Programmable Networks

By Miro, Roger, Tim, Jan and Jules

Find high-level requirements for NetComplete and visualize them

Visualize properties in a network graph

Visualization requirements

- Intuitive
- Easy sharing
- Clear
- Unambiguous
- Easy access to information

Input requirements derivation

- Derived from various papers, projects and startups
- Found:
  - Reachability
  - Failure resistance / N-connectivity
  - Security
  - Load balancing
  - Resource management
  - Preferential routing

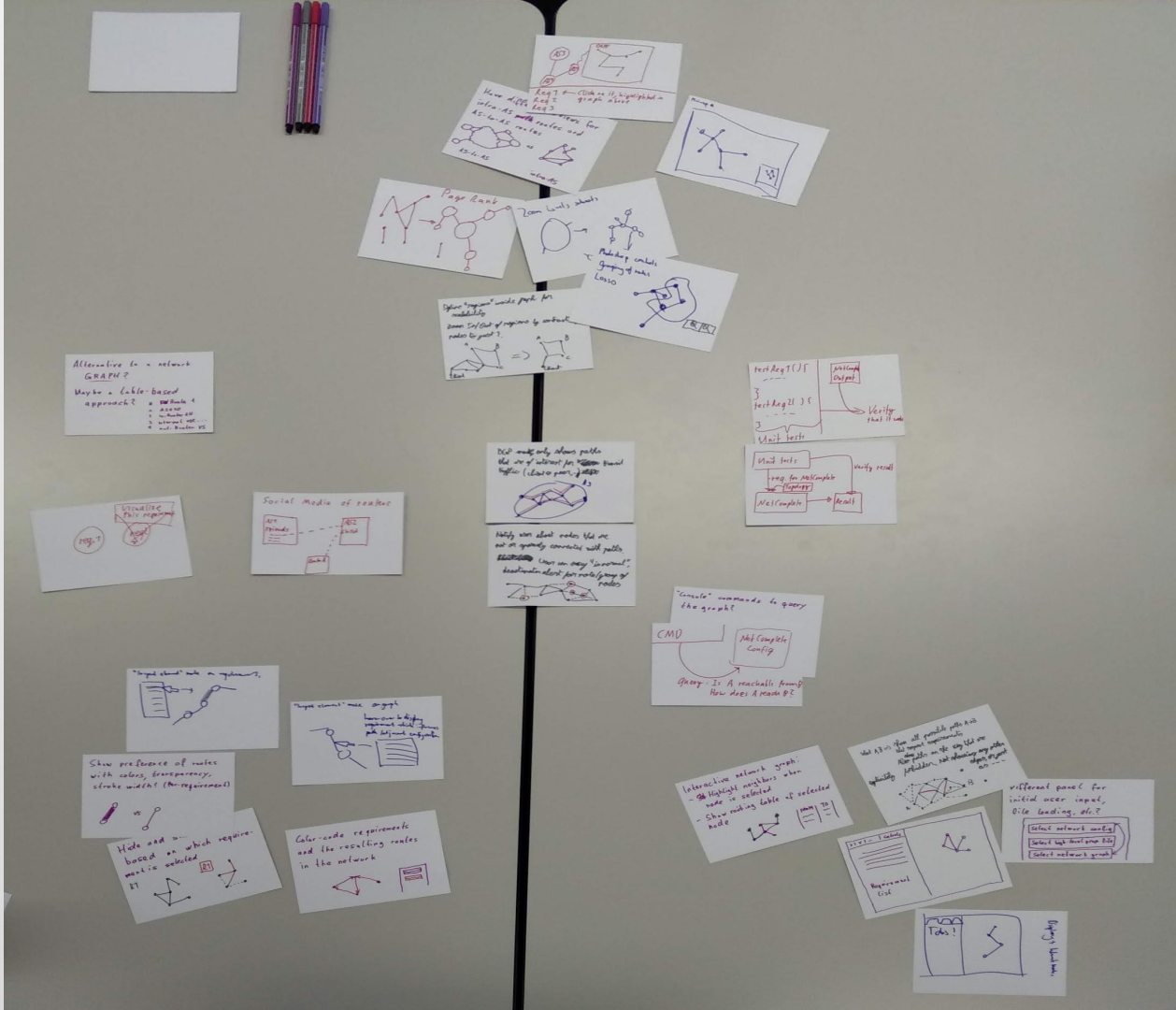
David Fischer

- ETH graduate, 26 years old, Junior Network Operator
- Ambitious & enthusiastic
- Goal:
  - Network stability, availability and performance; reduce human errors
  - Explainability of network topology
- Pain points:
  - Inheriting old burdens
  - Non-intuitive network configuration
  - Strongly dislikes repetitive tasks
  - Afraid of making severe mistakes

Hans-Peter Dullinger

- Stanford graduate, 52 years old, Senior Network Operator
