



# YIN-YANG

ninjaX tracerouting



Edward Zambrano, Spotify  
Orlin Tenchev, Sofia Connect  
Sascha Bleidner, DE-CIX  
Rickard Östman, Spotify

# The YIN-YANG Team



And the prize?



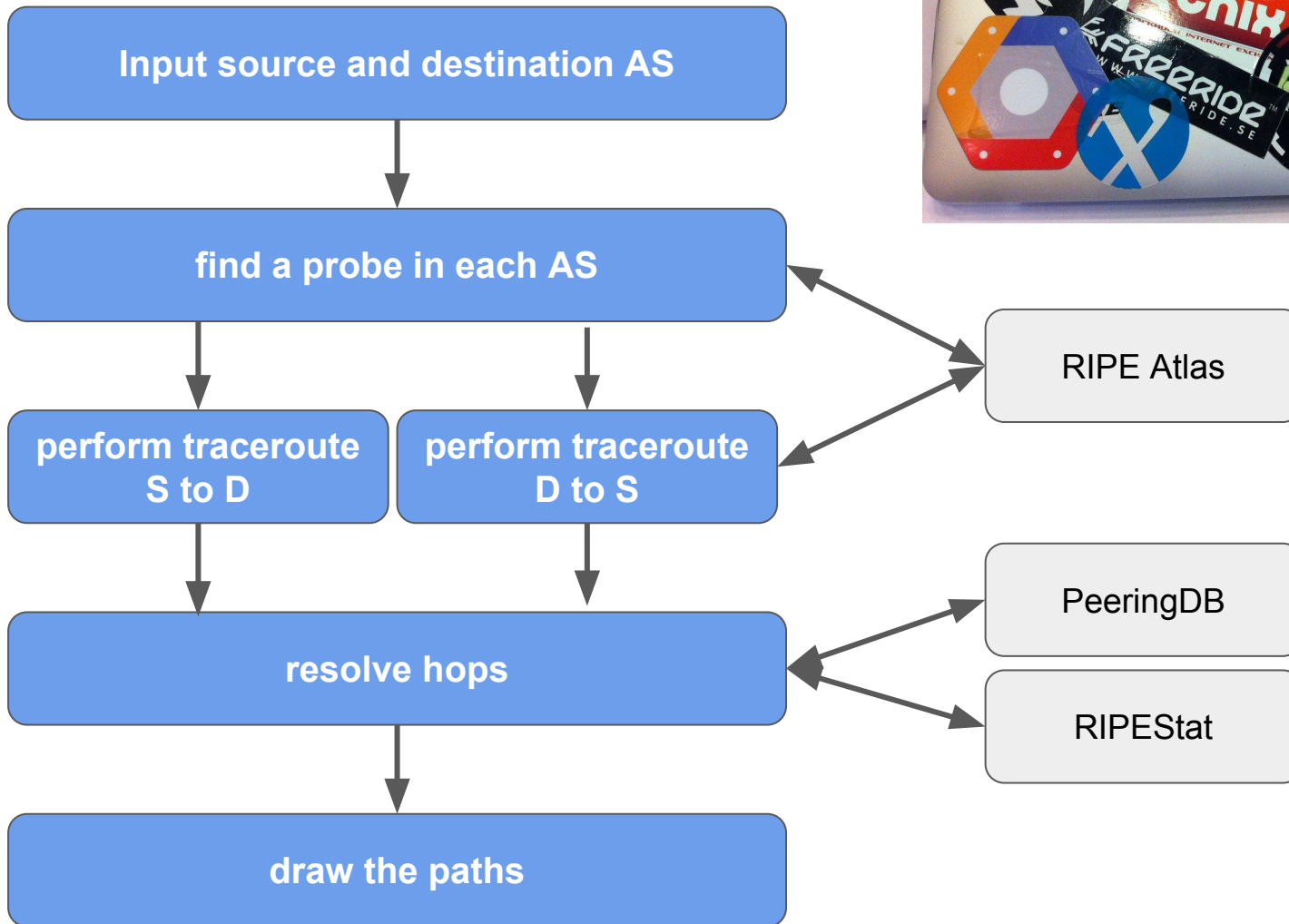
# Motivation



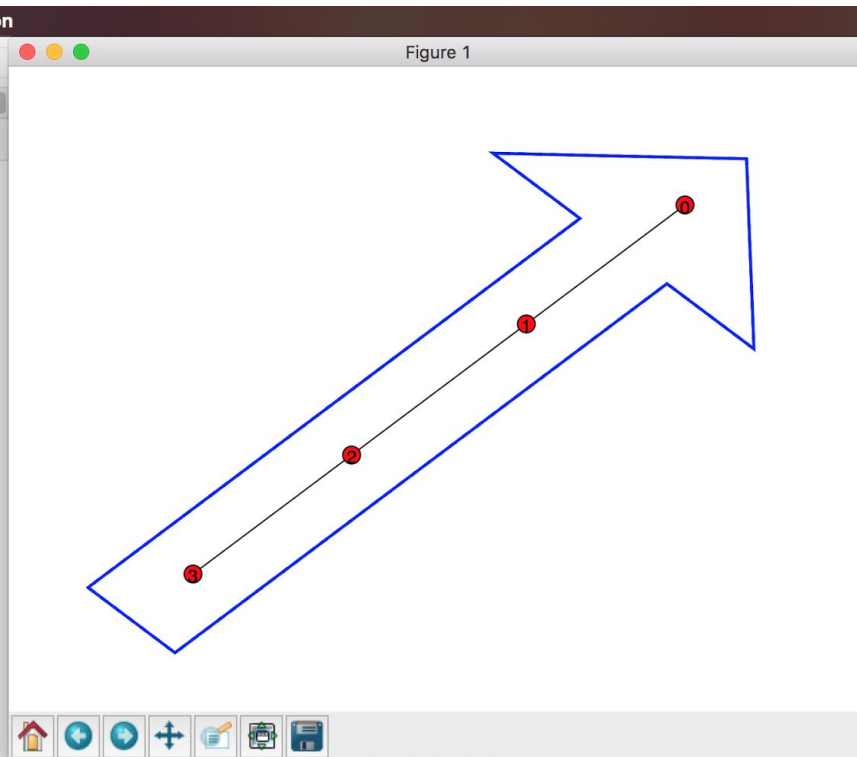
How to troubleshoot your two way connection?

You need control over the destination!

