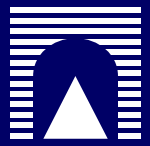


UNIVERSITÀ DEGLI STUDI ROMA TRE
Dipartimento di Informatica e Automazione

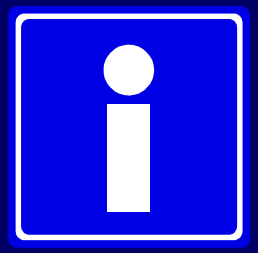
Digging Trustworthy

An IRR Analysis Service that Extracts BGP Peerings from the IRR

Tiziana Refice

RIPE 53 Meeting
2 October 2006
Amsterdam, The Netherlands

What's the Point with the IRR?



- Born with the purpose of supporting stable and consistent routing policies
- A valuable source of information to understand Internet routing

BUT…

- Maintained on a voluntary basis
- Information is often inconsistent and/or out-of-date
- Useful information is still there
- How to extract it? No systematic approach

Our Contribution(s)

- A method to extract BGP peerings (and more) from the IRR
  - An investigation of the constructions used to specify peerings
- An implementation of the method as an on-line service
- A proof of effectiveness by comparison with the state of the art
- A comparison with actual routing data

Extracting Peerings is Easy

```
aut-num: AS137
import: from AS20965 action pref=100;
        from AS1299 action pref=100;
        accept ANY
export: to AS1299 announce AS-GARR
changed: noc@garr.it 20000830
source:  RIPE
```

Thank you!

Questions?

Extracting Peerings is Not That Easy

- Structured policies
- Set objects
- Complex expressions
- Multi-protocol extensions

```
aut-num: ASX5
import: { from ASX2:AS-Z2 accept 100.0.0.0/8;
        } refine {
          from ASX1 ASX2 accept 100.1.0.0/16;
        } except {
          from ASX3 accept 100.1.1.0/24;}
export: to ASX1:PRNG-Y1
        to ASX1:AS-Z1 except ASX9
        announce 100.1.1.0/24
mp-export: to ASX11 at 2001::1 announce 2001::/48
default: to ASX12 action pref=10
default: to ASX13 100.1.1.1 at 100.1.1.2
```

Extracting Peerings is Not That Easy

- Existing tools (e.g., the RIPE Routing Registry Consistency Check) do not deal with these constructions

```
aut-num: ASX5
import: { from ASX2:AS-Z2 accept 100.0.0.0/8;
        } refine {
          from ASX1 ASX2 accept 100.1.0.0/16;
        } except {
          from ASX3 accept 100.1.1.0/24;}
export: to ASX1:PRNG-Y1
        to ASX1:AS-Z1 except ASX9
        announce 100.1.1.0/24
mp-export: to ASX11 at 2001::1 announce 2001::/48
default: to ASX12 action pref=10
default: to ASX13 100.1.1.1 at 100.1.1.2
```

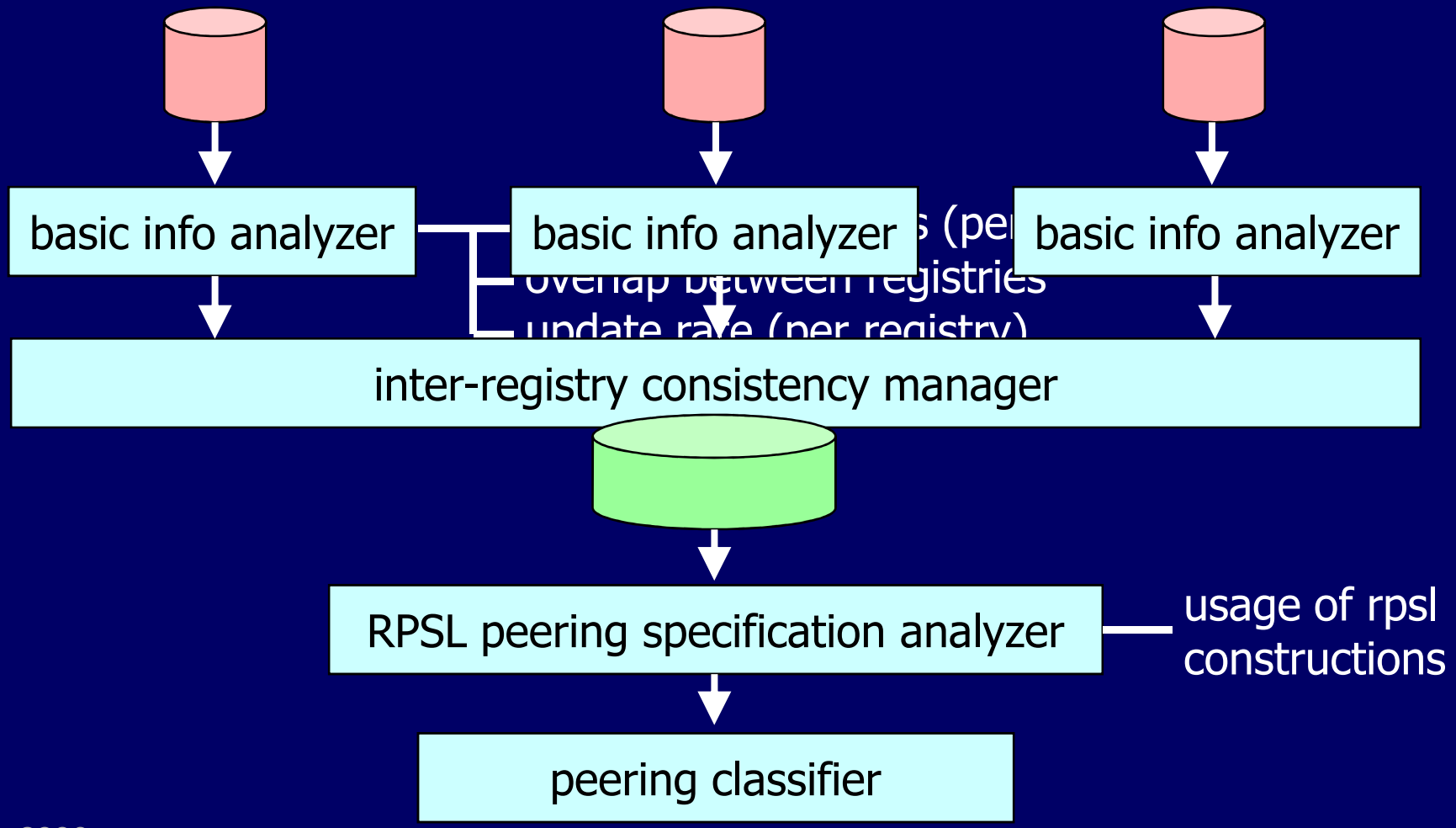
Extracting Peerings is Not Easy At All

```
aut-num: AS24336

import:  from AS17685
         accept ANY
export:  to AS17685
         announce AS24336

changed: matsuo@po.d-b.ne.jp
         20050220
source:  RADB
```

