Tutorial:
Using RIPEstat

MENOG 13

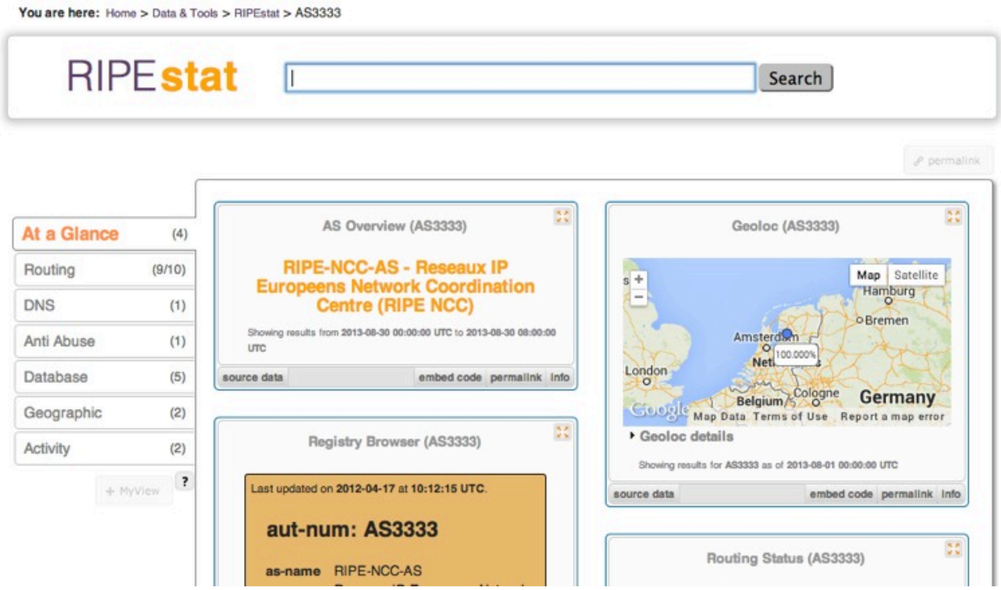
Christian Teuschel

RIPE NCC

Recap From the RIPEstat Presentation

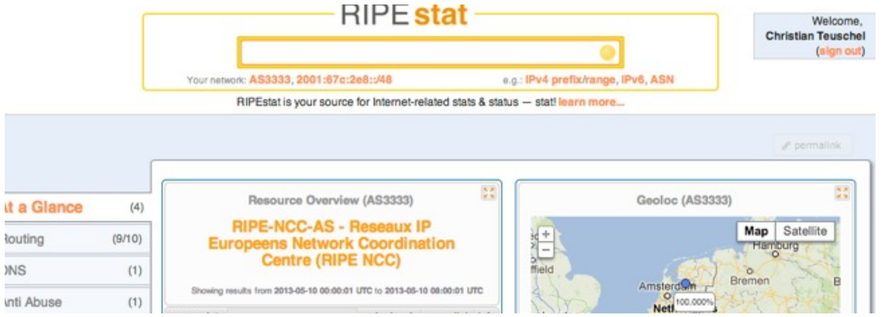
- Information system for Internet number resources

- Data
  - Routing data
    - Collected by RIS: http://ris.ripe.net
  - Registration data (whois)
    - RIPE Database & other RIR databases
  - MaxMind's geolocation data
  - Blacklist data
  - And many more: https://stat.ripe.net/data-sources