# The Hack



# First Steps



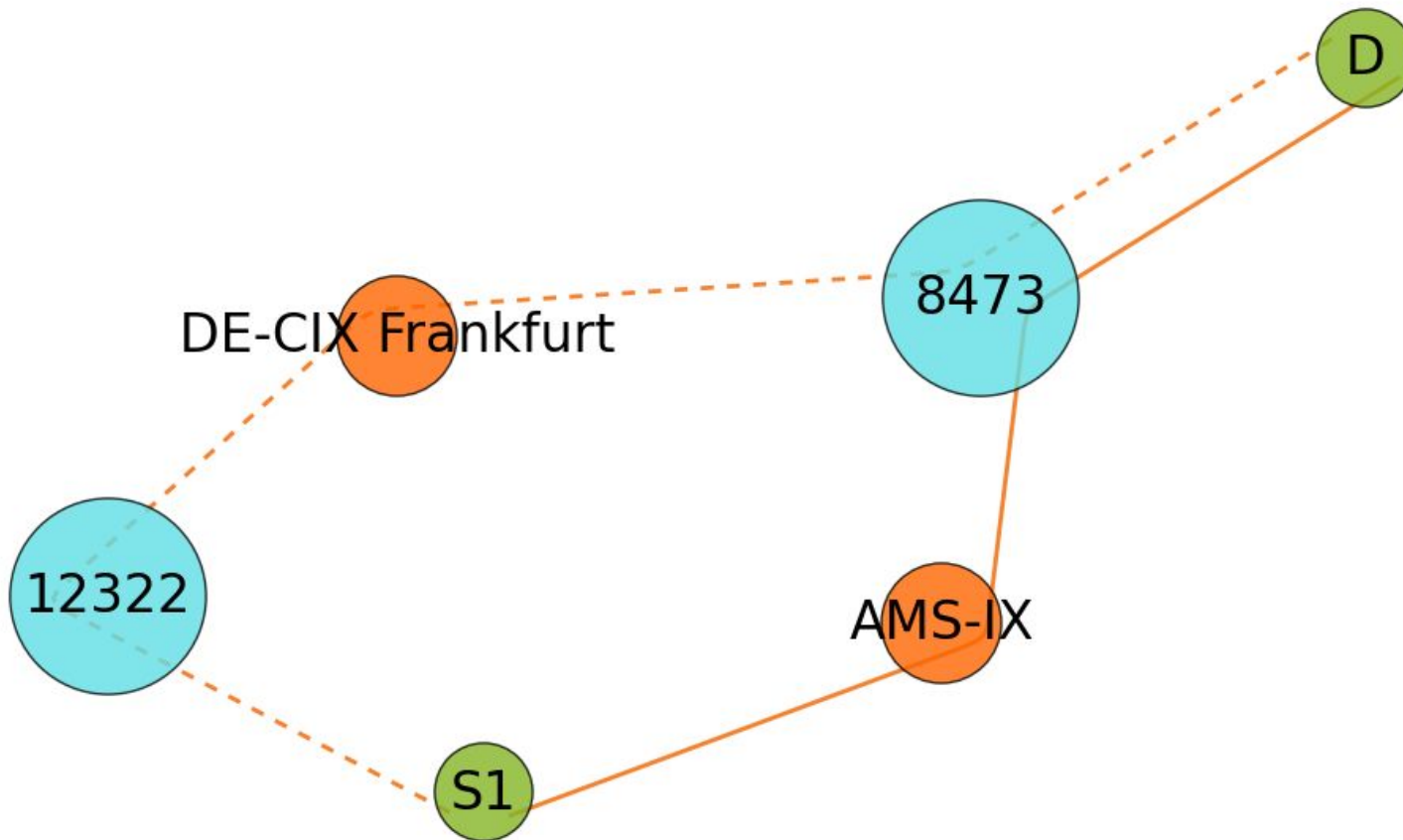
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[(Path, Transform)   Patch   None]
color	matplotlib color spec
contains	a callable function
edgecolor or ec	mpl color spec, or None for default, or 'none' for no color
facecolor or fc	mpl color spec, or None for default, or 'none' for no color
figure	a matplotlib.figure.Figure instance
fill	[True   False]

```
PathDrawer.2.py — yinyang
test.py x PathDrawe x vega.json x openipma x geolocatio x lxp_lan.py x
250
251 # First operation must be a MOVETO, move pen to first
252 vertices += [corners[0][0]]
253 codes += [MPath.MOVETO]
254
255 for i,corner in enumerate(corners):
256
257     # If there is not enough space to draw a corner,
258     if i>0:
259         if vector_length(norm_vector(corner[0]-verti
260             vertices.pop();
261             vertices.pop();
262             vertices += corner[-2:]
263             continue
264
265     codes+= [MPath.LINETO, MPath.CURVE4, MPath.CURVE4]
266     vertices+=corner
267
268 # Finally, create a nice path and display it
269 path = MPath(vertices, codes)
270 patch = matplotlib.patches.PathPatch(path, edgecolor
271 patch = matplotlib.patches.Arrow(0,0,1,1, edgecolor
272 ax=matplotlib.pyplot.gca()
273 ax.add_patch(patch)
274 ax.update_datalim(((0,0),(1,1)))
275 ax.autoscale_view()
276
277 return
278
279
280 #####
281 def draw_many_paths(G, pos, paths, max_shift=0.02, linew
282     '''Draw every path in 'paths' in graph G.
283     Colors and linestyle are chosen automatically.
284     All paths are visible — no path section can be cover
```

/Users/sbleidner/Downloads/PathDrawer.2.py:227: FutureWarning: comparison to 'None' for non-equivalency is not supported. The edges are parallel