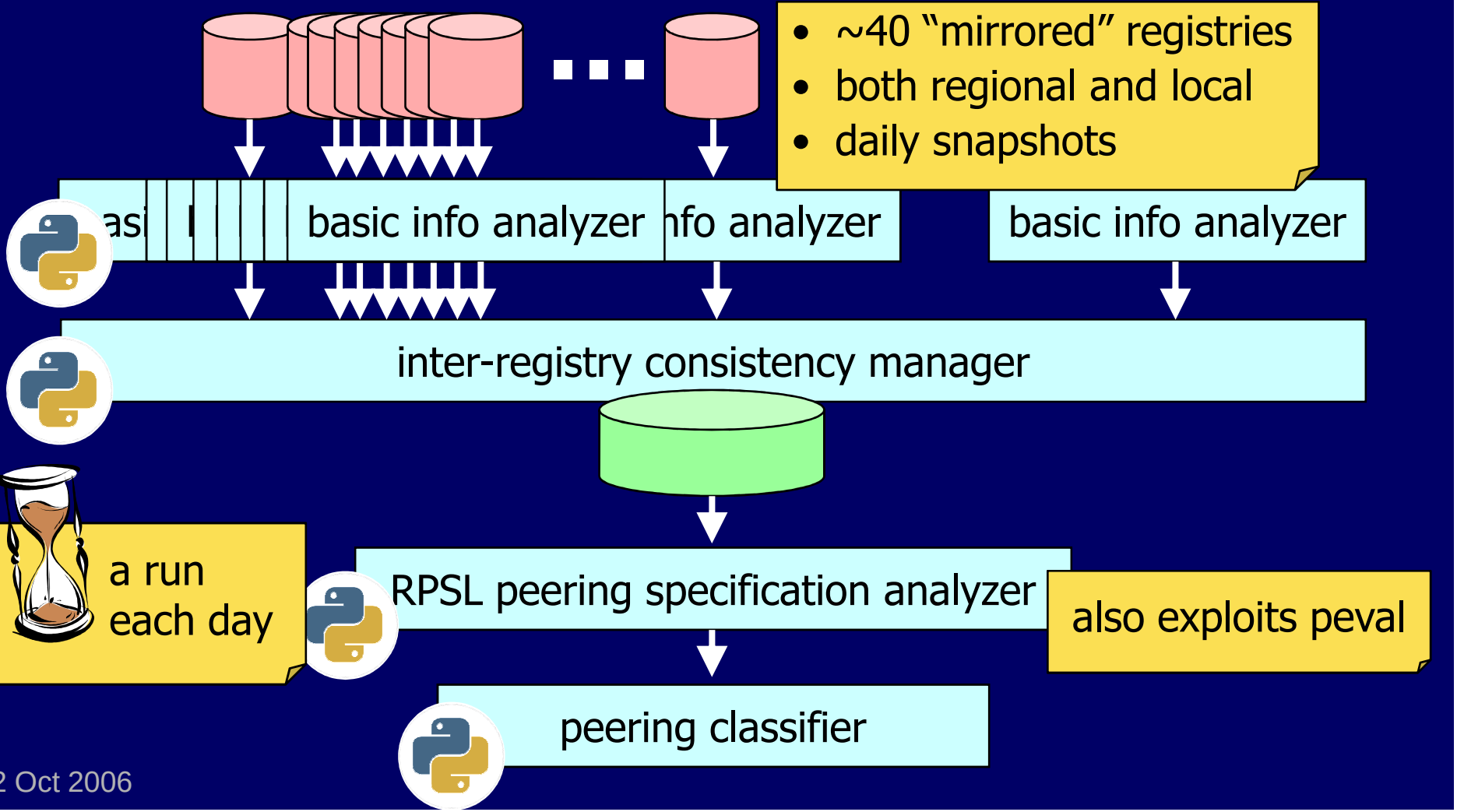
```
aut-num:  AS24336

import:   from AS17685
          action pref=100; accept ANY
import:   from AS7682
          action pref=100; accept ANY
export:   to AS17685
          announce AS24336
export:   to AS7682
          announce AS24336

changed:  hm-changed@apnic.net
          20050210
source:   APNIC
```

1. Identify stubs
2. Look at the last update timestamp
3. Consider highest ranked (i.e. largest) registry

An Overview of the Method



From the Method to the On-Line Service



The Peering Classifier


```
aut-num: ASX1                    aut-num: ASX2
import: from ASX2
        accept ASX2
```

- Build topologies with different levels of confidence

A Look at the Data

- 68 registries downloaded on 03/31/06
  - ftp://ftp.ripe.net/ripe/dbase
  - ftp://ftp.radb.net/radb/database
- Overlapping **`aut-nums`**