- Key characteristics
  - Efficient & effective
  - Has too much to do
- Goal: A running network with little additional effort
- Pain points:
  - Stress
  - Worries about job security
  - Annoyed by big changes in the network (and the required work)

50 thoughts


Page Rank
Alternative to a network GRAPH?
Maybe a table-based approach?
Social Media of routers
Show preference of routes with colors, transparency, stroke width (for requirements)
Color-code requirements and the resulting routes in the network
Interactive network graph:
Tabs!

Three ideas

1. Split screen: Controls + graph
2. Social media for network graphs
3. Full screen graph with VR

Inspect | Requirements | Search

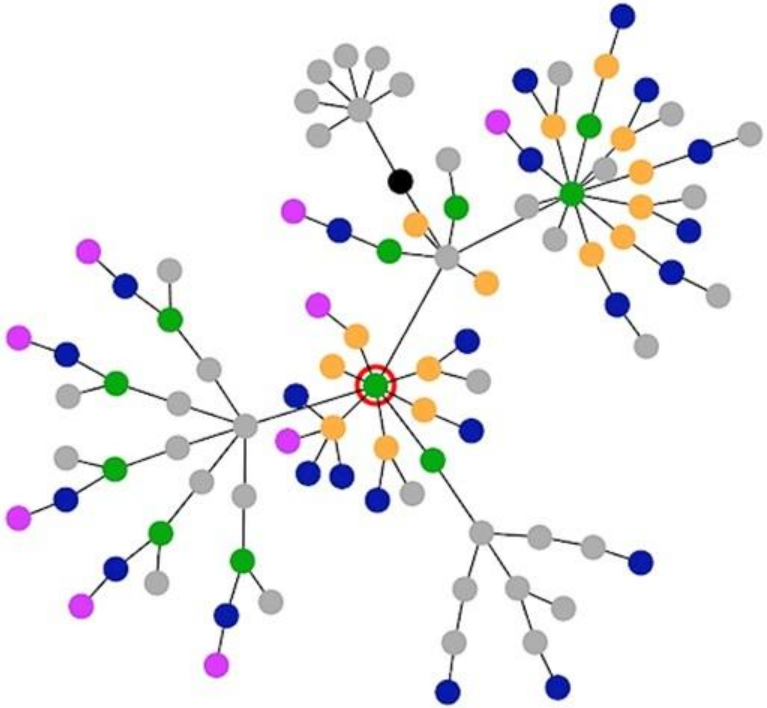
This Node

Hostname: group2.local
Type: EdgeRouter
IP: 192.168.123.123
NetMask: 255.255.255.0
AS: 123

Requirements

Req 123: 4-way connectivity
From a.group2.local
To b.group2.local

Req 124: Fault tolerance if Link **XY** fails
From c.group2.local
To d.group2.local