# PROXAD (FR) -> Bahnhof (SE)

RTT 69.586ms

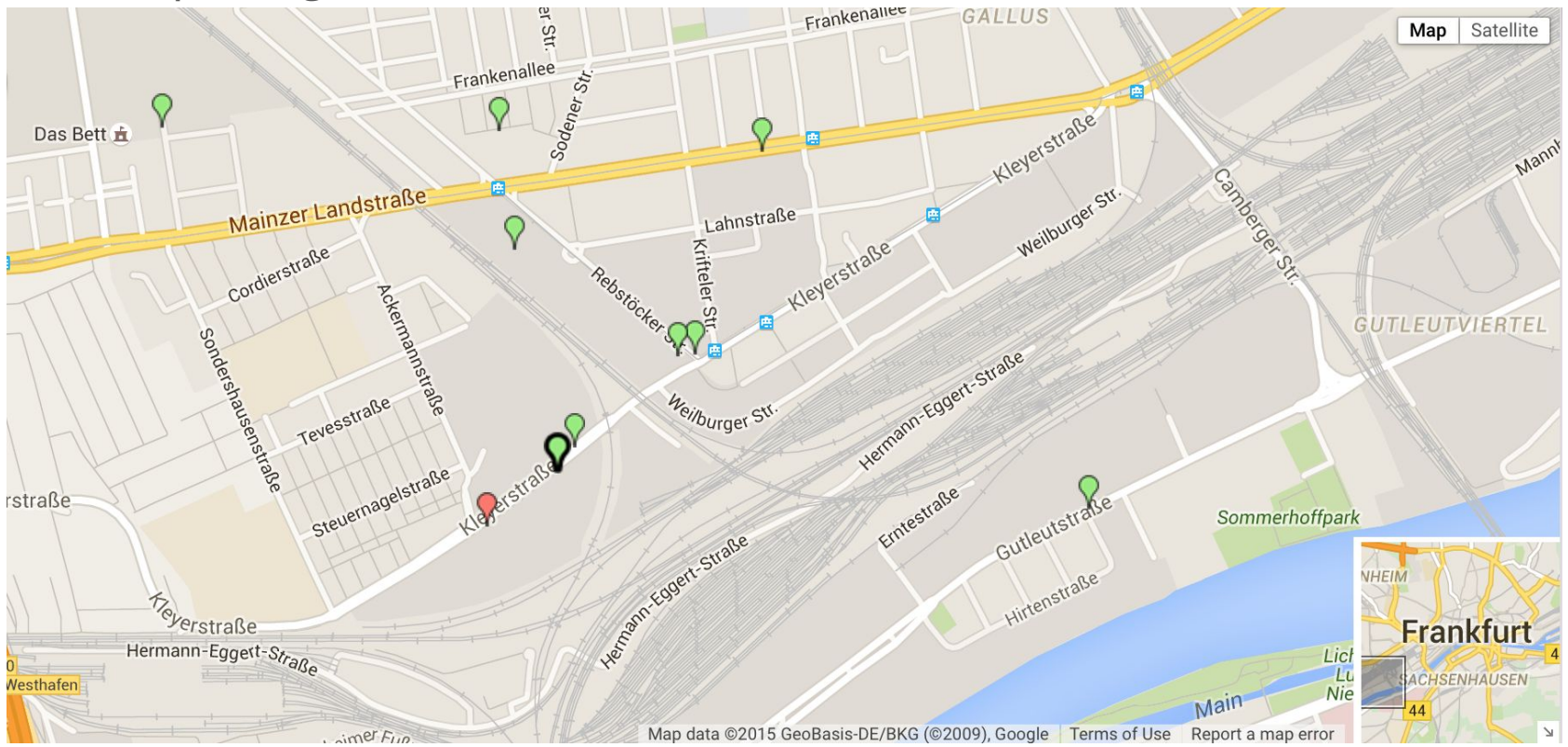




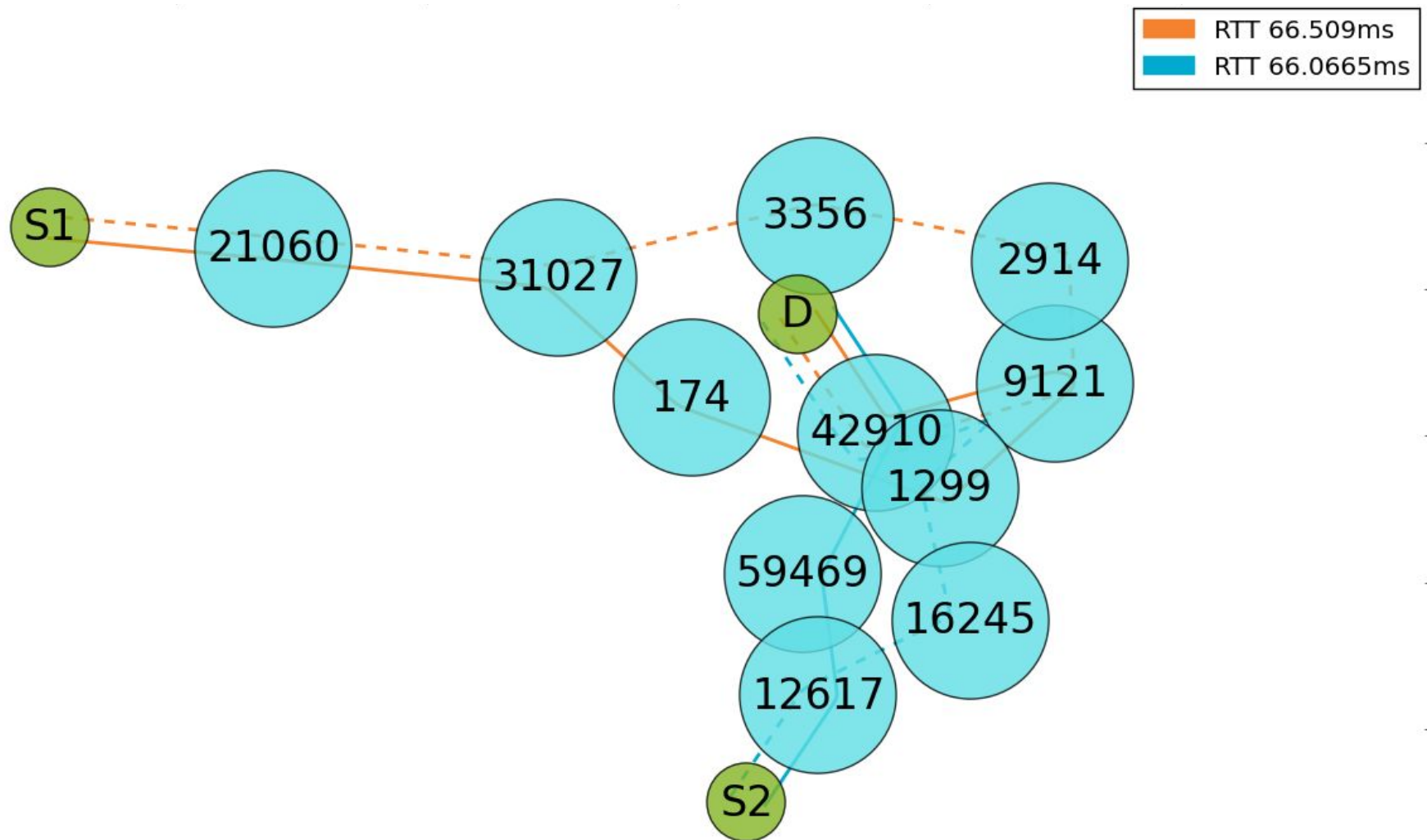
# How is my neighbour doing?

How is a neighbour in terms of geo location connected to the same destination?

- What does his path look like?
- Does it have a faster path?
- Is he peering somewhere I could do as well?

