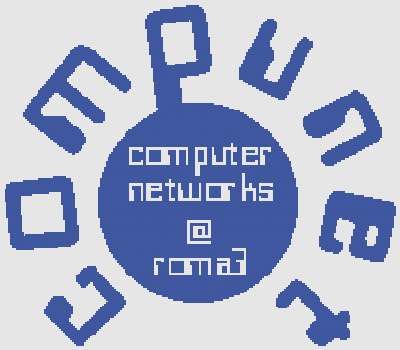
NetML
Network Markup Language

Ivan Santarelli
i.santarelli@tiscali.it

Alexandra Bellogini
a.bellogini@tiscali.it



Computer Networks Research Group
University of Rome 3

NetML
Network Markup Language

- A language for describing computer networks
- Based on XML
- Describes a network at different levels of abstraction
- The focus is on conceptual aspects rather than on configuration issues
- Helps simplify network design
- Allows quick specification and implementation of a network test-bed

NetML Capabilities

- ***Network topology***
  - ***Data link level***
  - ***AS (Autonomous System) level***
- ***Router configurations***
  - ***Interfaces***
  - ***BGP***
  - ***RIP***
  - ***Policies and Lists***

The Netml approach

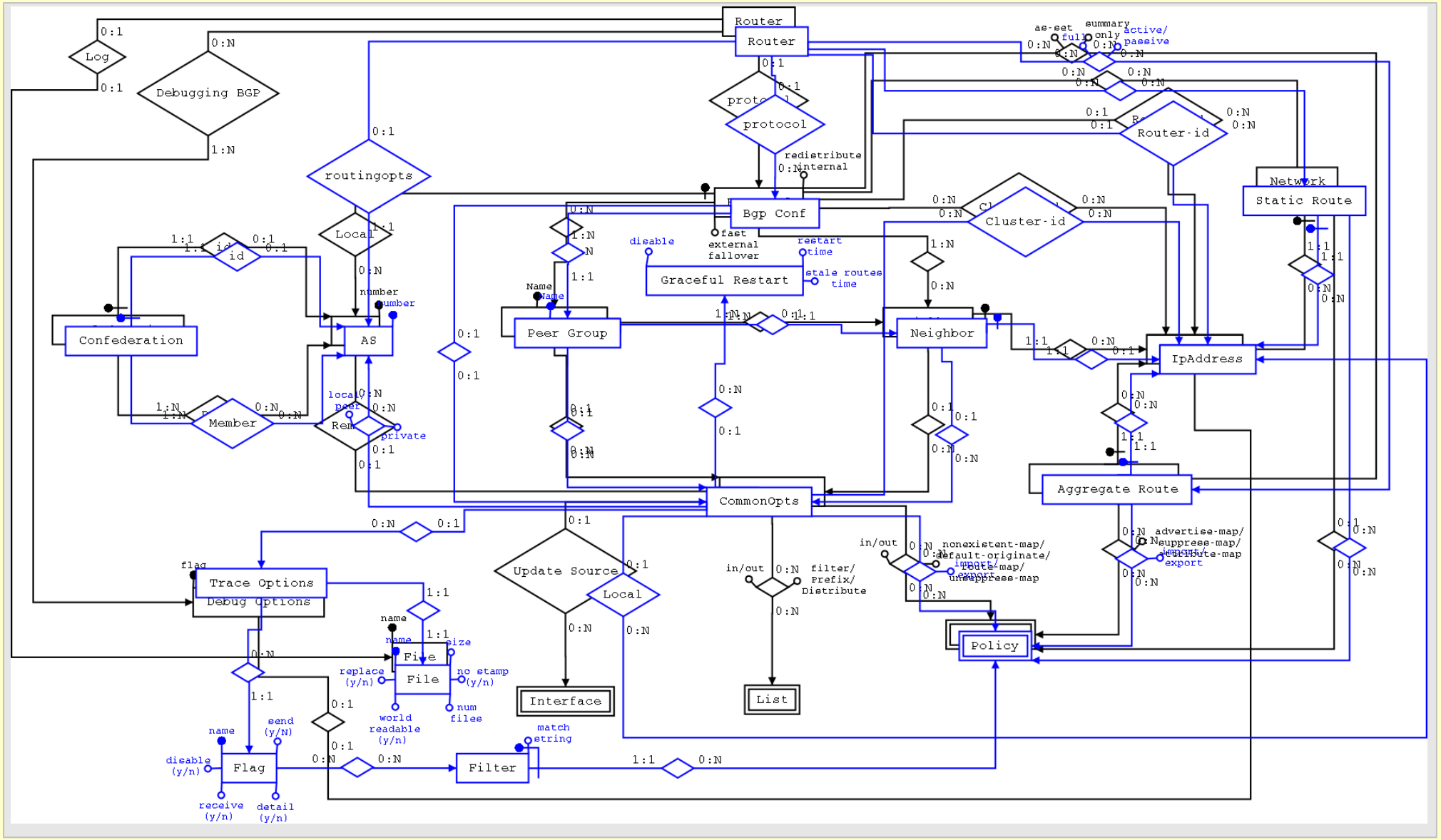
- Analysis of the current implementation of the protocols
  - Cisco, Juniper, Zebra
- Description using the ER (Entity-Relationship) model
  - ER-schemas for Cisco, Juniper, Zebra