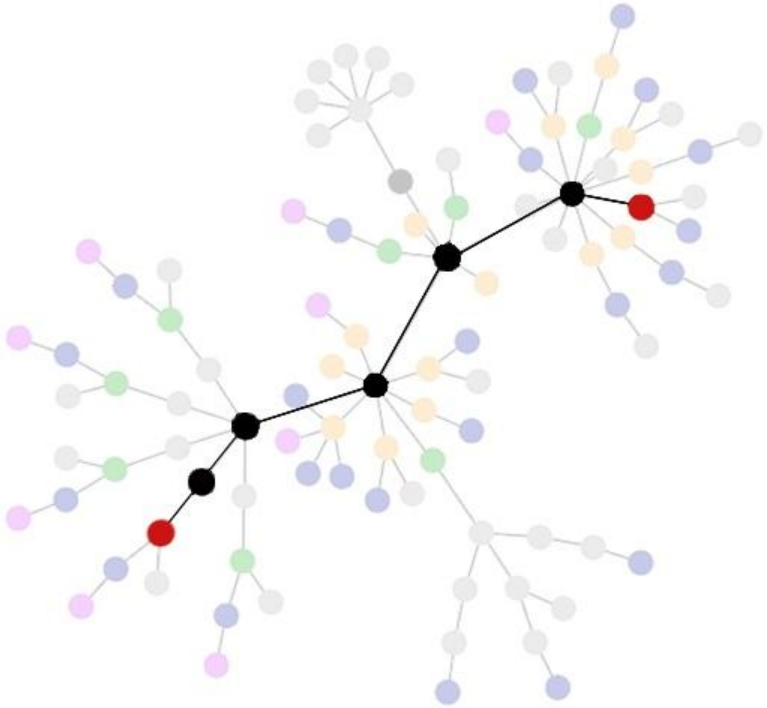
Inspect | Requirements | Search

Req 123: 4-way connectivity

Req 124: Fault tolerance if Link **XY** fails

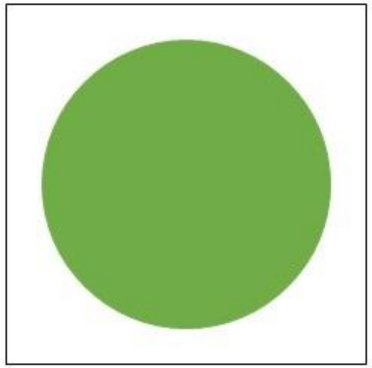
Req 125: Reachability

From	a.group2.local
To	b.group2.local



Strengths

+ Easy & intuitive navigation (familiar UI)
  + Search for named requirements
+ Detailed information
+ Good overview
  + Highlighted nodes / requirements
+ Easy to implement



Hostname: group2.local
Type: EdgeRouter
IP: 192.168.123.123
NetMask: 255.255.255.0
AS: 123

Neighbors:


192.168.123.124
a.group2.local


192.168.123.125
b.group2.local


192.168.123.126
c.group2.local


192.168.123.127
d.group2.local


192.168.123.128
e.group2.local


192.168.123.129
f.group2.local

display more...

Groups:

Req 123: 4-way connectivity
from  a.group2.local
192.168.123.124
to  b.group2.local
192.168.123.125

Req 124: Fault tolerance if Link **XY** fails
from  c.group2.local
192.168.123.126
to  d.group2.local
192.168.123.127

display more...

Blocked:

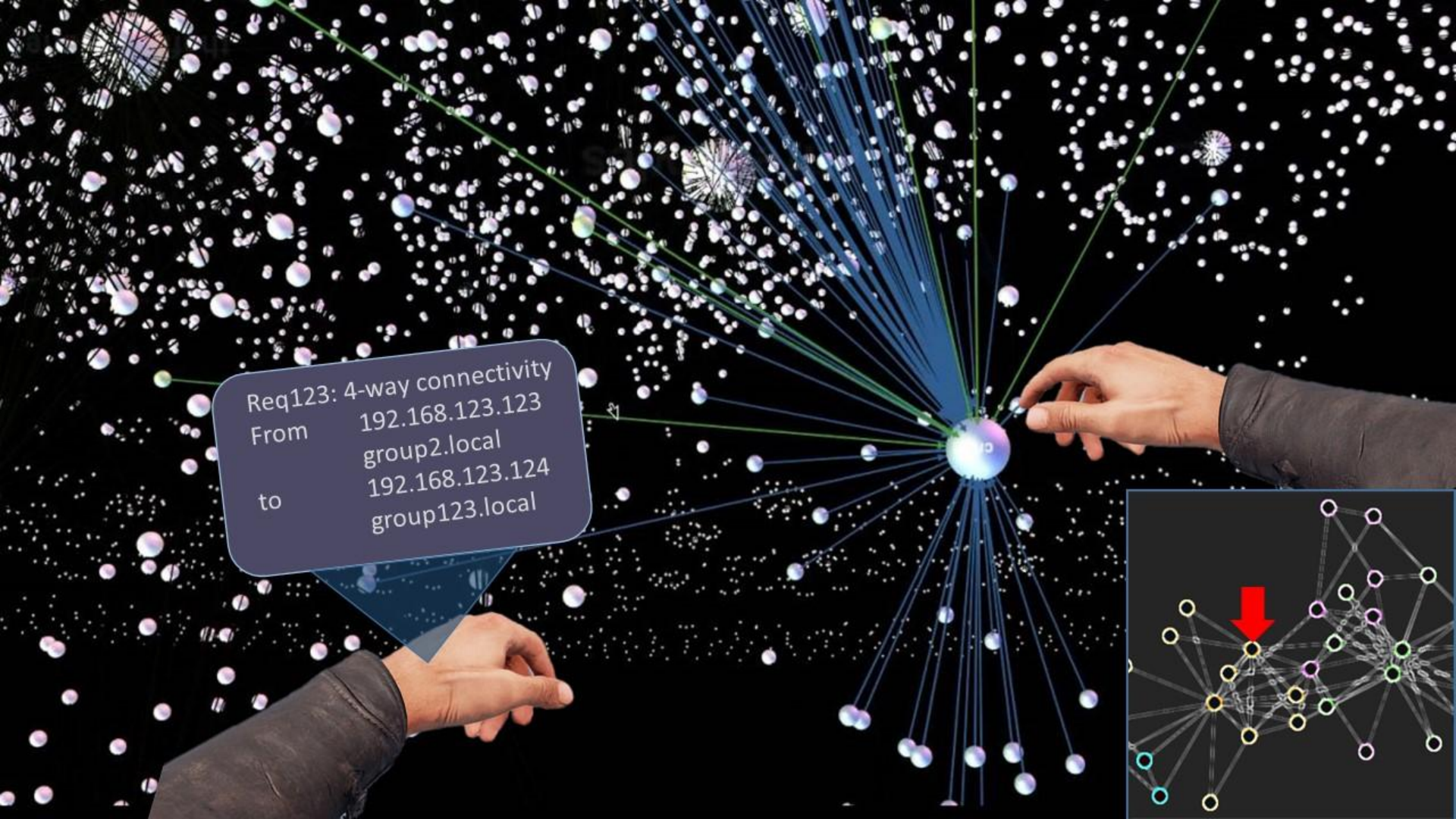

192.168.123.130
g.group2.local


192.168.123.140
h.group2.local

display more...

Strengths

+ Good local view
+ Easy to learn
  + Familiar UI
+ Hides complexities of large network graphs
+ Concise profile for each requirement
+ Requirements as important as nodes


Req123: 4-way connectivity
From 192.168.123.123
group2.local
to 192.168.123.124
group123.local

Strengths

+ Simple, hides clutter
+ Graph is central
+ Very visual way to show network to others
+ Various possible input methods
+ Also possible as desktop or AR application

Feedback or Questions?

Resources

- Personas:
  - Both from https://www.pexels.com (10.2018), attribution not required