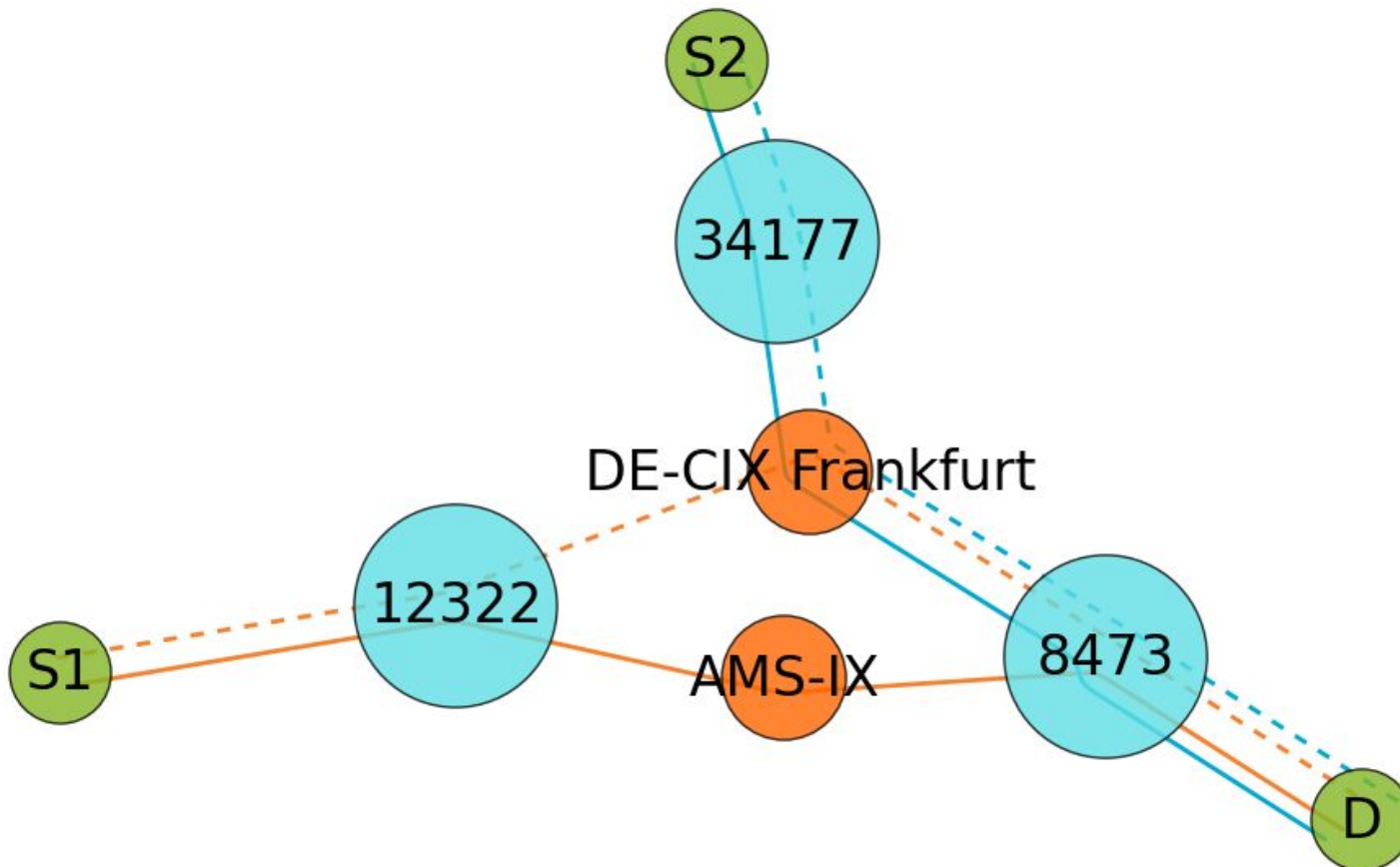
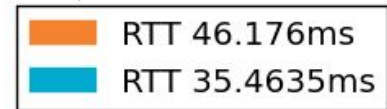


# ATEA (DK) -> SADECEHOSTING-COM (TR)





# PROXAD (FR) -> Bahnhof (SE)



# Roadmap

- **IPv6 support**
  - support for probeless networks
  - useful CLI output
  - support even more sources and destination at once
  - convince people to answer to ICMP packets for traceroutes!
  - ...





# The WINNERS



# It was so much fun!



# The PRICE

Box of stroopwaffles, bugfixes and even  
**THIS PRESENTATION**



# THANK YOU

Check out the project:



<http://git.io/v4lhE>