The Netml approach (2)

- Compare schemas
- Identify a common schema: common features plus the most interesting vendor-dependent parts
- Translation of the ER common schema into an XML schema (an improved version of a DTD)

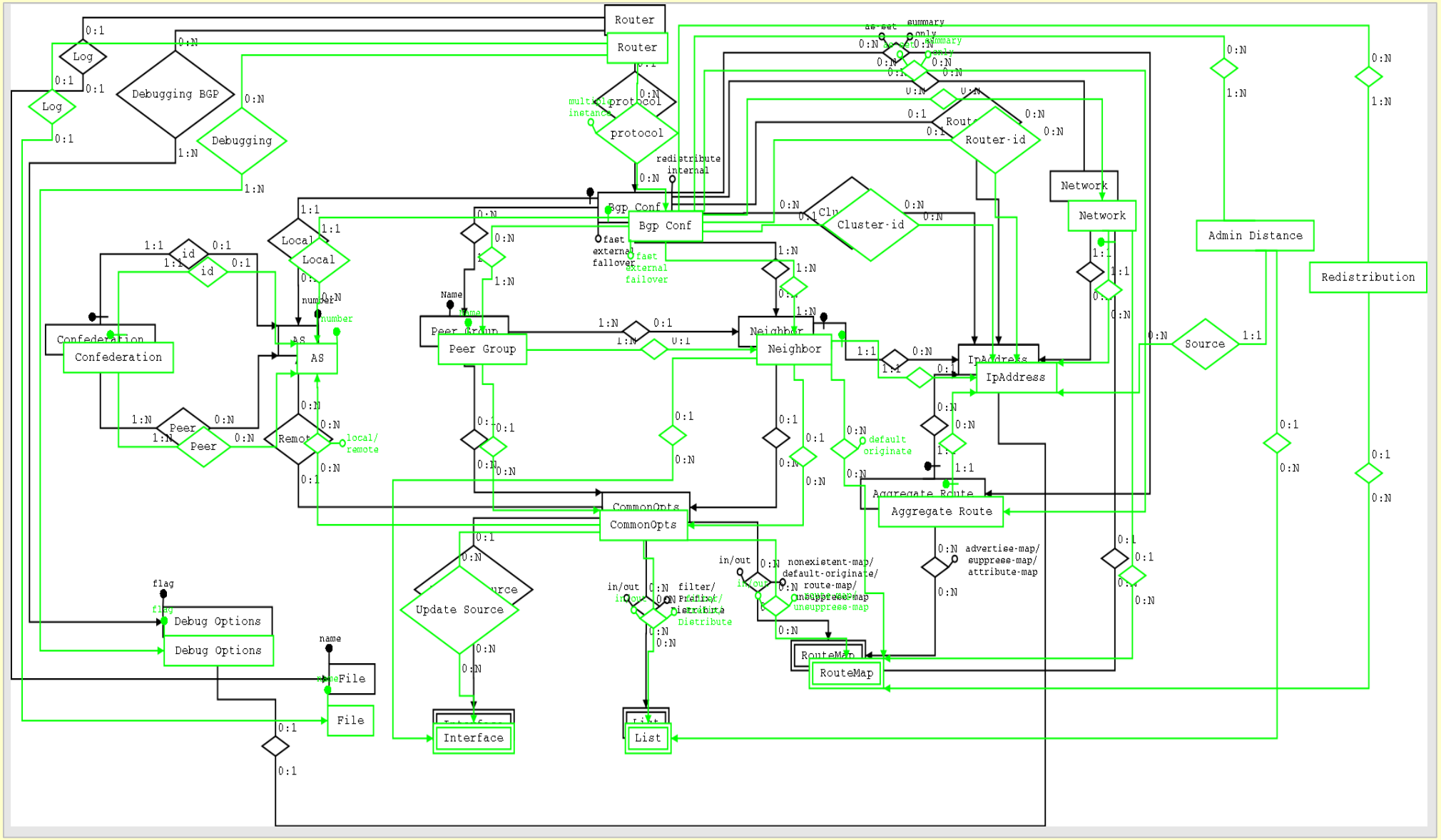
Cisco

Juniper

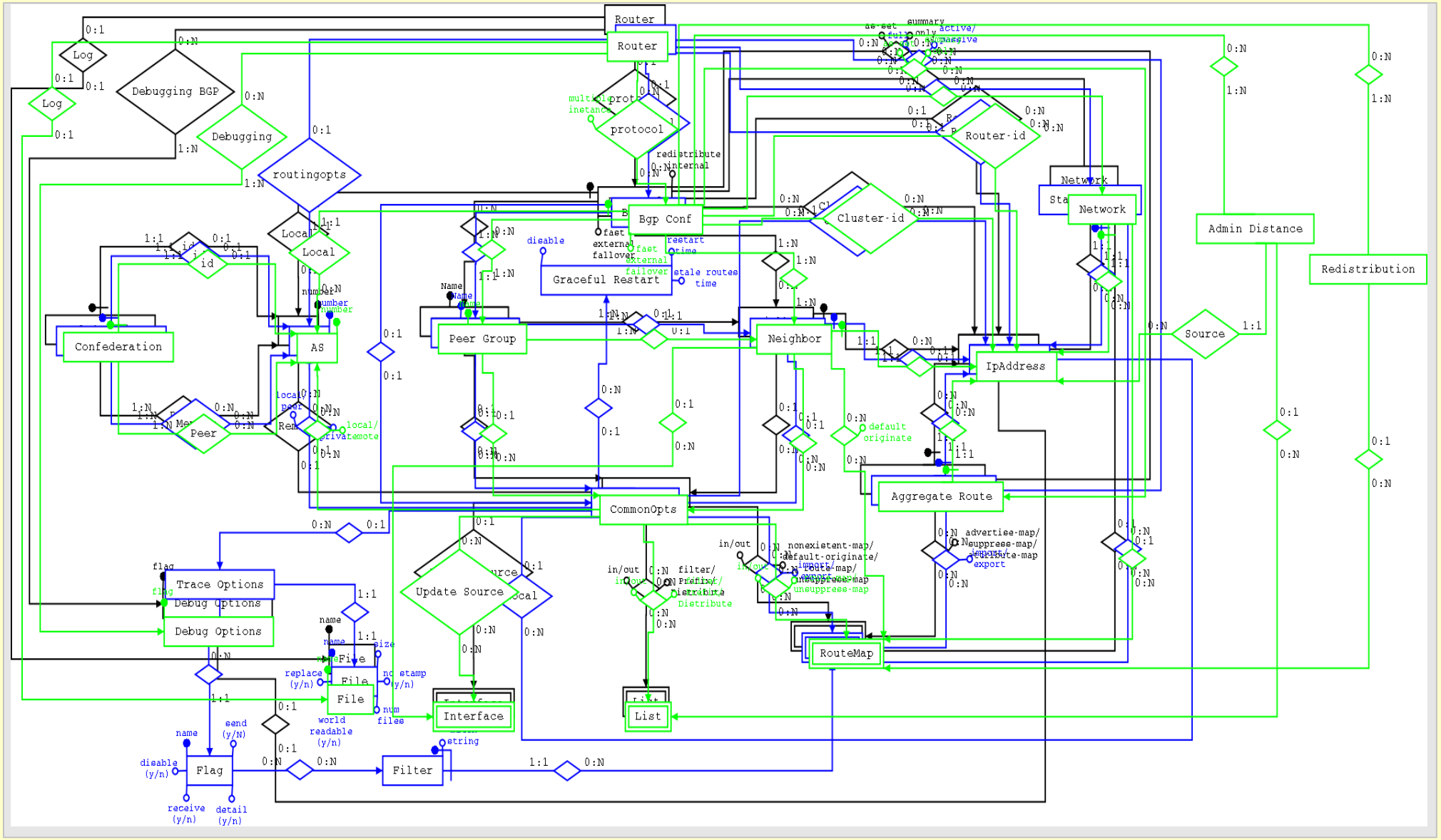


Cisco

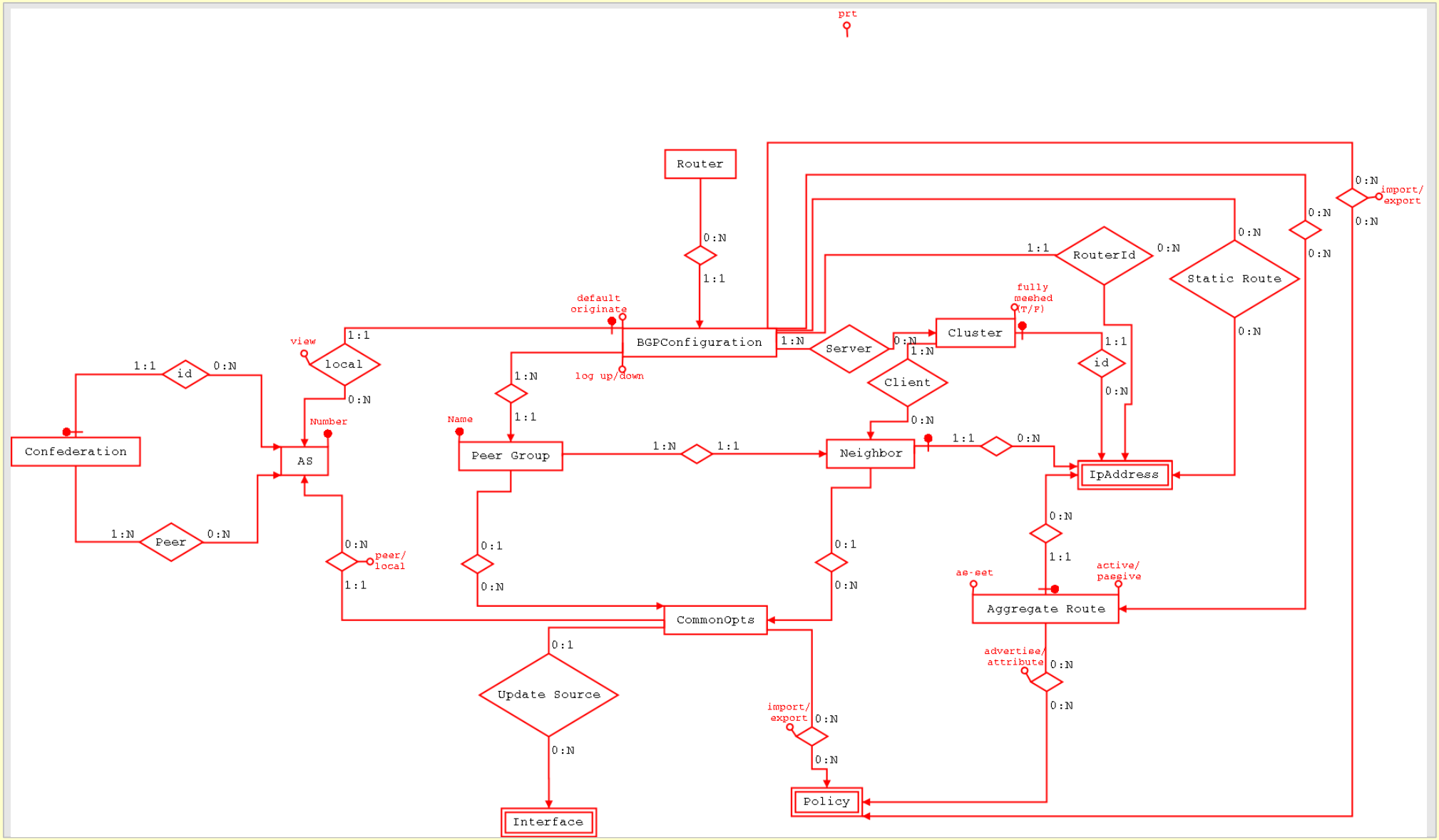
Zebra


Router
Router
Log
Log
Debugging BGP