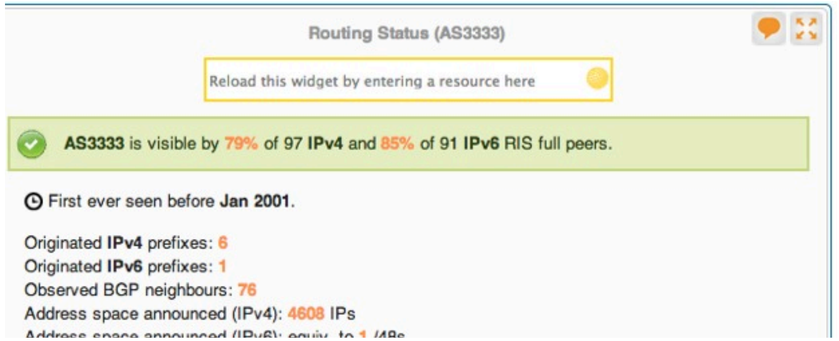


Three Interfaces of RIPEstat

RIPEstat Web Interface



RIPEstat Widget API

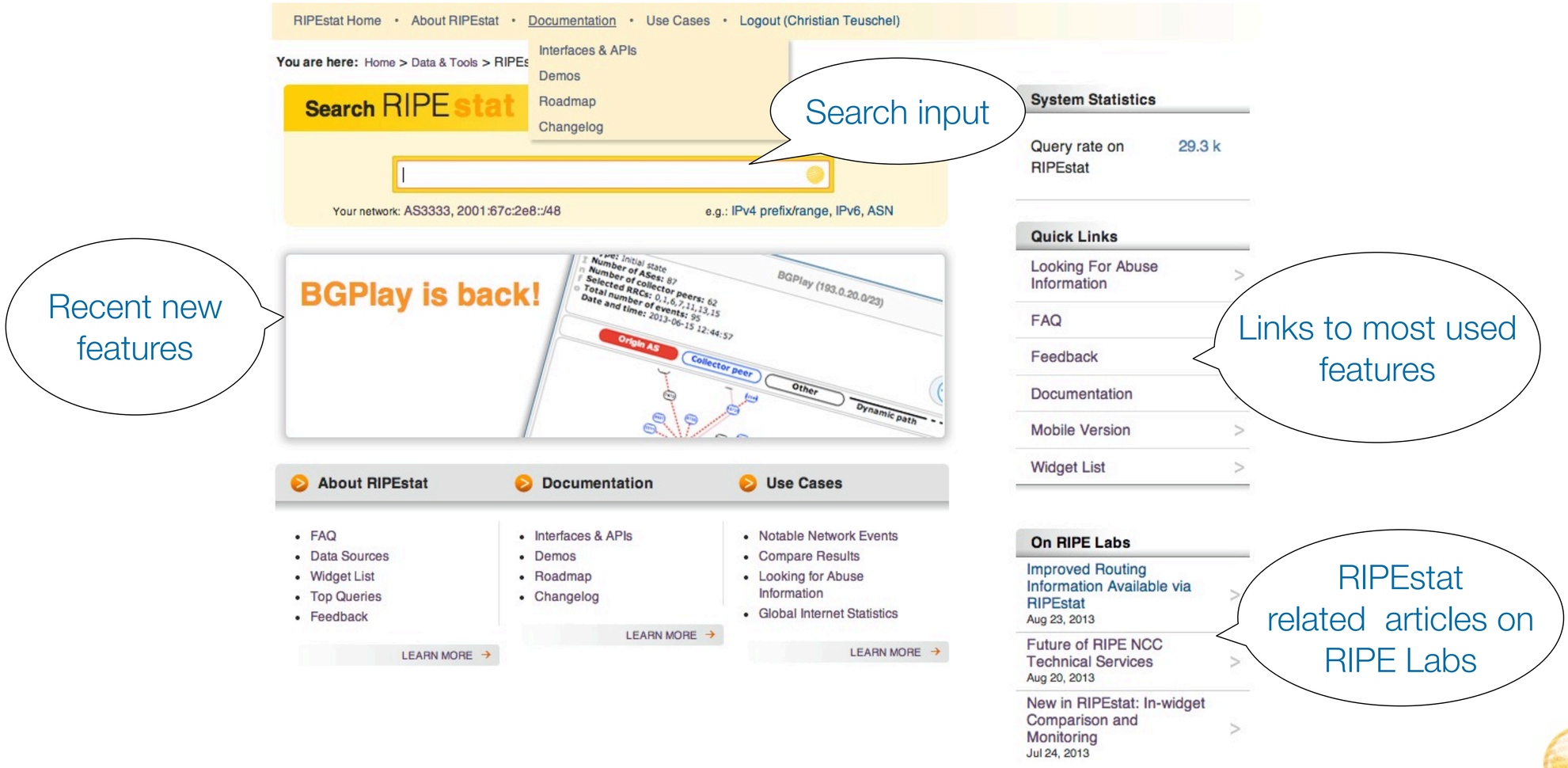


RIPEstat Data API / RIPEstat Text

https://stat.ripe.net/data/routing-status/data.json?resource=AS3333

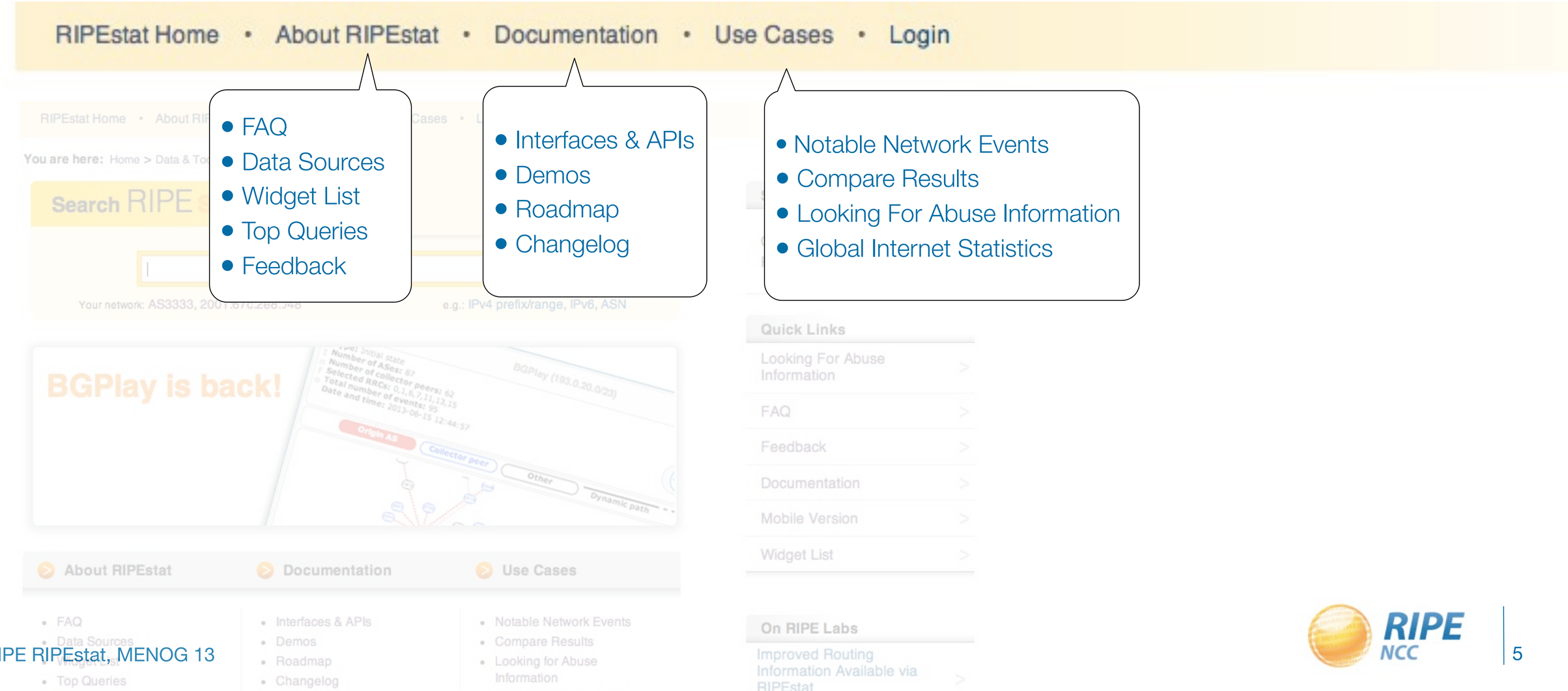
RIPEstat Web Interface

- Finding Information on RIPEstat



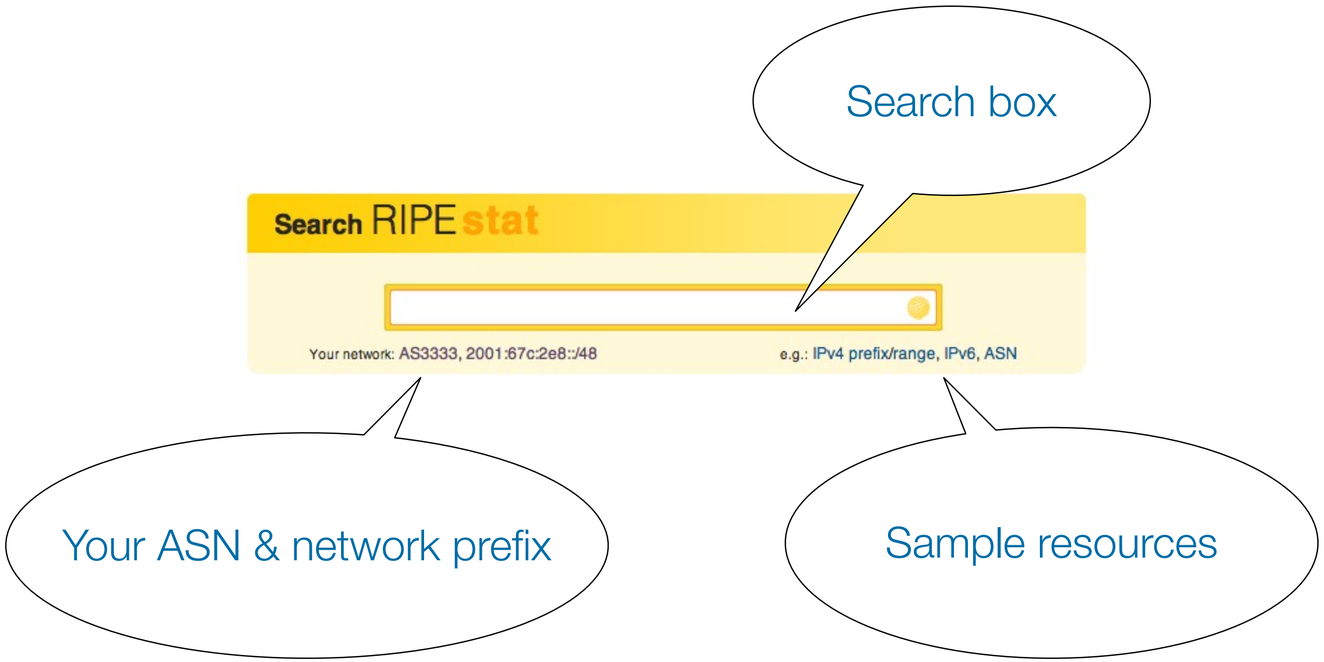
RIPEstat Web Interface

- Information structure on RIPEstat



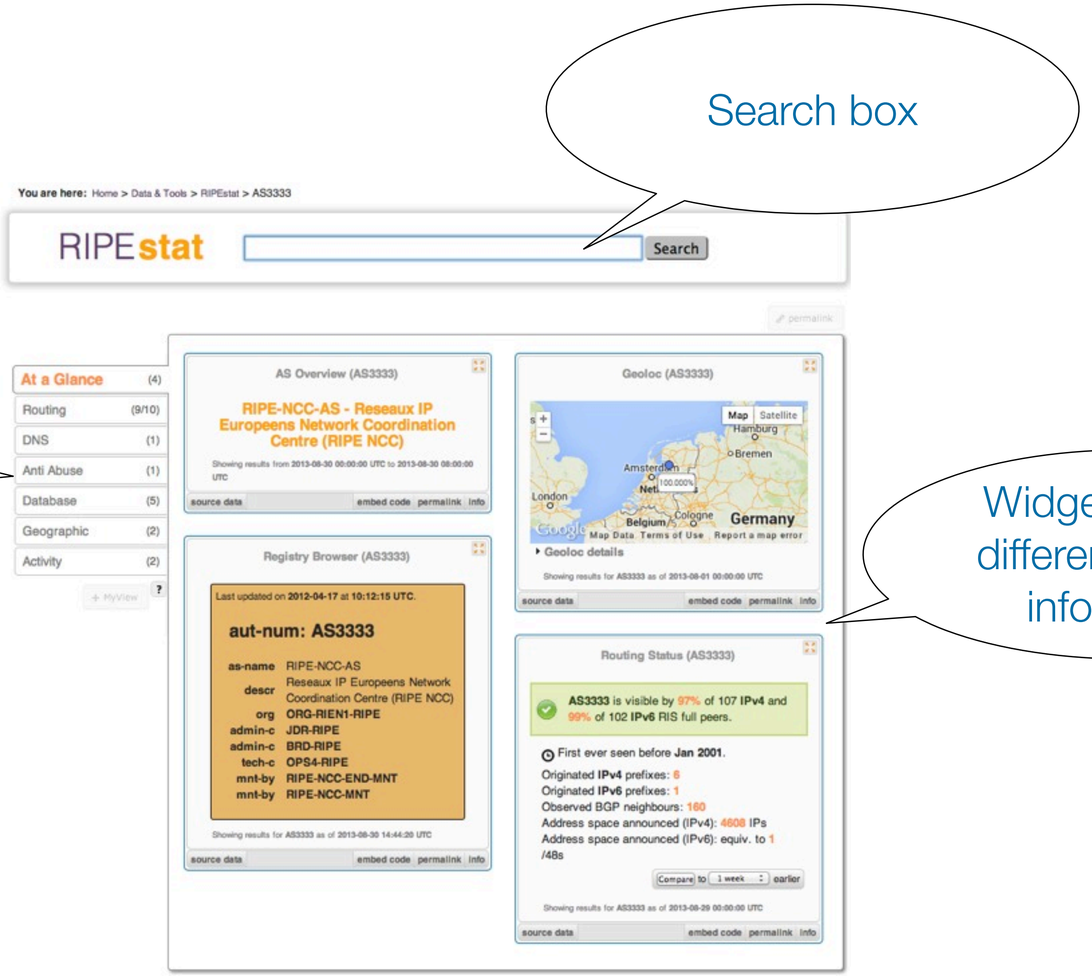
1. Querying for a Resource (Web Interface)

Example: Enter “AS3333” in the search box



