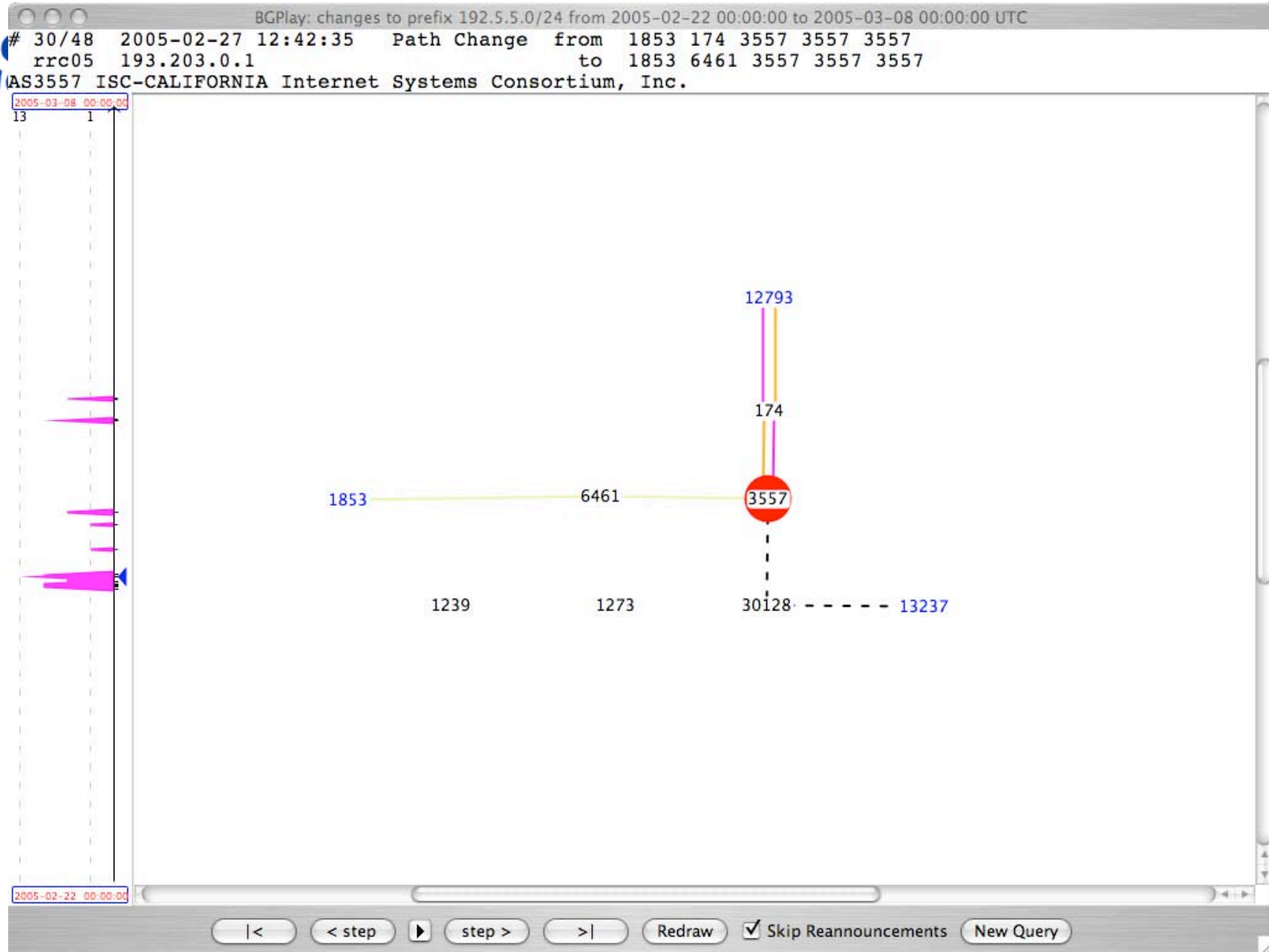


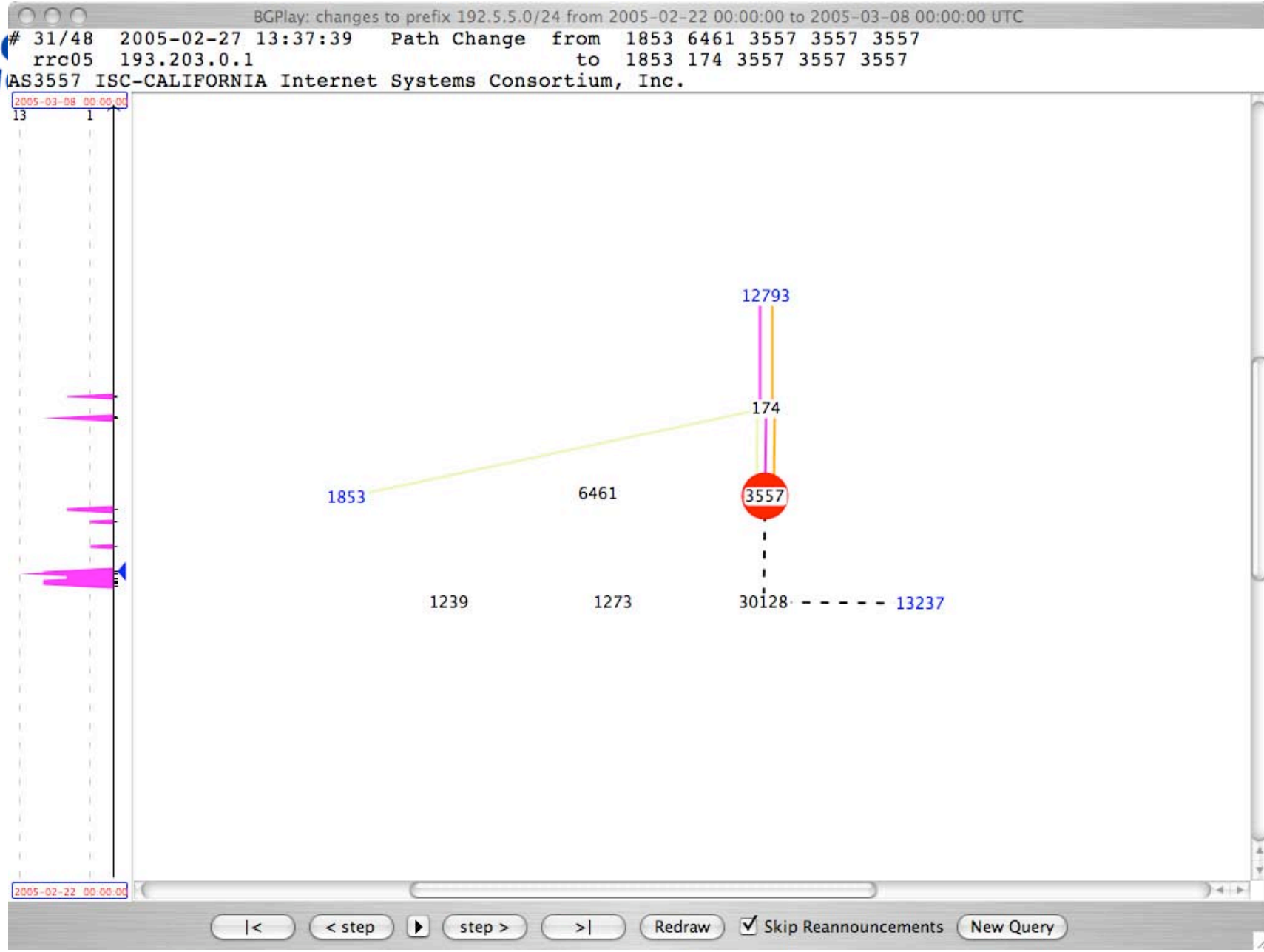














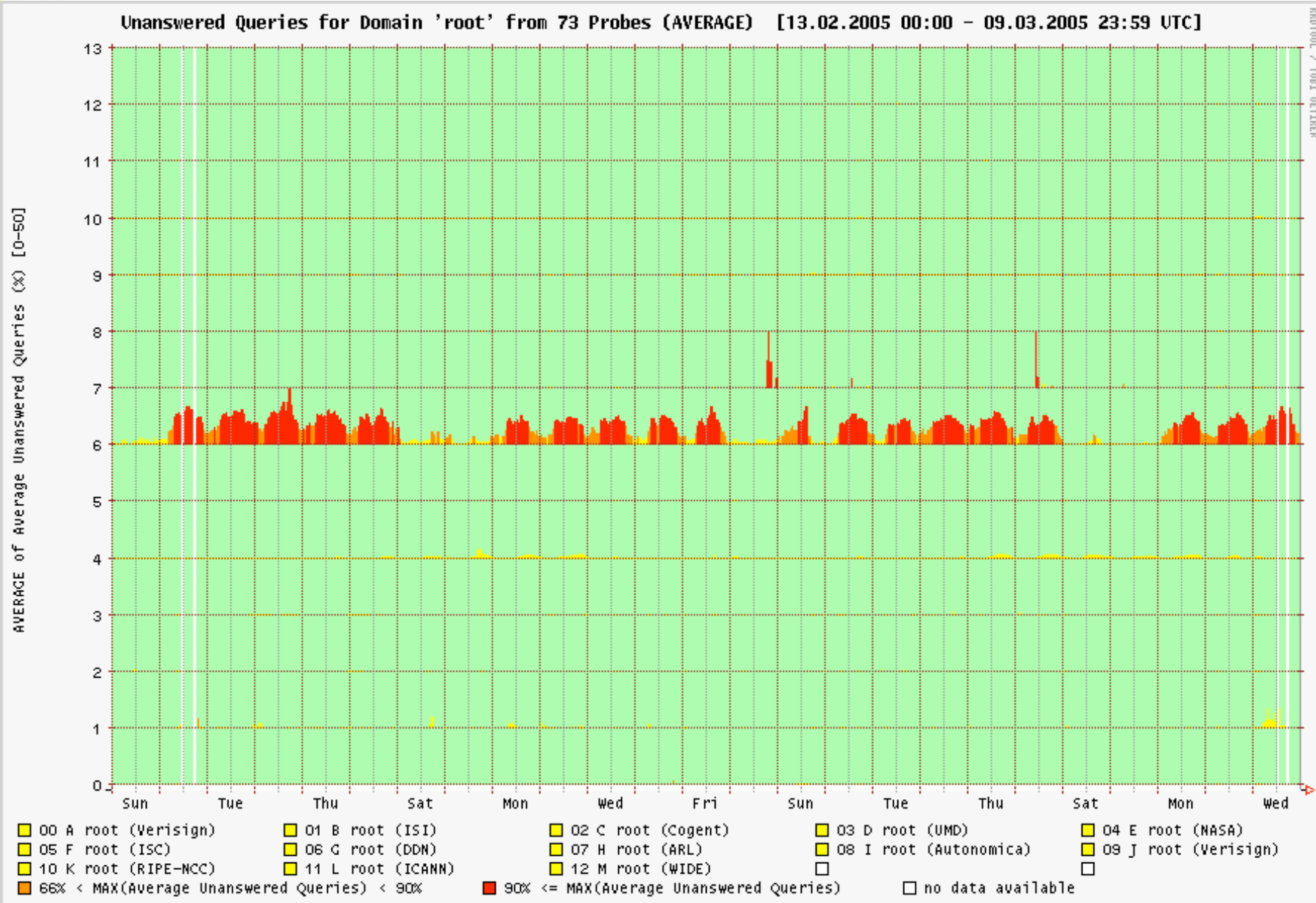
Some Questions for Looking Further

- Intuition suggests that there are more possible paths to anycast prefixes than to unicast prefixes. Is this true?
- Do anycast prefixes show more BGP paths of equal length than unicast prefixes?
- Might this be (one of the) causes of the large amount of instance switches observed by others?
- Or is this just normal BGP churn otherwise not visible?
- Or are there other factors?



Reader Warning

- This does not say much about DNS service quality!
- Instance switches per-se do not mean DNS service degradation at the anycast address, we observe lots of switches with no concurrent loss.
- In general DNSMON shows service at anycast addresses to be better than at unicast address. See next slide.
- To Randy: Better diagnostics need protocol work to report instance-ID together with answer on request.





Plea

This does **not mean that
anycast for DNS root service
is unstable or broken.**

**Please do not spread this false
rumor!**



Thanks

- Mark and Randy for continuing to look.
- TTM and RIS folk at RIPE NCC for the tools.
 - Especially Lorenzo and the Roma III folks for bgplay.
- Beate and the kids for giving me time to produce the movies. ;-)
- I have no intention of working further on this other than maybe present it at RIPE.