Debugging
protocol
protocol
multiple
instance
redistribute
internal
Bgp Conf
Bgp Conf
fast
external
failover
Cluster-id
Network
Network
Admin Distance
Redistribution
Source
Local
Local
id
id
number
AS
Confederation
Confederation
Peer
Peer
Remot
local/
remote
Peer Group
Neighbor
IpAddress
default
originate
Aggregate Route
CommonOpts
Update Source
in/out
filter/
prefix/
distribute
Distribute
nonexistent-map/
default-originate/
route-map/
unsuppress-map
advertise-map/
suppress-map/
attribute-map
RouteMap
Debug Options
flag
name
File
Interface
List

Cisco Juniper Zebra



BGP in NetML: Entity Relationship



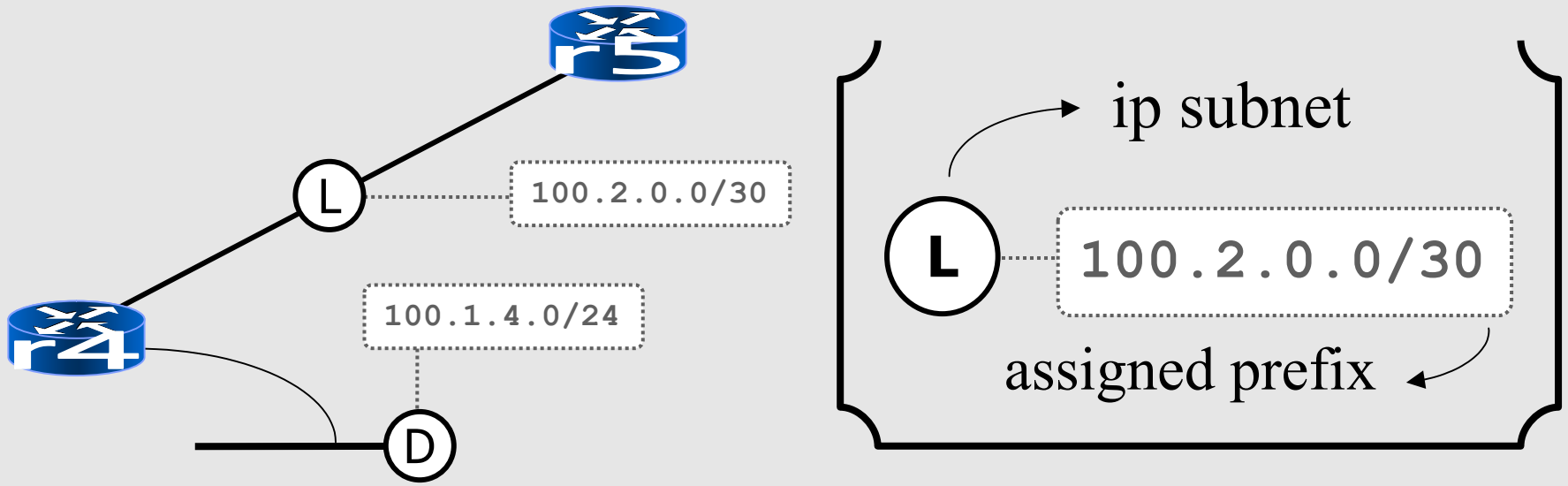
The NetML tools

- An XML validating parser
- An XML translator to vendor-specific configuration languages
  - Uses XSLT (eXtensible Stylesheet Language - Transformations)