1. Querying for a Resource (Web Interface)

Result page



1. Querying for a Resource (Web Interface)

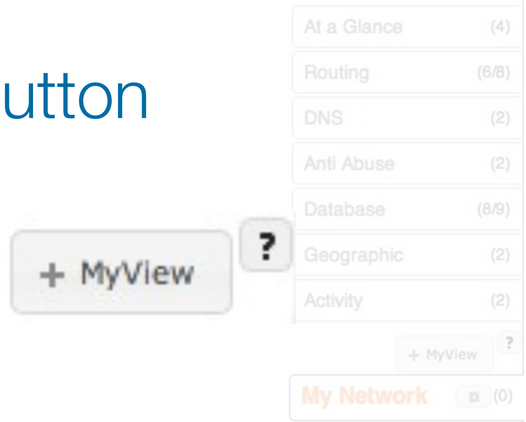
Tasks:

- What network announces 140.78.50.90?
- Is 192.3.4.2 routed?
- In which country is 91.229.42.0/23 used?
- What is its corresponding INETNUM object?
- What widget provides real-time routing status?
- By what percent did the number of prefixes announced by Kuwait increase over the last two years?
- How would you share interesting network events with a colleague?

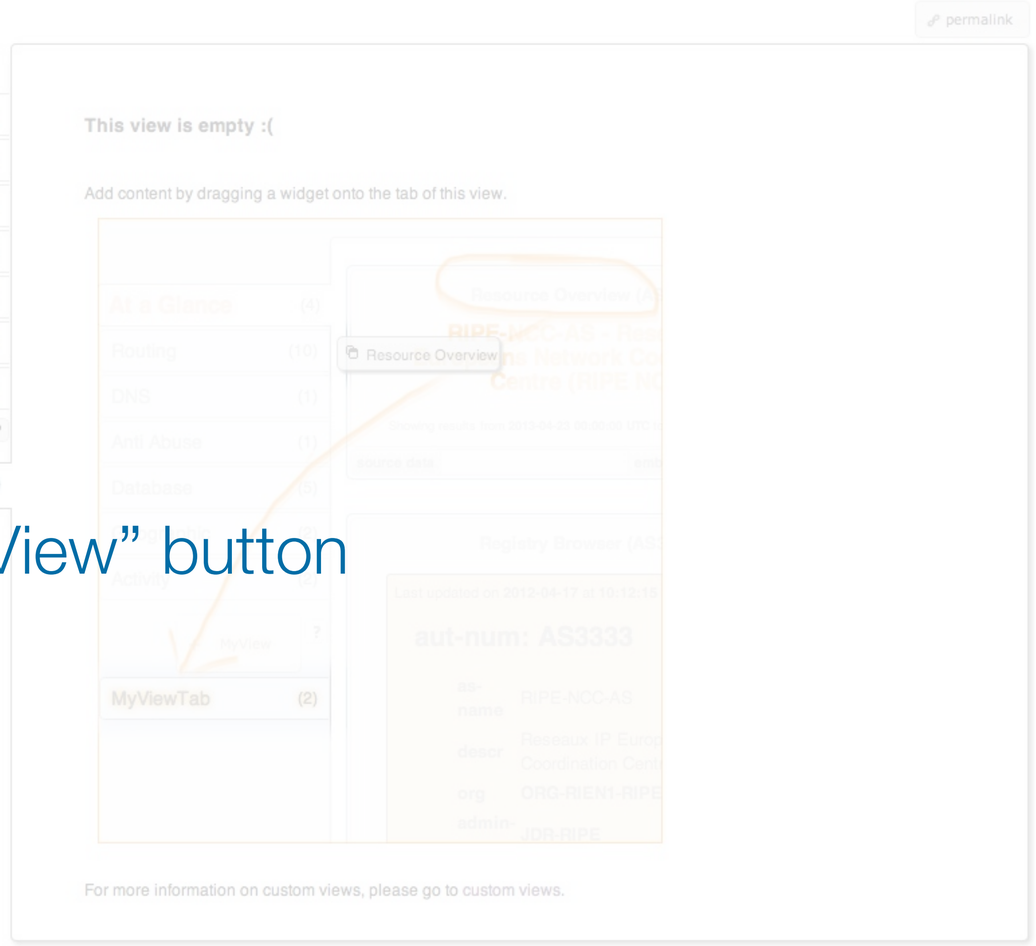
2. MyViews

Create custom views:

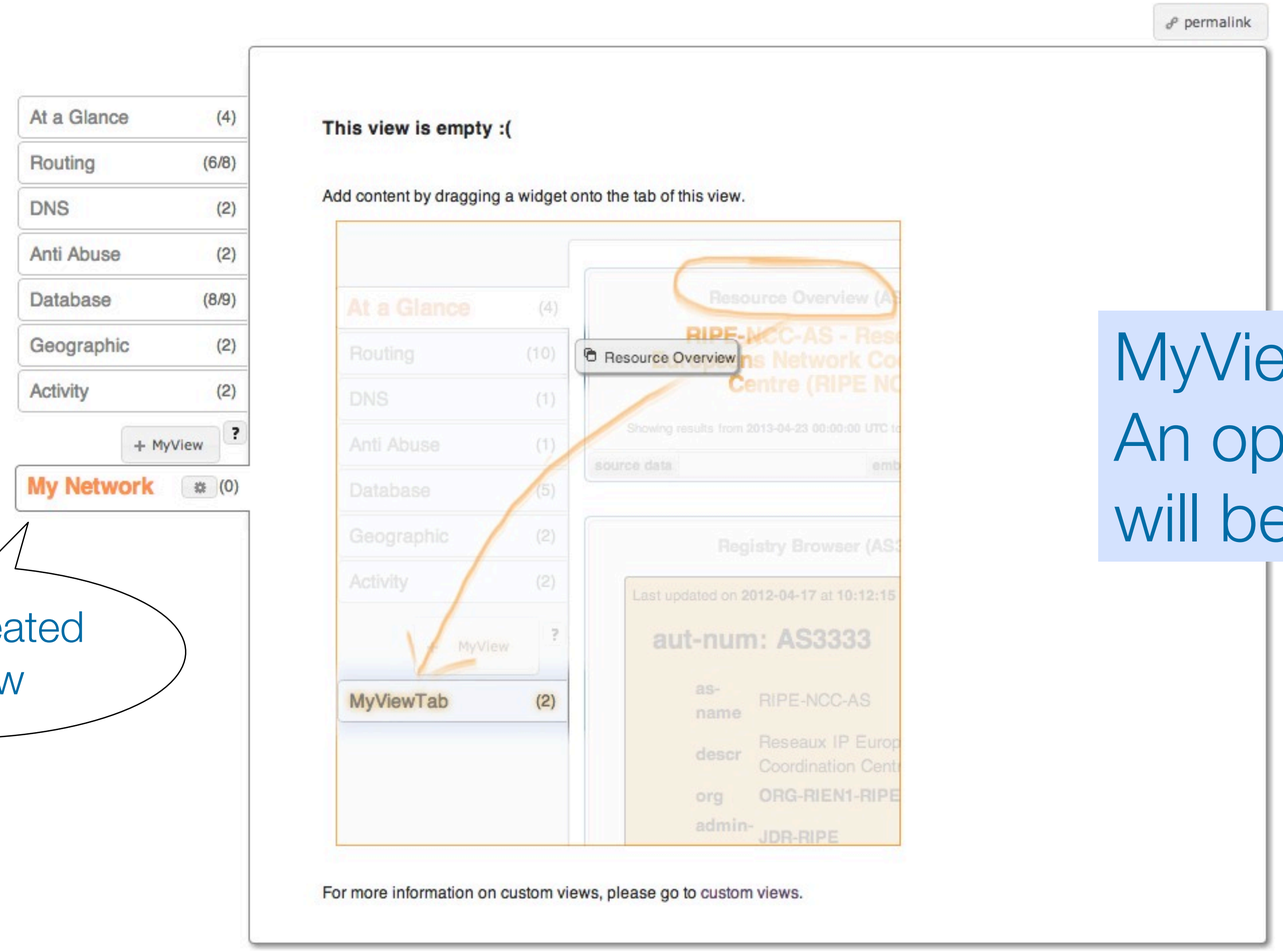
- Click the“MyView” Button



- Drag and drop a widget onto the “MyView” button



2. MyViews



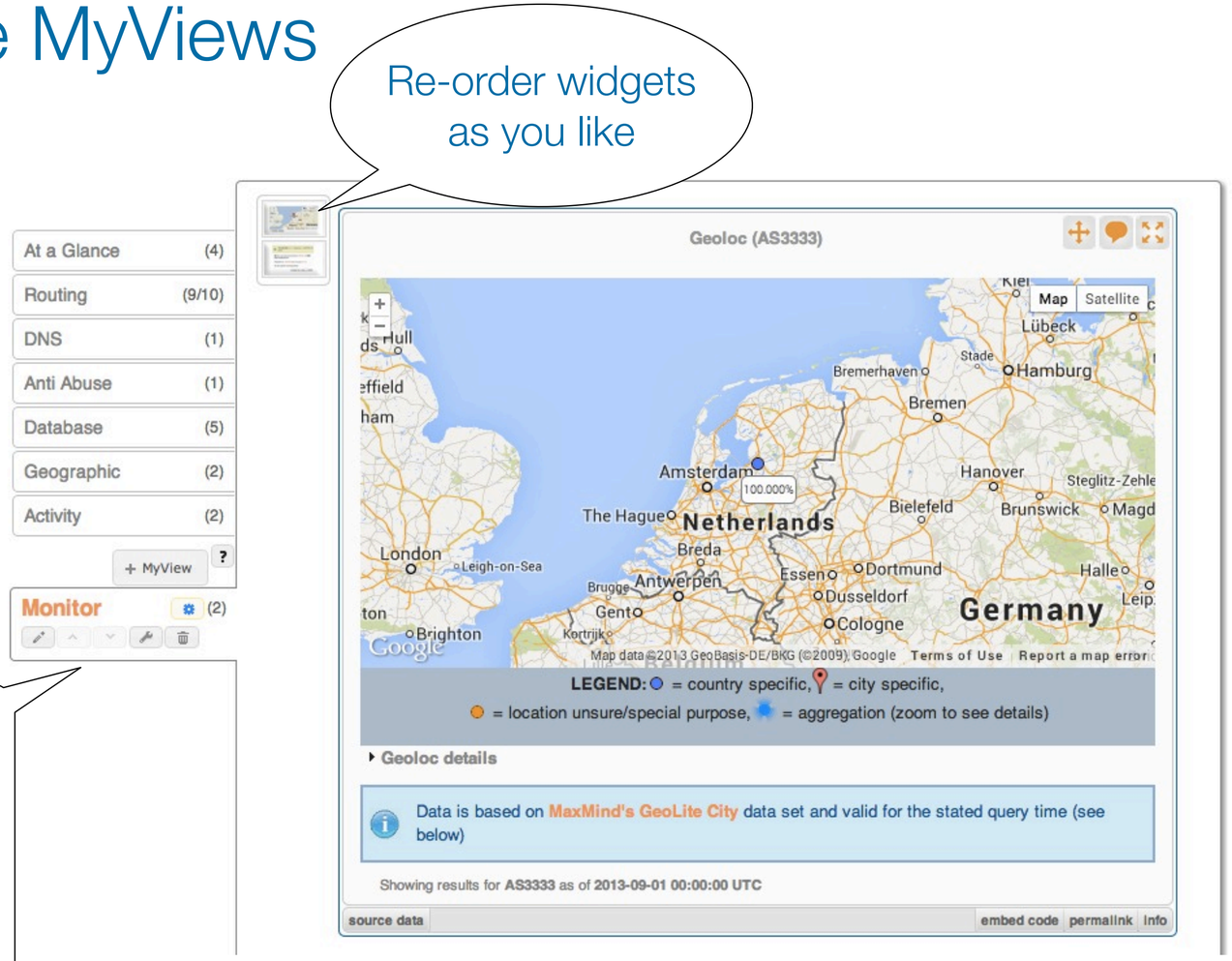
Newly created MyView

MyViews are only visible to you. An option to share your views will be available soon!