	# of **`aut-nums`**	ripe	apnic	radb	arin	verio
ripe	11468	11238	19	50	7	23
apnic	3299	19	2688	423	1	113
radb	2695	50	423	2037	37	45
arin	555	7	1	37	463	14
verio	498	23	113	45	14	310

A Look at the Data

- 68 registries downloaded on 03/31/06
  - ftp://ftp.ripe.net/ripe/dbase
  - ftp://ftp.radb.net/radb/database
- Overlapping **aut-nums**

	# of left after aut-nums removing		ripe	apnic	radb	arin	verio
ripe	11468	82%	1128	19	50	7	23
apnic	3298	84%	19	2688	423	1	113
radb	2695	57%	50	423	2037	37	45
arin	555	51%	7	1	37	463	14
verio	498	62%	23	113	45	14	310

Extracted Peerings

	peerings
This work	**236,663**
RIPE RRCC	108,521
[1] (RIPE only)	56,949
[2] (RIPE only)	70,222
[3]	127,498

[1] P. Mahadevan et al., *The Internet AS-Level Topology: Three Data Sources and One Definitive Metric*. SIGCOMM Computer Communication Review, 2006.

[2] B. Zhang et al., *Collecting the Internet AS-Level Topology*. SIGCOMM Computer Communication Review, 2005.

[3] G. Siganos et al., *Analyzing BGP Policies: Methodology and Tool*. INFOCOM 2004.

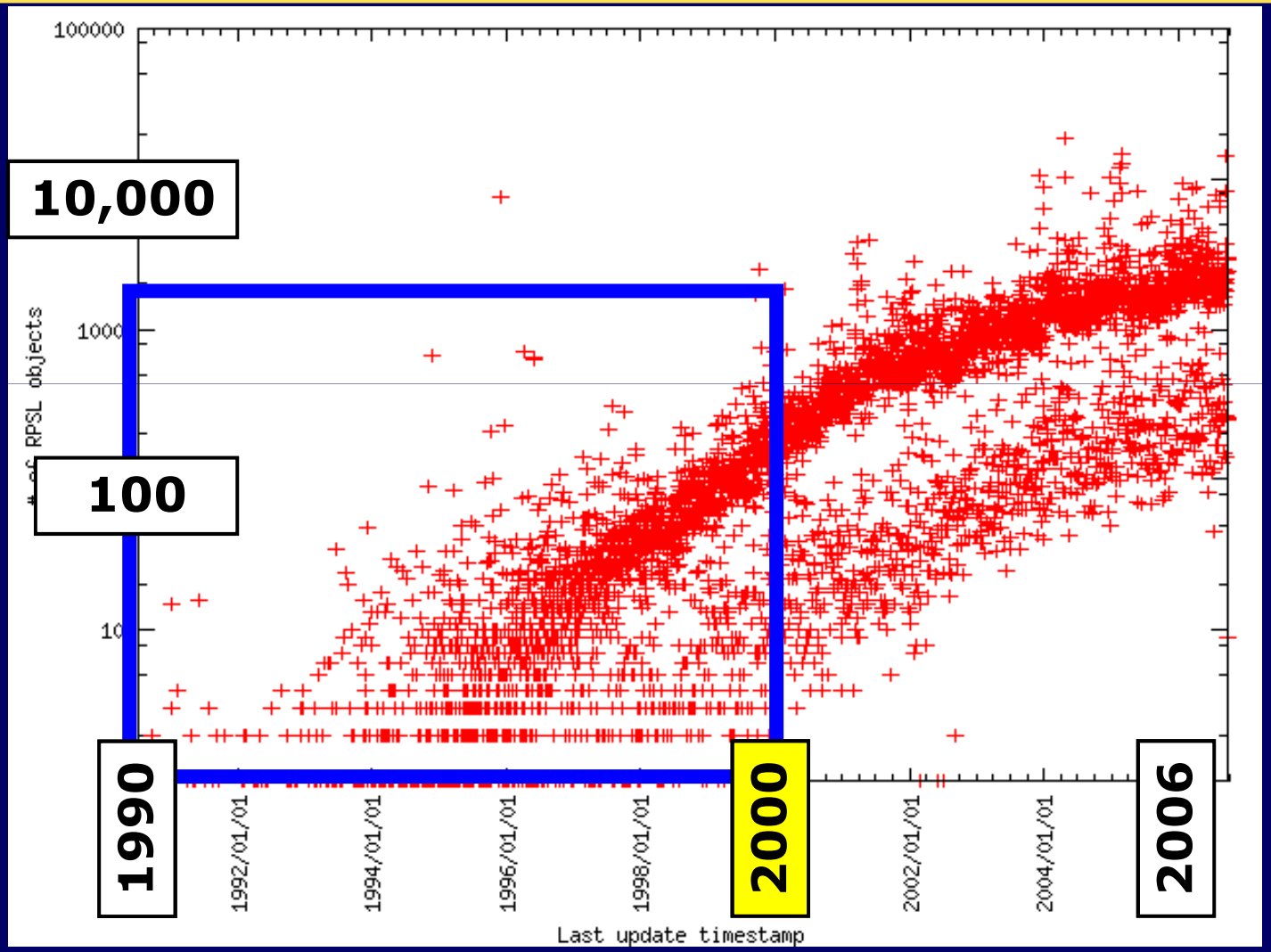
Extracted Peerings

peering type	count	from RIPE only
HALF_SIDE	143,342	58.4%
FULL	42,599	94.6%
1/4_E	34,155	7.7%
1/4_I	13,997	23.7%
3/4_NOT_E	1,373	80.3%
3/4_NOT_I	1,013	82.2%
HALF	114	57.9%
1/2_I	51	66.7%
1/2_E	19	47.4%

I just don't like figures!!

...yet I do like graphs!