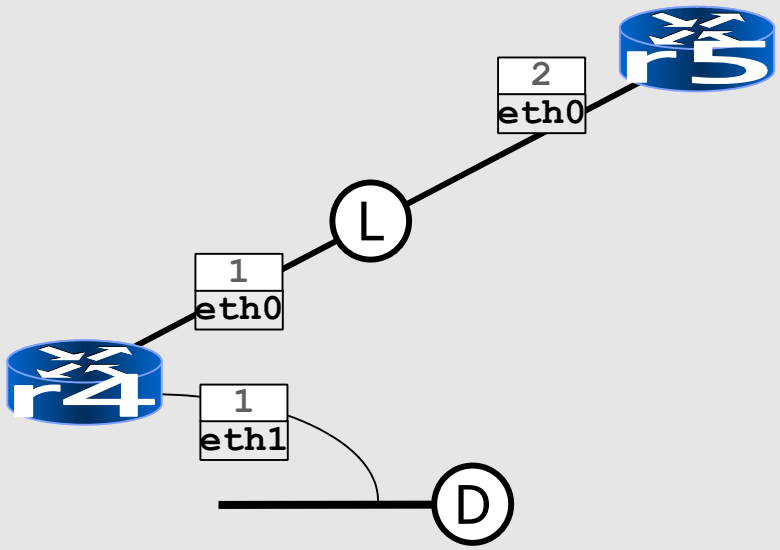
What the user can do

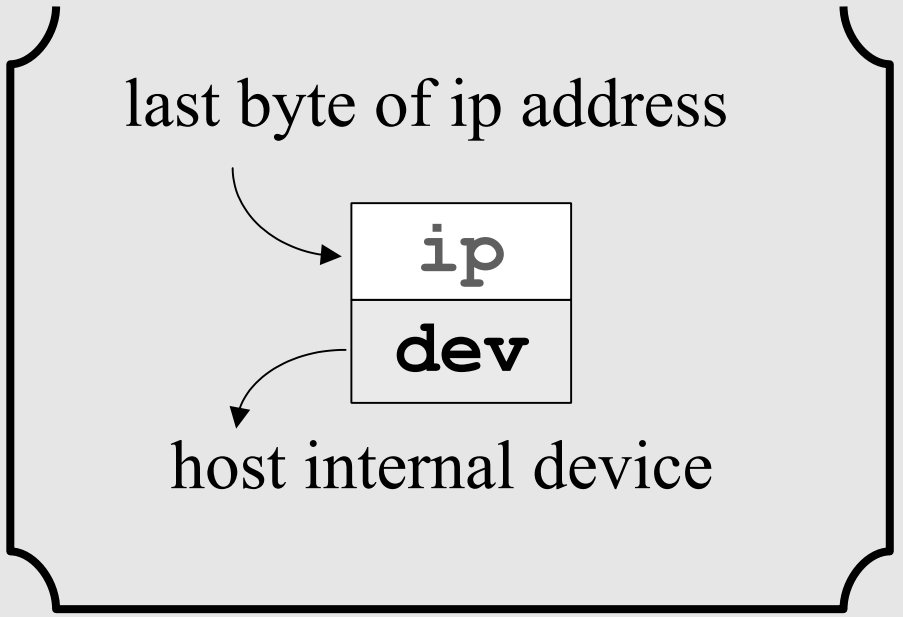
- Describe the network using NetML grammar
  - With a text editor
  - With an XML editor (with syntax checking and tag auto-completion)
- Use the tools
  - Generate the configuration of each router
  - Generate a script for Netkit (virtual network environment)