2. MyViews

Customise MyViews

Re-order widgets as you like



- Rename
- Re-order
- Control visibility
- Remove

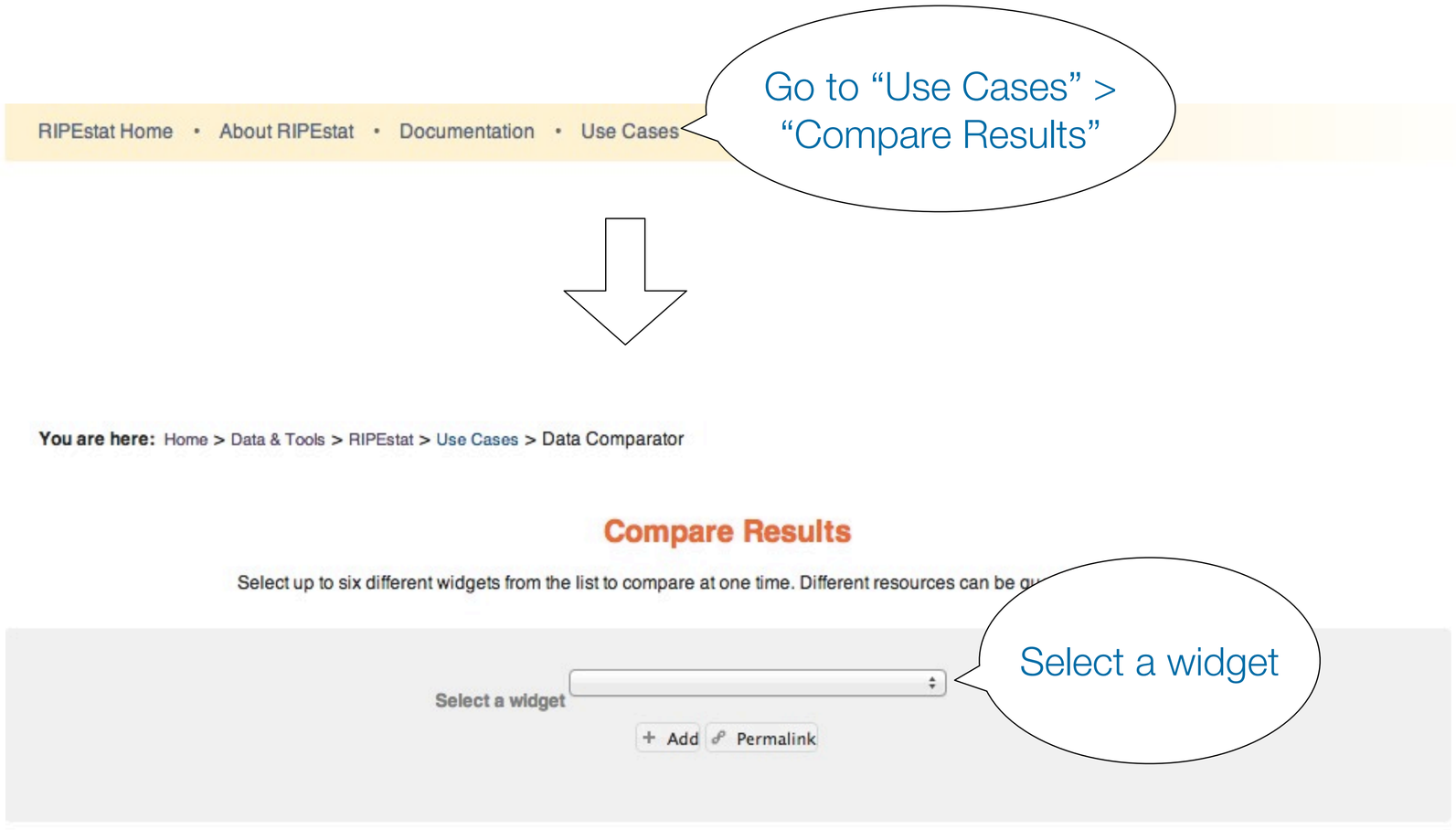
2. MyViews

Tasks:

- Create a RIPE Access account (if you don’t already have one!)
- Create a MyView for a prefix containing the following widgets:
  - Routing Status
  - Looking Glass
  - Routing History
- Create another MyView with at least two widgets, and give it a meaningful name

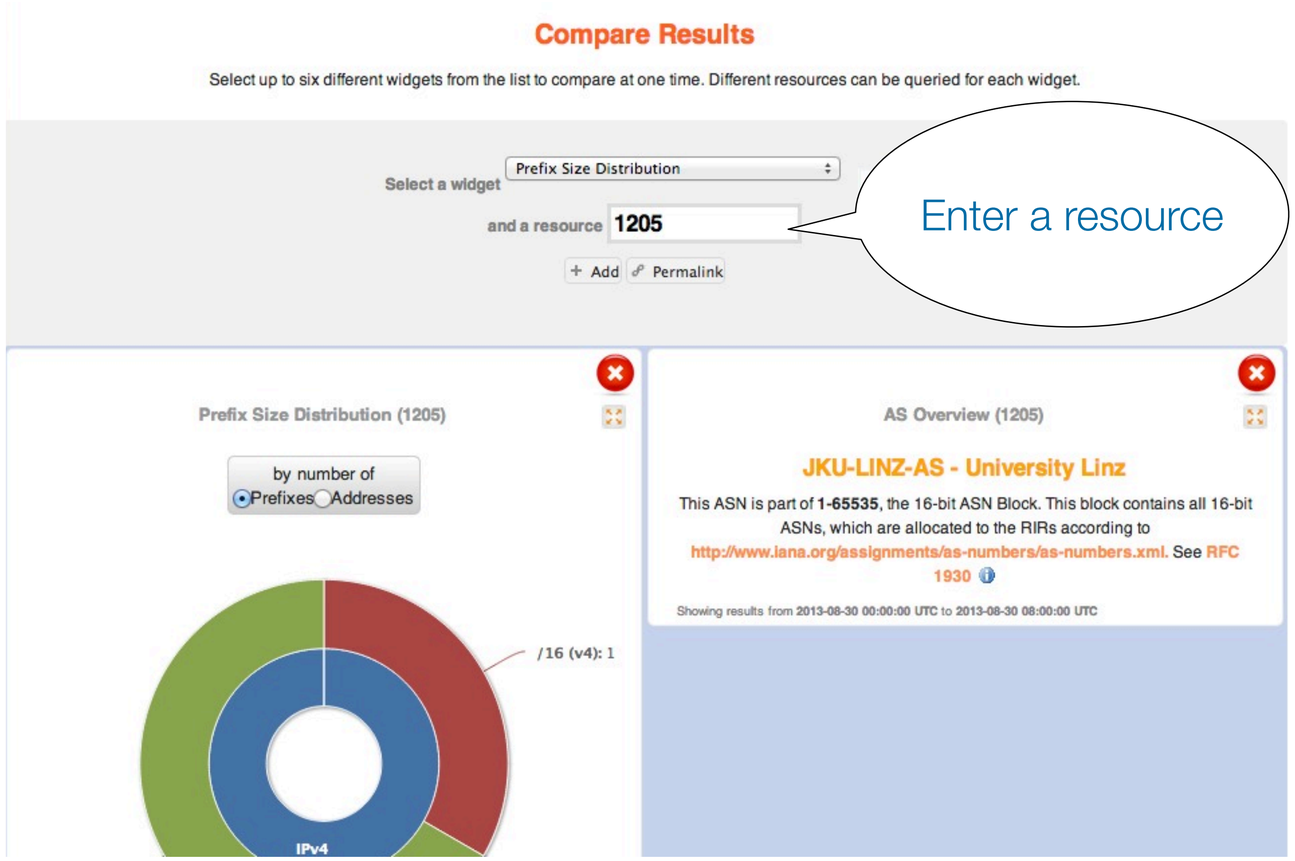
3. Comparing Resources (Web Interface)

- Compare results in different widgets



3. Comparing Resources (Web Interface)

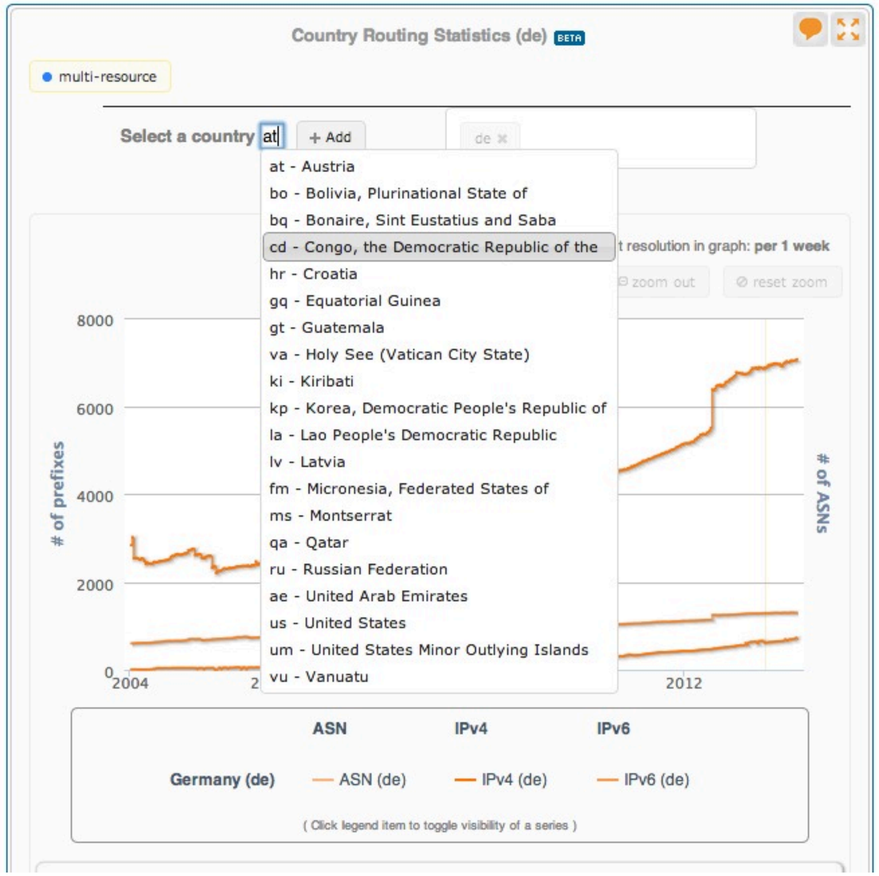
- Compare results in different widgets



- Select the “Prefix Size Distribution” widget
- Enter “AS1205”