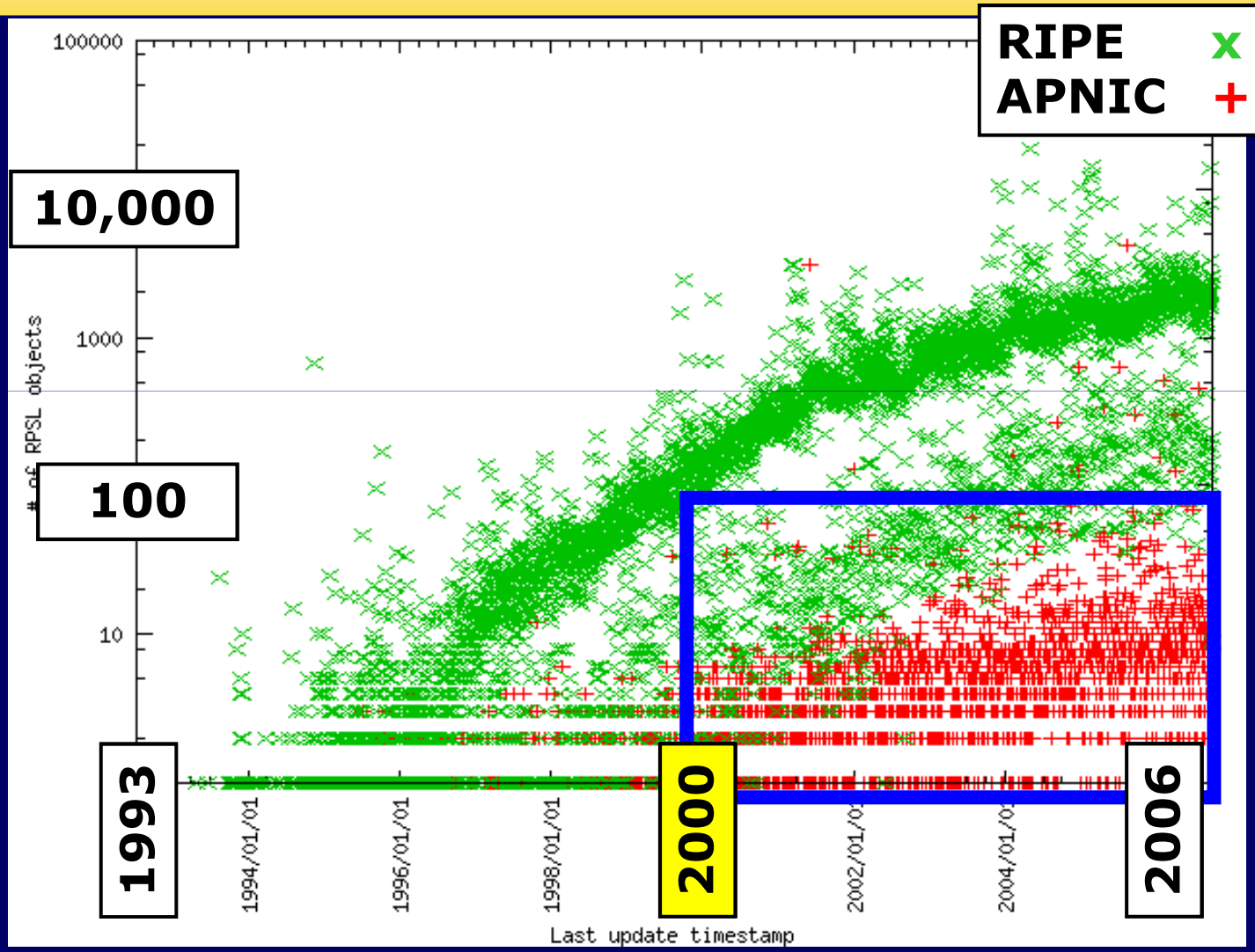
Timestamp Distribution



# of RPSL objects

Last update timestamp

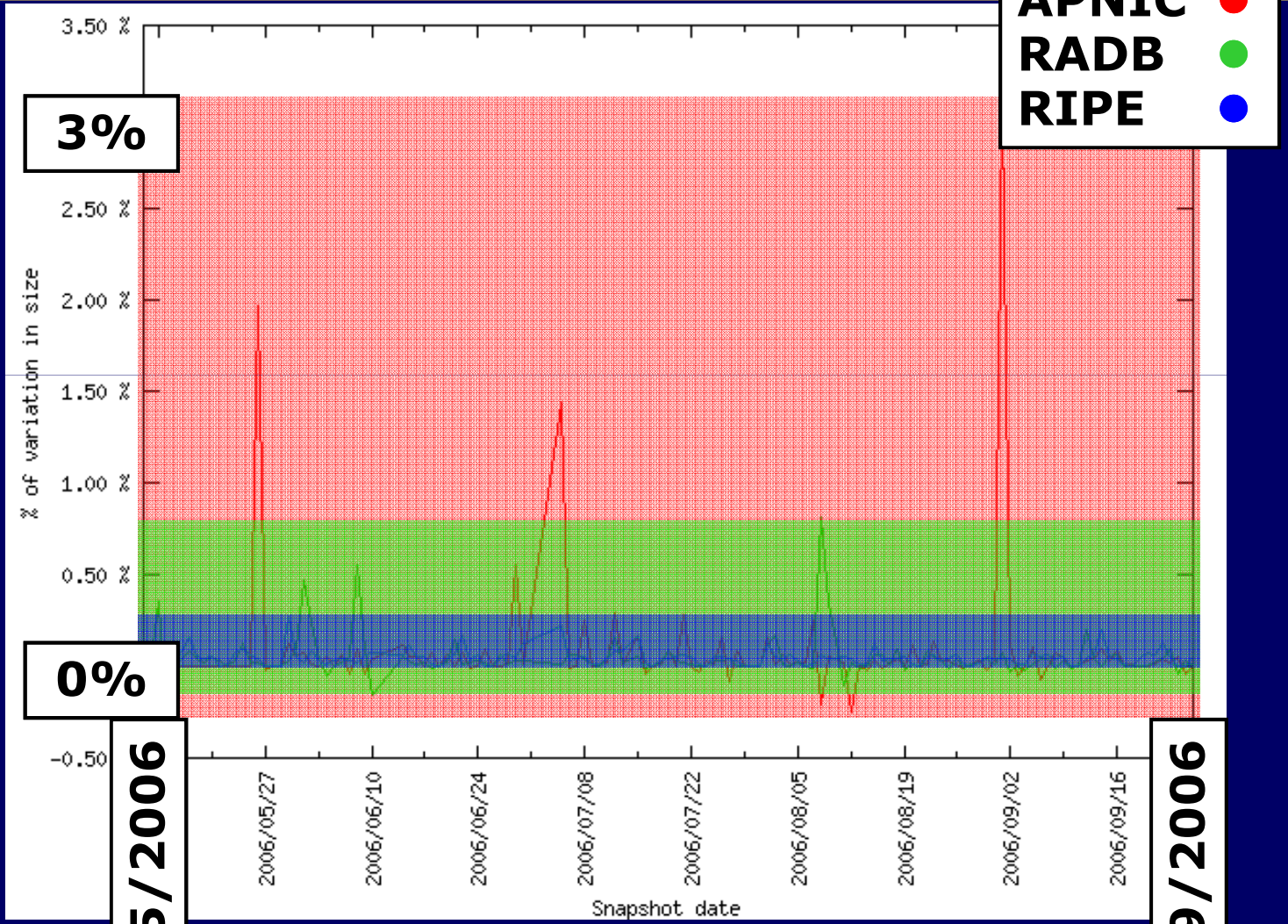
Timestamp Distribution



# of RPSL objects

Last update timestamp