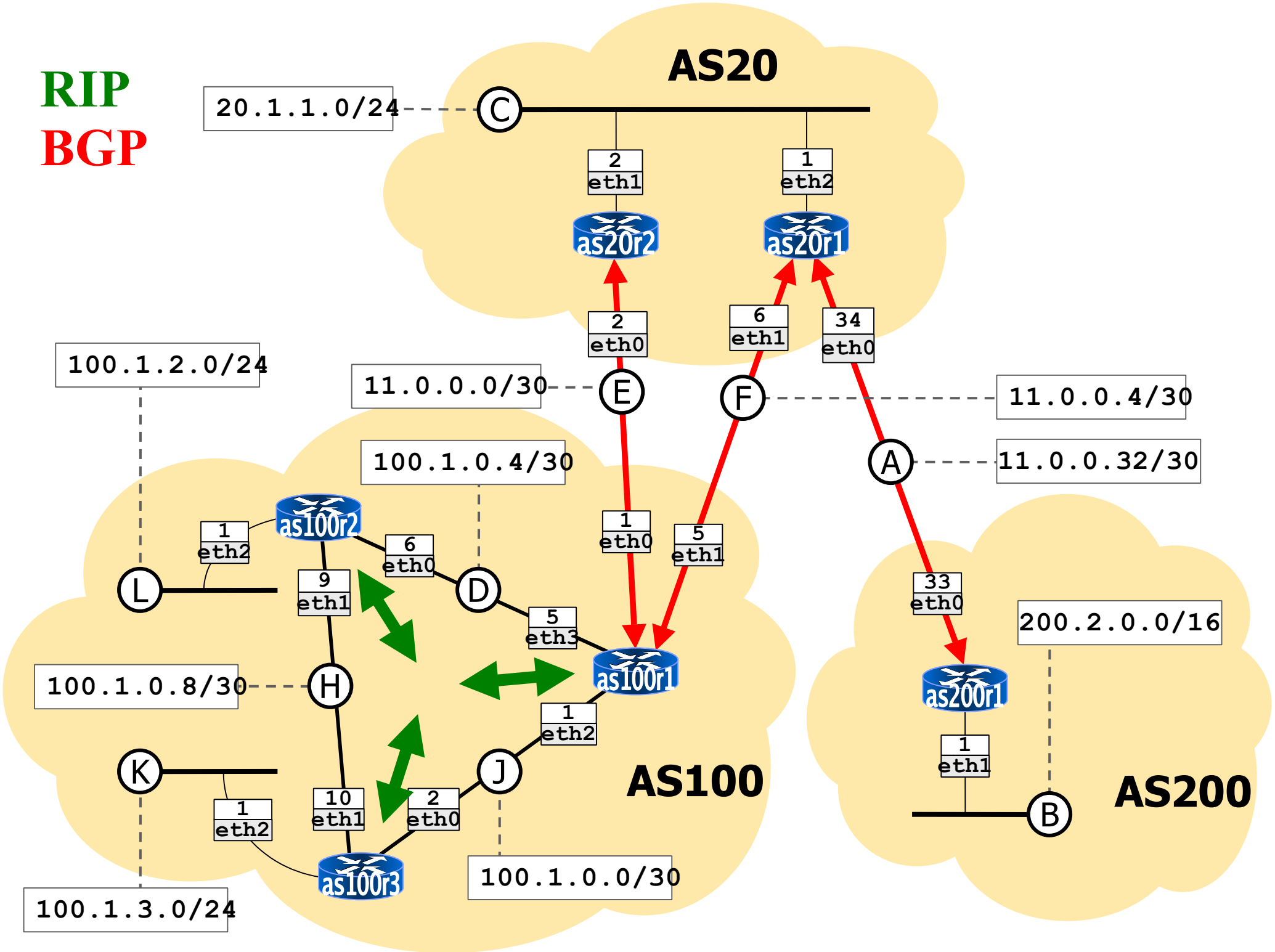
An example network in NetML Conventions



An example network in NetML Conventions (2)






RIP
BGP
AS20
20.1.1.0/24
C
2
eth1
1
eth2
as20r2
as20r1
2
eth0
6
eth1
34
eth0
100.1.2.0/24
11.0.0.0/30
E
F
11.0.0.4/30
A
11.0.0.32/30
100.1.0.4/30
as100r2
1
eth2
6
eth0
1
eth0
5
eth1
L
9
eth1
D
5
eth3
33
eth0
200.2.0.0/16
100.1.0.8/30
H
as100r1
as200r1
1
eth2
1
eth1
K
J
AS100
AS200
1
eth2
10
eth1
2
eth0
B
as100r3
100.1.0.0/30
100.1.3.0/24

NetML
Top level structure

```
<NetML>

  <AS-List>
  </AS-List>

  <ebgpPeerings>
  </ebgpPeerings>

  <Networks>
  </Networks>

  <Routers>
  </Routers>

</NetML>
```

NetML
Top level structure

```
<NetML>

  <AS-List>
  </AS-List>

  <ebgpPeerings>
  </ebgpPeerings>

  <Networks>
  </Networks>

  <Routers>
  </Routers>

</NetML>
```

The Autonomous System level

NetML
The AS level

```
<AS-List>
    <AS number="AS100" >
    ......
    </AS>
    <AS number="AS20" >.....</AS>
    <AS number="AS200" >.....</AS>
  </AS-List>
```

A list of all the Autonomous Systems present in the network

NetML
The AS level

```
<AS number="AS100" >
   <InternalNetworks>
      <n id="n-D"/>
      <n id="n-H"/>
      <n id="n-J"/>
      <n id="n-L"/>
      <n id="n-K"/>
   </InternalNetworks>
   <BorderRouters>
      <r id="r_100_1"/>
   </BorderRouters>
</AS>
```

The references to networks are described in the “Network” section

```
<Networks>
 <n id="n-D">
    ...
 </n>
</Networks>
```

NetML
The AS level

```
<AS number="AS100" >
   <InternalNetworks>
      <n id="n-D"/>
      <n id="n-H"/>
      <n id="n-J"/>
      <n id="n-L"/>
      <n id="n-K"/>
   </InternalNetworks>
   <BorderRouters>
      <r id="r_100_1"/>
   </BorderRouters>
</AS>
```

The routers referenced here are the AS's border routers

```
<RouterConf id="r_20_1"
 Hostname="as20r1>
 ...
 ...
</RouterConf>
```

NetML
Top level structure

```
<NetML>

  <AS-List>
  </AS-List>

  <ebgpPeerings>
  </ebgpPeerings>

  <Networks>
  </Networks>

  <Routers>
  </Routers>

</NetML>
```