3. Comparing Resources (Web Interface)

In-widget comparison



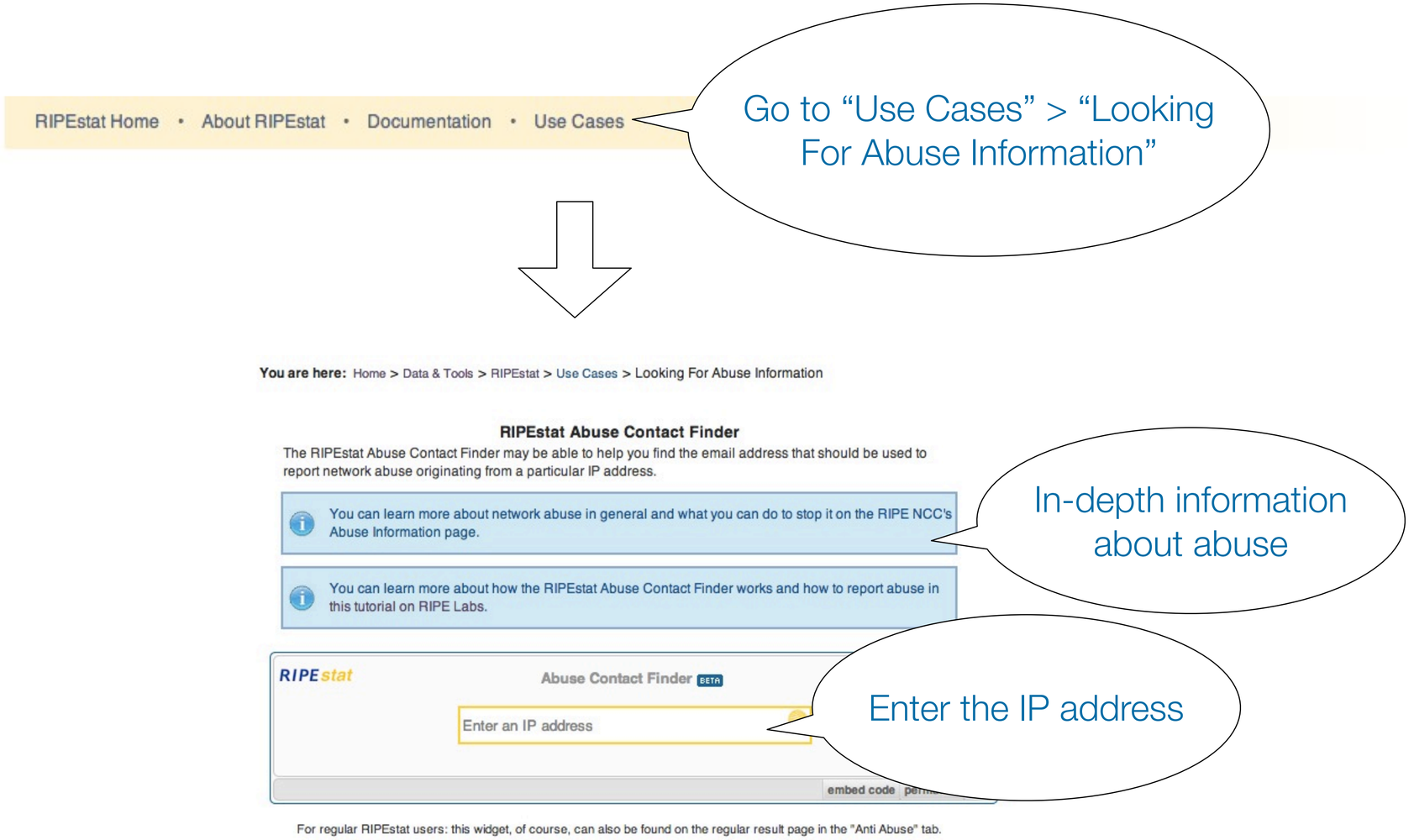
3. Comparing Resources (Web Interface)

Tasks:

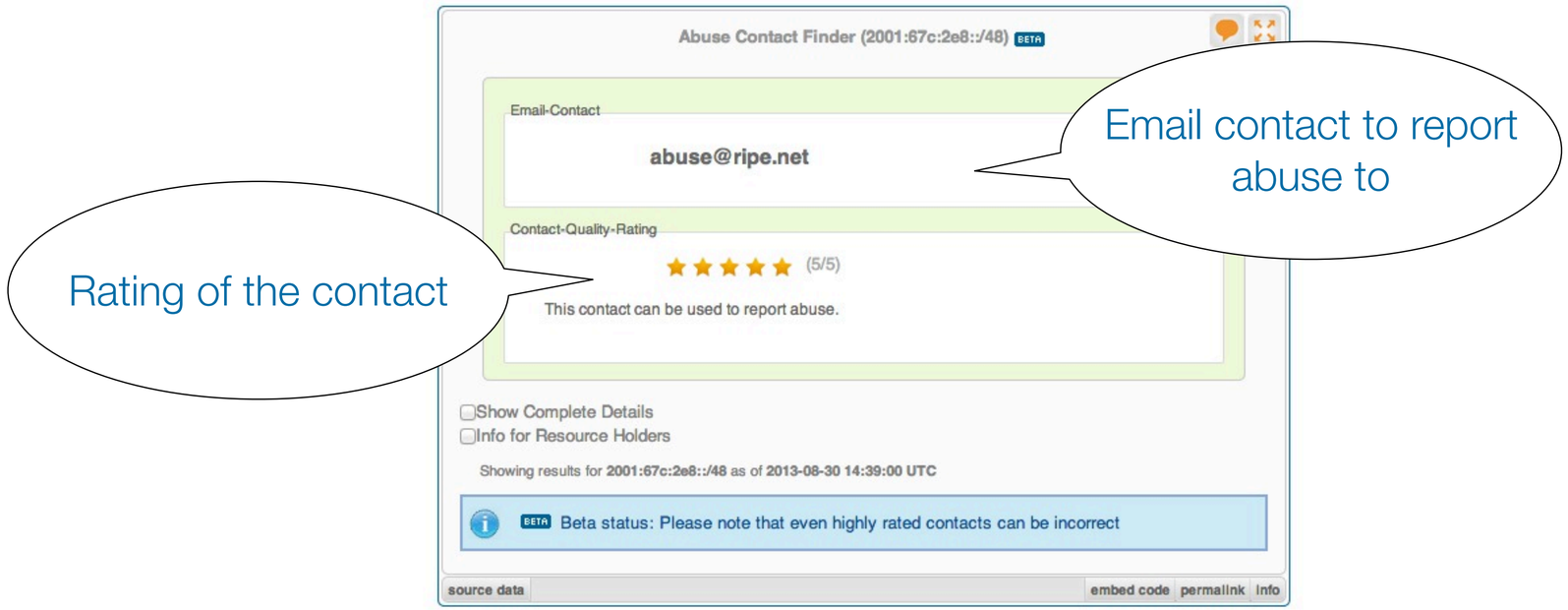
- Compare the number of announced prefixes for two networks over the past two years using the widget comparison page
- How does the Internet in Kuwait compare to the UK? Use in-widget comparison!