Growth



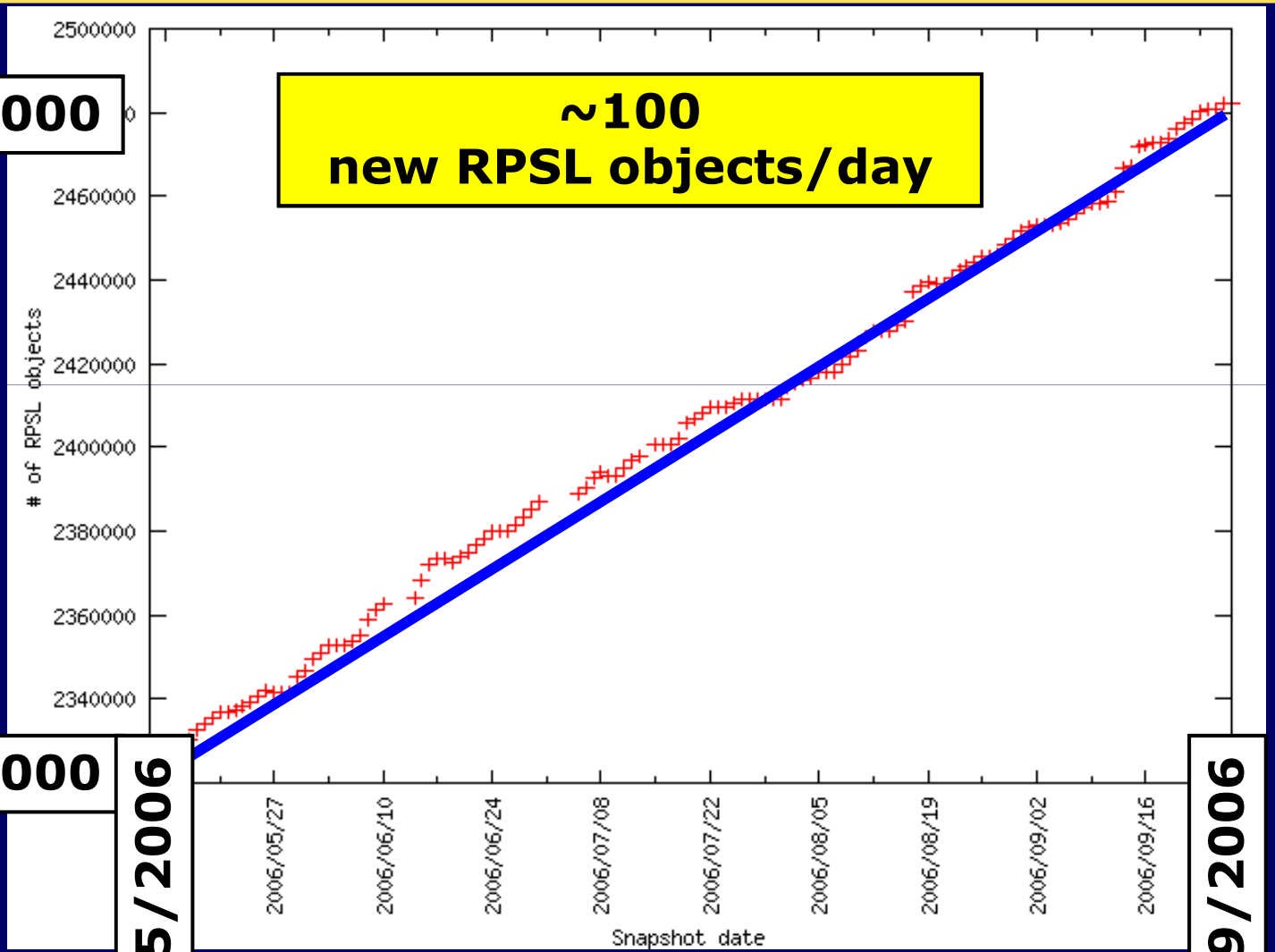
% of variation in size

05/2006

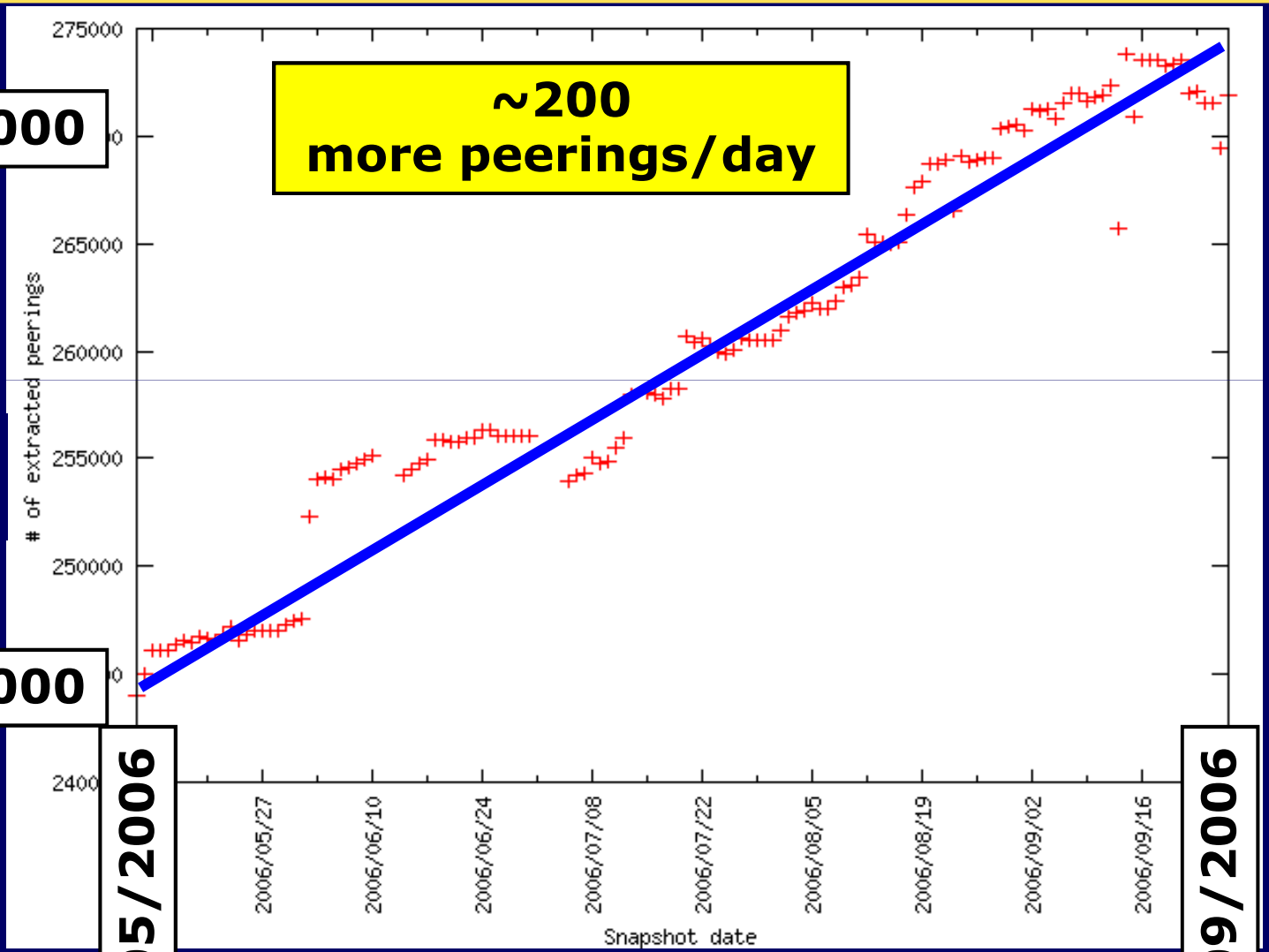
09/2006

Snapshot date

Growth


248,000
~100
new RPSL objects/day
# of RPSL objects
2500000
2460000
2440000
2420000
2400000
2380000
2360000
2340000
232,000
05/2006
2006/05/27
2006/06/10
2006/06/24
2006/07/08
2006/07/22
2006/08/05
2006/08/19
2006/09/02
2006/09/16
Snapshot date
09/2006
Snapshot date