Contains the list of BGP peerings held between different AS's

NetML
Specifying peerings

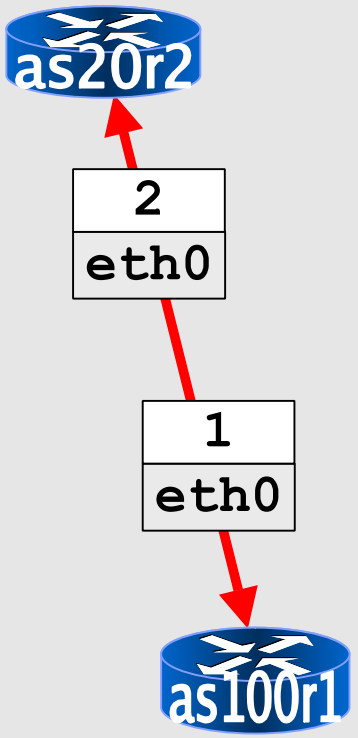
```
<ebgpPeerings>
  <P id="p01">... </P>
  <P id="p02">... </P>
  <P id="p03">... </P>
</ebgpPeerings>
```

Contains the list of BGP peerings held between different AS's

NetML
Specifying peerings (2)

```
<ebgpPeerings>
  <P id="p01">
    <PeeringSide>
      <as>AS100</as>
      <router>r_100_1</router>
      <iface>eth0</iface>
    </PeeringSide>
    <PeeringSide>
      <as>AS20</as>
      <router>r_20_2</router>
      <iface>eth0</iface>
    </PeeringSide>
  </P>
  <P id="p02">... </P>
  <P id="p03">... </P>
</ebgpPeerings>
```

Each peering is made up by two peering sides



NetML
Top level structure

```
<NetML>

  <AS-List>
  </AS-List>

  <ebgpPeerings>
  </ebgpPeerings>

  <Networks>
  </Networks>

  <Routers>
  </Routers>

</NetML>
```

Contains information about networks

NetML
Networks and collision domains

```
<Networks>
   <n id="n-A">...</n>
   <n id="n-B">...</n>
   <n id="n-C">...</n>
</Networks>
```

List of networks

NetML
Networks and collision domains

```
<Networks>
  <n id="n-A">
    <networkAddress>11.0.0.32/30</networkAddress>
    <CollisionDomains>
      <c id="A">
        <iface r_id="r_200_1" i
        <iface r_id="r_20_1" if=
      </c>
    </CollisionDomains>
  </n>
  <n id="n-B"> </n>
  <n id="n-C"> </n>
</Networks>
```

The address and the netmask of the network

NetML
Networks and collision domains

```
<Networks>
  <n id="n-A">
    <networkAddress>11.0.0.32/30
    <CollisionDomains>
      <c id="A">
        <iface r_id="r_200_1" if="eth0"/>
        <iface r_id="r_20_1" if="eth0"/>
      </c>
    </CollisionDomains>
  </n>
  <n id="n-B"> </n>
  <n id="n-C"> </n>
</Networks>
```

One network can have several collision domains

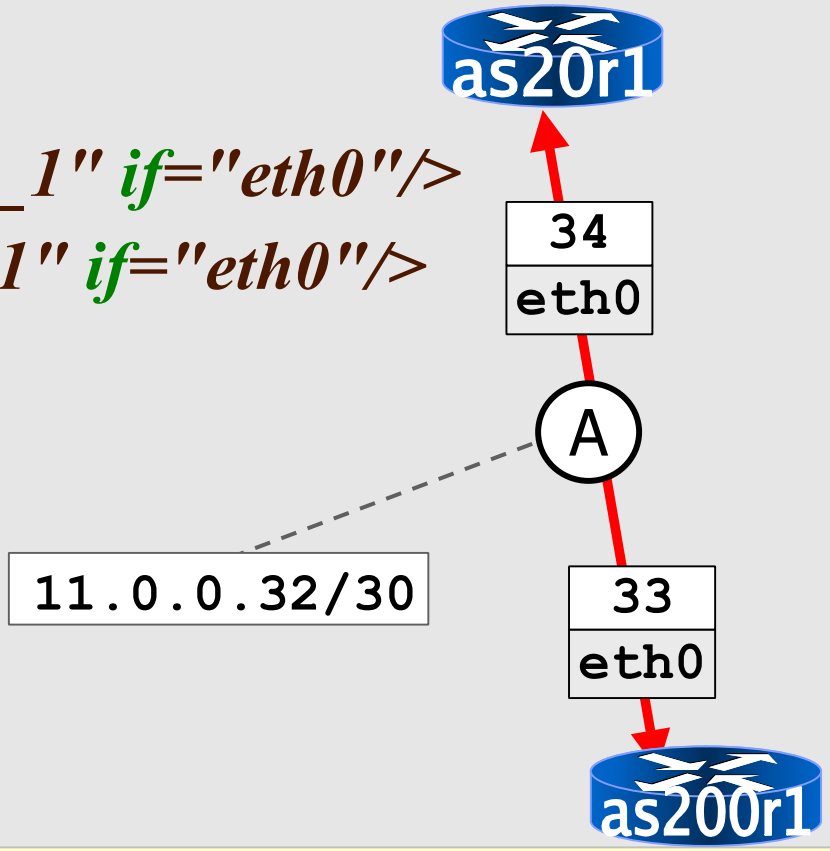
NetML
Networks and collision domains

```
<Networks>
  <n id="n-A">
    <networkAddress>11.0.0.32/30
    <CollisionDomains>
      <c id="A">
        <iface  r_id="r_200_1" if="eth0"/>
        <iface  r_id="r_20_1" if="eth0"/>
      </c>
    </CollisionDomains>
  </n>
  <n id="n-B"> </n>
  <n id="n-C"> </n>
</Networks>
```