4. Finding Abuse Contacts & Reporting Abuse

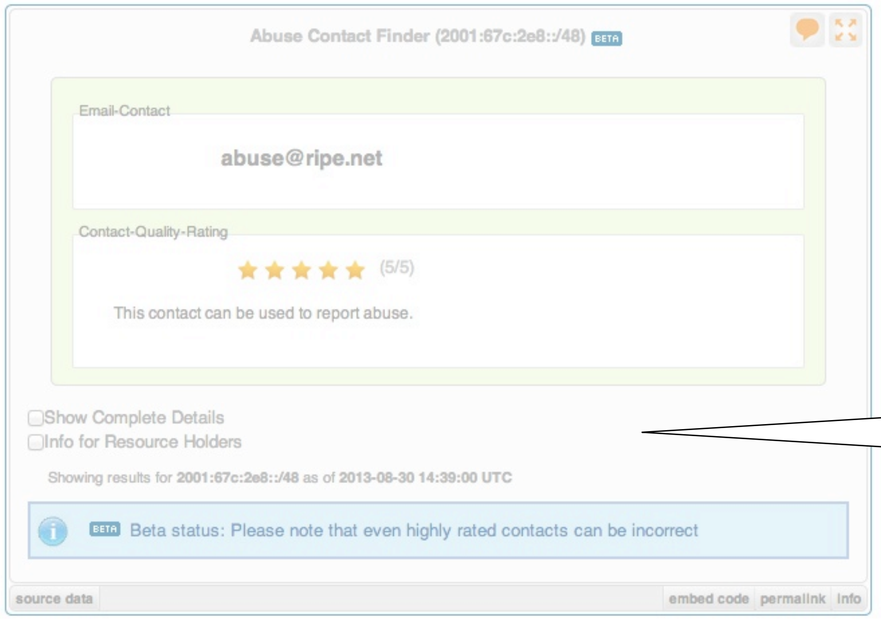
Take action in an abuse case with the Abuse Contact Finder



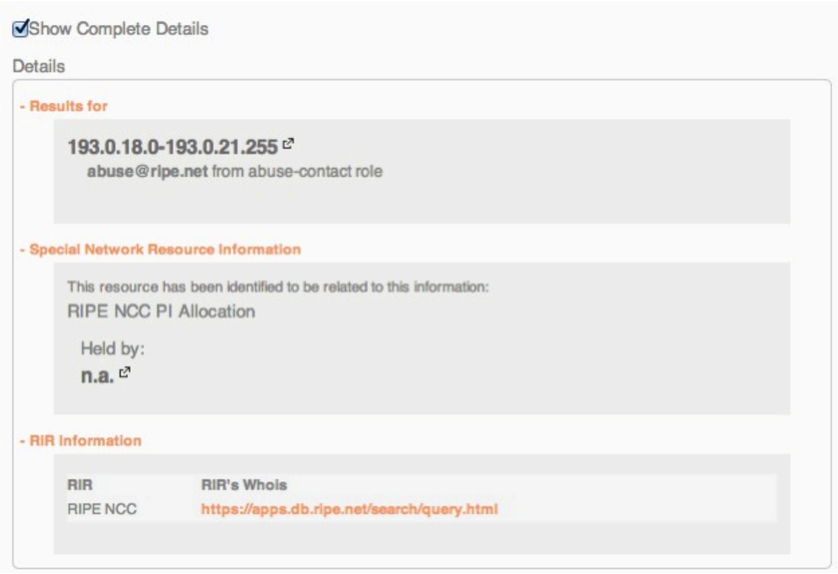
4. Finding Abuse Contacts & Reporting Abuse



4. Finding Abuse Contacts & Reporting Abuse



Details about the resource and abuse contact:



4. Finding Abuse Contacts & Reporting Abuse

Tasks:

- What is the abuse contact for 193.0.20.22 or the hotel network?
- Check an IP address from your home network
- Discussion:

  What can you do in these cases?

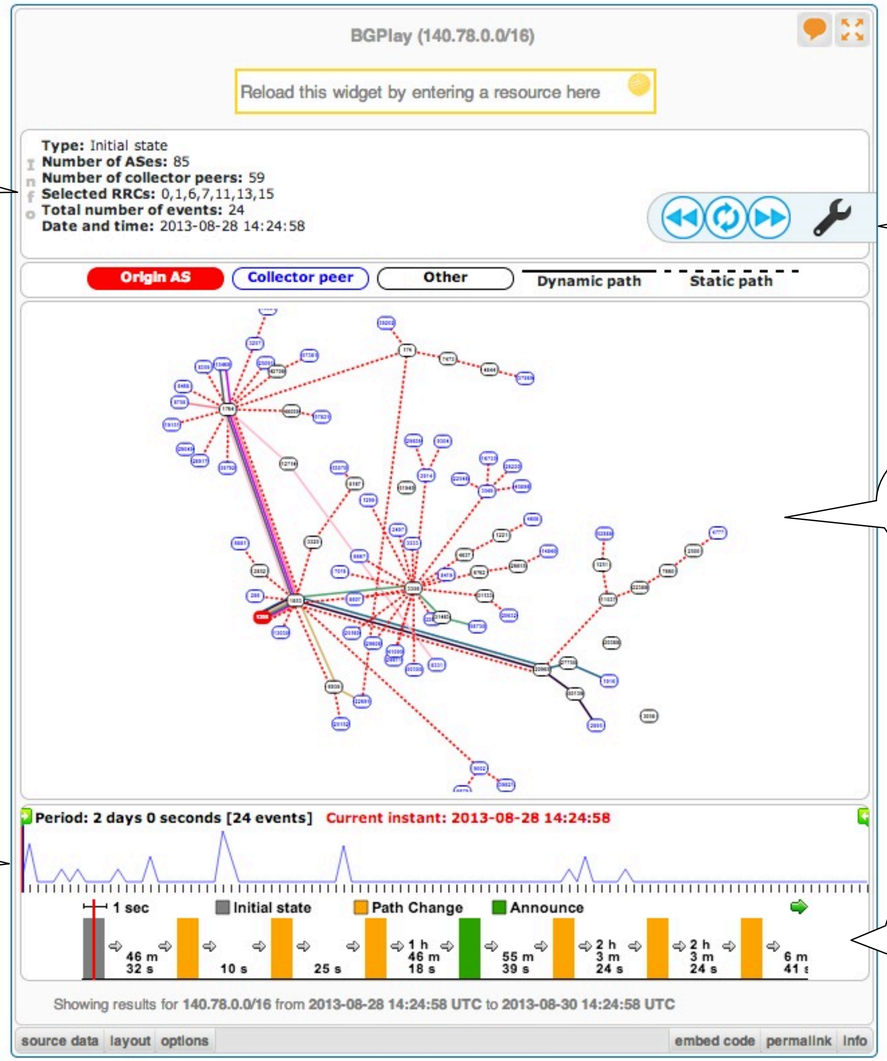
  - No abuse contact found
  - No response on an abuse report

5. Let's BGPlay!

- Use BGPlay to see how your network is routed
  - BGPlay is a tool that show routing history in an animated and highly-interactive manner
- Go to: https://stat.ripe.net/widget/bgplay