Extracted Peerings



270,000

# of peerings

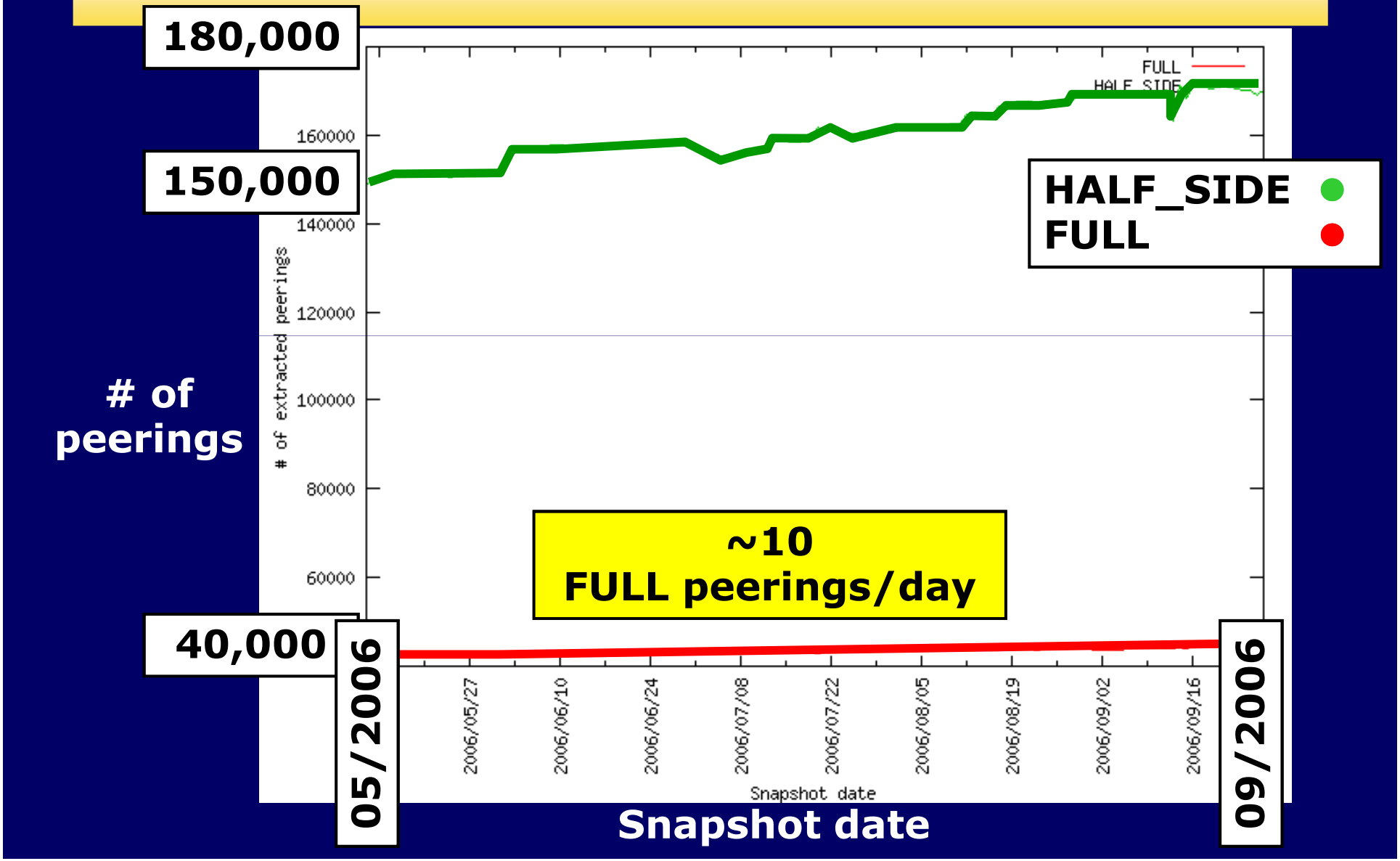
245,000

05/2006

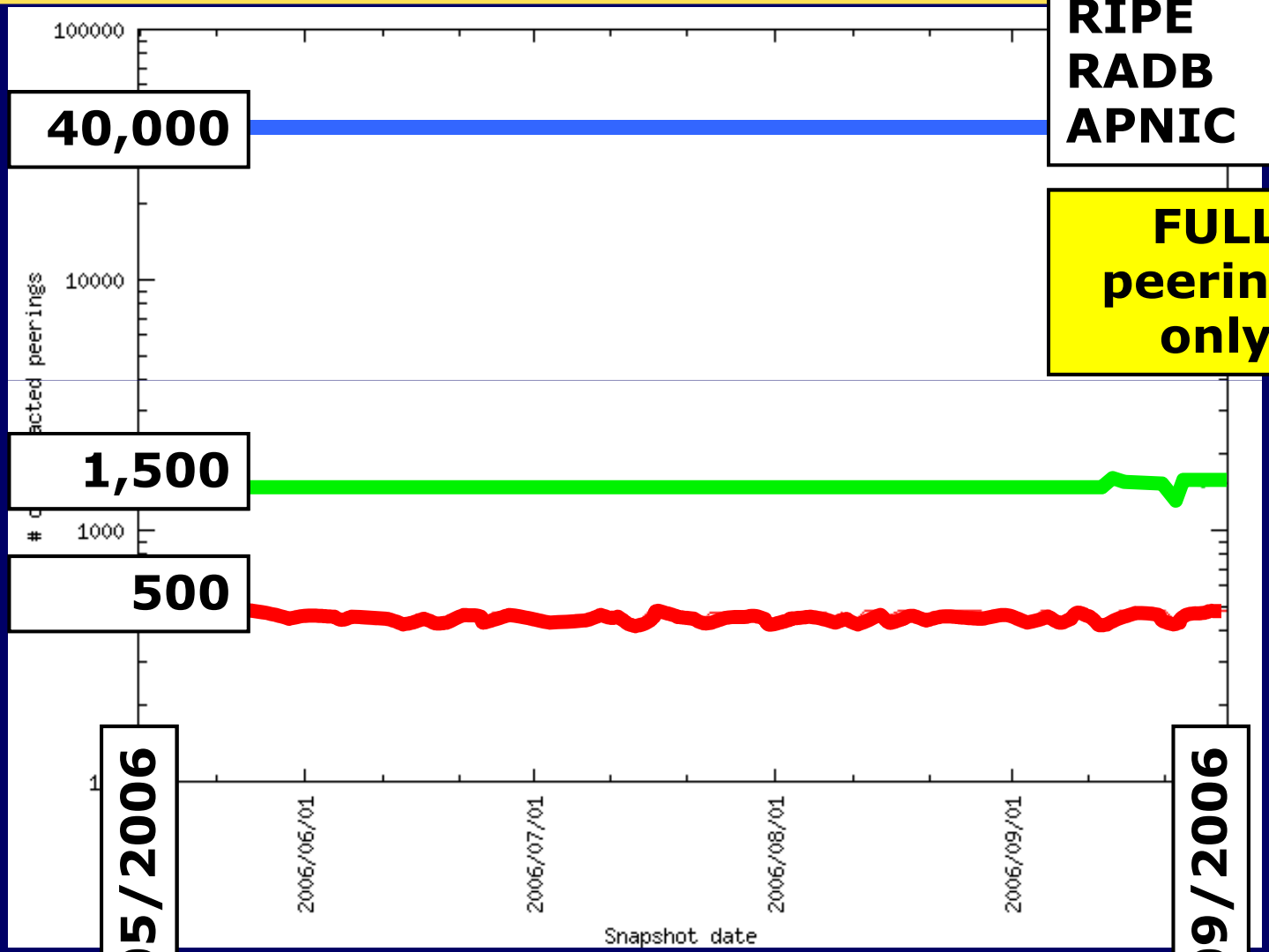
09/2006

Snapshot date

Extracted Peerings



Extracted Peerings



Wanna play?