The interfaces that belong to the collision domain

NetML
Networks and collision domains

```
<Networks>
  <n id="n-A">
    <networkAddress>11.0.0.32/30</networkAddress>
    <CollisionDomains>
      <c id="A">
        <iface  r_id="r_200_1" if="eth0"/>
        <iface  r_id="r_20_1" if="eth0"/>
      </c>
    </CollisionDomains>
  </n>
  <n id="n-B"> </n>
  <n id="n-C"> </n>
</Networks>
```



NetML
Top level structure

```
<NetML>

  <AS-List>
  </AS-List>

  <ebgpPeerings>
  </ebgpPeerings>

  <Networks>
  </Networks>

  <Routers>
  </Routers>

</NetML>
```

Contains the list of the routers of the network

NetML
Router configuration

```
<Routers>

  <RouterConf id="r_20_1"
Hostname="as20r1">...</RouterConf>

  <RouterConf id="r_20_2"
Hostname="as20r2">...</RouterConf>

  <RouterConf id="r_100_1"
Hostname="as100r1">...</RouterConf>

    ...

</Routers>
```

Each router has its own configuration

NetML
Router configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
    <Interface name="eth0">
      <ipAddress>11.0.0.34/30</ipAddress>
    </Interface>
    <Interface name="eth1">...</Interface>
    <Policy name="RedistConnecteds">
      <Redistribution>
        <from_protocol>connected</from_protocol>
        <to_protocol>bgp</to_protocol>
      </Redistribution>
    </Policy>
  <BGPConf as="...">...</BGPConf>
  <RipConf> ..... </RIPConf>
</RouterConf>
```

The set of interfaces

NetML
Router configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
   <Interface name="eth0">...</Interface>
  <Interface name="eth1">...</Interface>
   <Policy name="RedistConnected">
     <Redistribution>
       <from_protocol>connected</from_protocol>
       <to_protocol>bgp</to_protocol>
     </Redistribution>
   </Policy>
  <BGPConf as="...">...</BGPConf>
  <RipConf> ..... </RIPConf>
</RouterConf>
```

The set of policies

NetML
BGP configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
  <BGPConf as="20">
    <StaticRoutes>
     <s address="0.0.0.0/0"/>
     <s address="..."/>
   </StaticRoutes>
     <Policy type="export">RedistConnected</Policy>
     <PeerGroup name="EBGP">... </PeerGroup>
  </BGPConf>
</RouterConf>
```

Specify static routes

NetML
BGP configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
  <BGPConf as="20">
    <StaticRoutes>
     <s address="0.0.0.0/0"/>
     <s address="..."/>
   </StaticRoutes>
     <Policy type="export">RedistConnected</Policy>
     <PeerGroup name="EBGP">... </PeerGroup>
  </BGPConf>
</RouterConf>
```

Specify which policies to apply

NetML
BGP configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
  <BGPConf as="20">
    <StaticRoutes>...</StaticRoutes>
    <Policy type="export">RedistConnected</P
    <PeerGroup name="EBGP">
      <Neighbor address="11.0.0.33">
        <Description>Router as200r1</Description>
        <Default-Originate/>
        <Peer-AS name="200"/>
        <List type="Prefix" inOut="export">defaultOut</List>
        <List type="Prefix" inOut="import">customerIn</List>
      </Neighbor>
      <Neighbor address="11.0.0.5">.....</Neighbor>
    </PeerGroup>
    <PeerGroup name="IBGP">.....</PeerGroup>
  </BGPConf>
</RouterConf>
```

Main configuration under PeerGroups

NetML
RIP configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
    <RipConf>
      <neighbour>
        <address>100.1.0.0/24</address>
      </neighbour>
    <policy type=”export” name=”redist-conn”/>
    </RipConf>
</RouterConf>
```

Specify where RIP must be enabled

NetML
RIP configuration

```
<RouterConf id="r_20_1" Hostname="as20r1">
    <RipConf>
      <neighbour>
        <address>100.1.0.0/24</address>
      </neighbour>
    <policy type=”export” name=”redist-conn”/>
    </RipConf>
</RouterConf>
```

Reference to a previously specified policy

```
<RouterConf id="r_20_1" ...>
...
    <Policy name="redist-conn">...</Policy>
...
</RouterConf>
```

NetML

Live example

References and links

- *Automatic generation of XML DTDs from conceptual database schema*
  Carsten Kleiner and Udo W. Lipeck
  http://dbs.uni-leipzig.de/webdb/wien/015.pdf
- NetML Homepage:
  http://www.dia.uniroma3.it/~compunet/netml/
  http://giga.dia.uniroma3.it/~ivan/NetML/
- Compunet Homepage:
  http://www.dia.uniroma3.it/~compunet/
- Netkit web site:
  http://www.netkit.org/

NetML

Questions?