5. Let's BGPlay!

BGP event, ASN or ASN path details



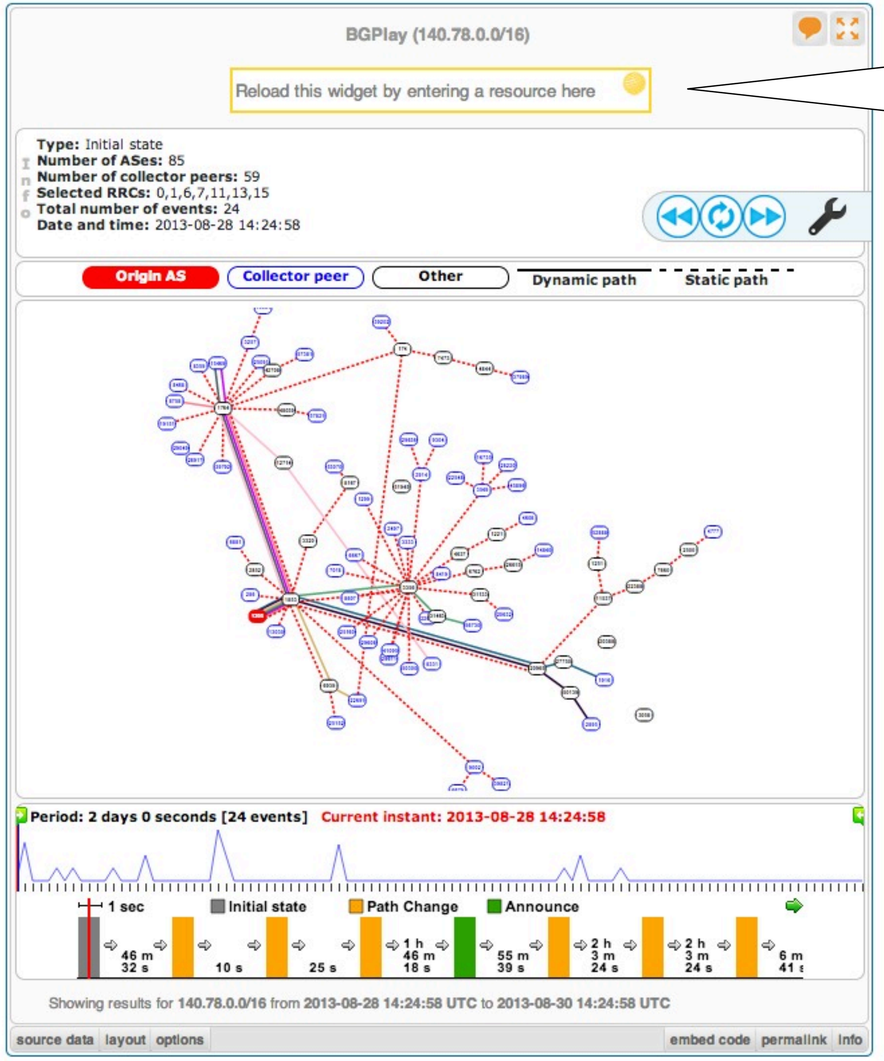
Control panel:
- Covered time period
- RRC selection

Interactive graph visualisation

Control timeline

Selection timeline

5. Let's BGPlay!



Examples:

- Prefix with announcements & withdrawals: 84.205.64.0/24
- Check IPv6 connectivity: 2001:67c:2e8::/48
- Multi-homed prefix: 199.7.80.0/24

5. Let’s BGPlay!

Tasks:

- Find the up-stream provider for AS1205
- Is AS3333 multi-homed?
- Check the IPv6 connectivity of your own network

6. Embedding Widgets On Your Own Site

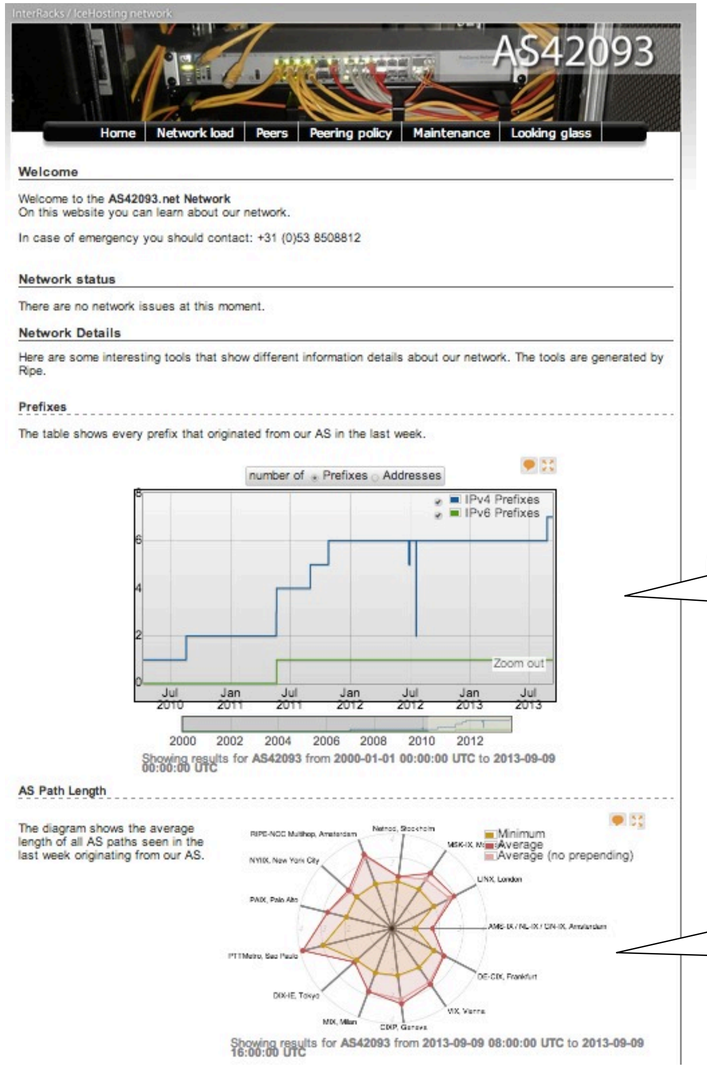
Embedding widgets

- What is the concept?

  Widgets are based purely on web standards and were developed with embeddability in mind. Every widget available via the RIPEstat web interface can be embedded on your own site.

- RIPEstat Widget API documentation:
  https://stat.ripe.net/docs/widget_api

6. Embedding Widgets On Your Own Site



This ISP embedded widgets on its page.

Prefix Count widget

AS Path Length widget

6. Embedding Widgets On Your Own Site *(Task)*

Tasks:

- Create a simple dashboard page to monitor your network
  - Create a simple HTML page
    - download the sample page:
      https://stat.ripe.net/widgets/menog/widget_home.html
  - Embed three widgets of your choice
  - Make the widget smaller and remove the RIPEstat logo

7. Create a Nagios check (optional)

- Create a Nagios check using the RIPEstat Data API
  - Nagios is a monitoring tool that supports the development of custom checks (like other similar tools)
- Result codes for a check show the state:

State	Result Code
Ok	0
Warning	1
Error	2

7. Create a Nagios check (optional)

Example of a Python based check:

```
if args.transit:
    transit = args.transit.split(',')
origin = args.origin.split(',')
nagios_status = 0
origin_errors = 0
transit_errors = 0
nagios_message = ""
url = "https://%s/data/looking-glass/data.json?resource=%s" % (args.statserver, urllib2.quote(args.prefix, ''))
lookingglass_raw = urllib2.urlopen(url)
lookingglass_json = json.load(lookingglass_raw)
if lookingglass_json['data_call_status'] != "supported":
    print "WARN: %s is under maintance " % (args.statserver)
    sys.exit(1)
for rrc in lookingglass_json['data']['rrcs']:
    for peer in lookingglass_json['data']['rrcs'][rrc]['entries']:
        aspath = peer['as_path'].split()
        details = peer['details'][0].split()
        peer_addr = details[0]
        nexthop = details[2]
        router_id = details[3]

        if len(aspath) > args.minpath:
            if aspath[-1]  not in origin :
                origin_errors += 1
                nagios_message = nagios_message + ( "Origin missmatch %s (%s): %s; " % (rrc, peer_addr, aspath[-1]))
            if args.transit:
                if aspath[-2]  not in transit :
                    transit_errors += 1
                    nagios_message = nagios_message + ( "Transit missmatch %s (%s): %s; " % (rrc, peer_addr, aspath[-2]))

if origin_errors == 0 and transit_errors == 0:
    nagios_message = "OK: %s Origin is %s " % (args.prefix, args.origin)
    if args.transit:
        nagios_message = "%s and all transits match %s" % (nagios_message, args.transit)
elif origin_errors >= args.crit or transit_errors >= args.crit:
    nagios_status = 2
    nagios_message = "ERROR: " + nagios_message
elif origin_errors >= args.warn or transit_errors >= args.warn:
    nagios_status = 1
    nagios_message = "WARN: " + nagios_message
```

RIPE NCC