Snapshot date: 29 Sep 2006

Last update within:
From: 11 Jul 1988 — Min: 11 Jul 1988
To: 29 Sep 2006 — Max: 29 Sep 2006

Registries:

ALTDB	AOLTW	APNIC	ARCSTAR	AREA151
ARIN	BCNET	BELL	BENDTEL	CSAS
DERU	DIGITALREALM	DODNIC	EASYNET	EBIT
EICAT	ENTERZONE	EPOCH	FASTVIBE	GT
GTS	GW	HOST	JPIRR	KOREN
LEVEL3	LOOK	MTO	NESTEGG	OPENFACE
OTTIX	PANIX	RADB	REACH	RETINA
RGNET	RIPE	RISQ	ROGERS	SAVVIS
SINET	SOUNDINTERNET	SPRINT	UNIVALI	VDN
VERIO				

All Registries (Push once to select all, 4 times to deselect all) | By Registry (Takes longer to respond)

Object types:

AS-BLOCK	AS-SET	AUT-NUM	DOMAIN	FILTER-SET
INET6NUM	INETNUM	INET-RTR	KEY-CERT	LIMERICK
MEMBERS	MNTNER	OUTE	PEERING-SET	PERSON
POEM	POETIC-FORM	RMUTE	ROLE	ROUTE
ROUTE6	ROUTE-SET	RTR-SET	*XXNER	*XXSET

All object types (Push once to select all, 4 times to deselect all) | By object type (Takes longer to respond)

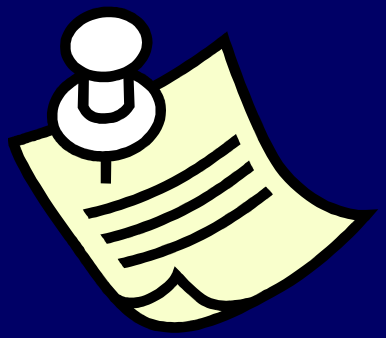
Cumulative plot: Cumulative distribution (CDF)

Plot size: 640x480

Plot: Plot

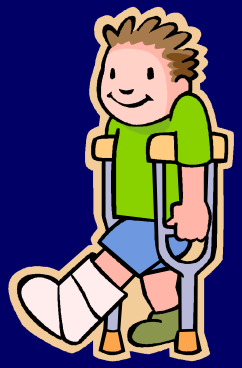
Reset query parameters: Reset

So What?



- Extracting peerings from the IRR is not trivial...

  ...yet it's possible
- A systematic approach
- An on-line service providing
  - data
  - plots
- Hints about the health of the IRR



IRR vs RIS+ORV

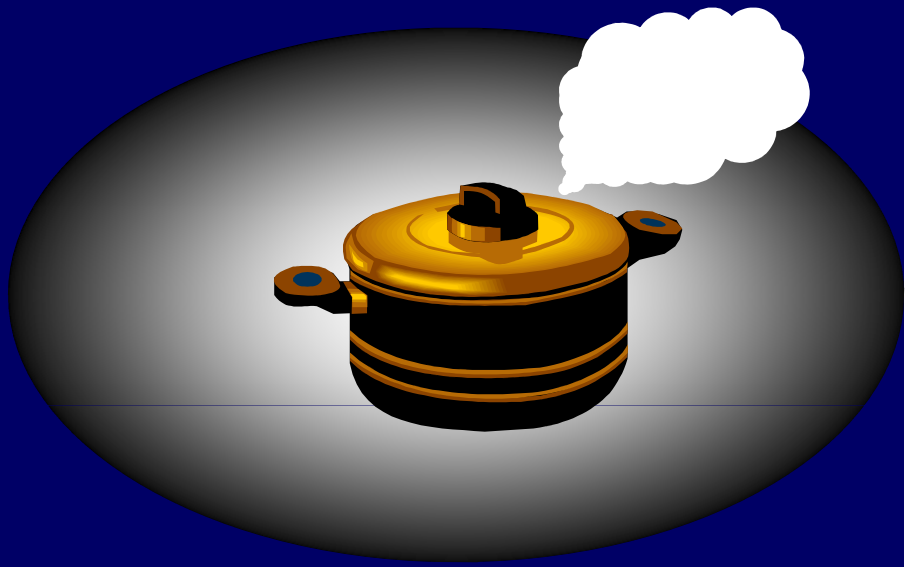
- BGP RIBs downloaded from RIS and ORV
- Reference date: 10/07/06

	IRR	RIS+ORV
# of peerings	254,660	56,916
Only in	222,506	24,762
Common	32,154	

Who is responsible for this?

What's in the pot?

- How many peerings not observable from routing data...
  - ...are up-to-date?
  - ...involve transit ASes?
  - ...are between two tier-1?
- BGP routing policies



- Estimate consistency of IRR data against actual routing
- Prevent abnormal routing scenarios

Care to Have a Look?

http://tocai.dia.uniroma3.it/~irr_analysis/

Thank you!
Questions?

G. Di Battista, T. Refice M. Rimondini,
How to Extract BGP Peering Information from the IRR,
SIGCOMM, 2006.