



APNIC

Asia Pacific Network Information Centre

APNIC Internet Routing Registry

Tutorial
Seoul

19 August 2003

Overview

- What is an IRR
 - Why use an IRR?
 - RPSL
 - IRR objects
- Recap attributes of some objects
- Routing Policy
 - What is routing policy?
 - Why define a Routing Policy?
 - Case studies and exercises
- Using the Routing Registry
 - IRRToolSet
- Summary



APNIC

Asia Pacific Network Information Centre

IRR

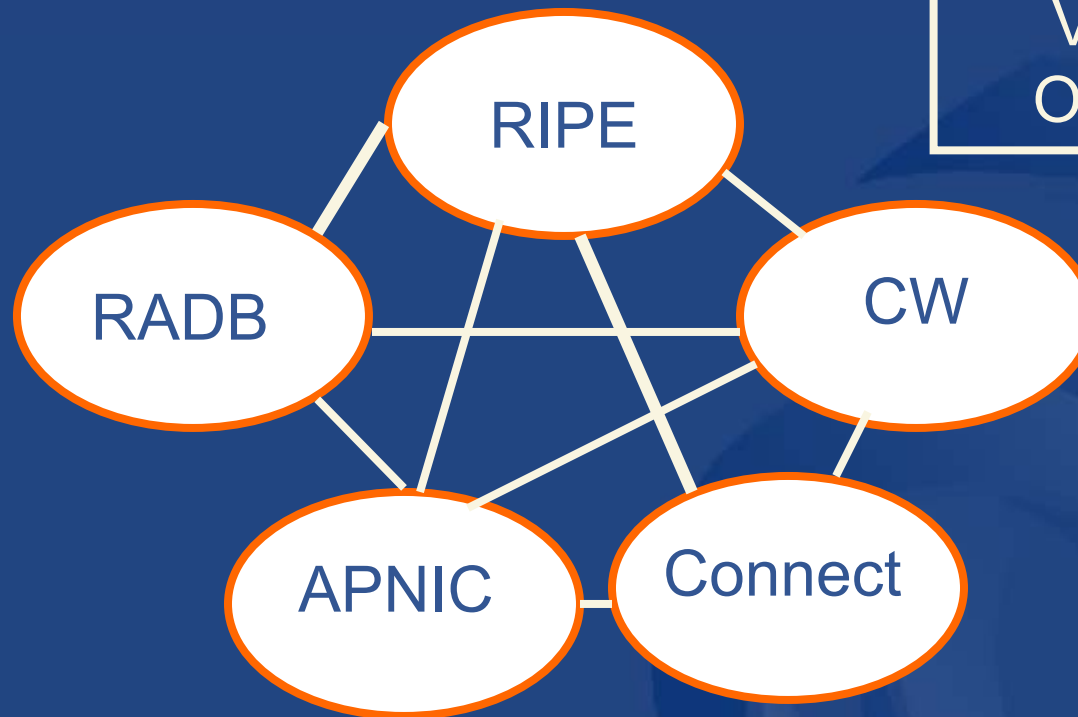
Internet Routing Registry



What is an IRR?

- Global Internet Routing Registry database
 - <http://www.irr.net/>
 - Uses RPSL
 - Established in 1995
- Stability and consistency of routing
 - network operators share information
- Both public and private databases
 - These databases are independent
 - but some exchange data
 - only register your data in one database

Internet Routing Registries



ARIN, ArcStar, FGC,
Verio, Bconnex,
Optus, Telstra, ...

IRR = APNIC RR + RIPE DB + RADB + C&W + ARIN + ...

Why use an IRR?

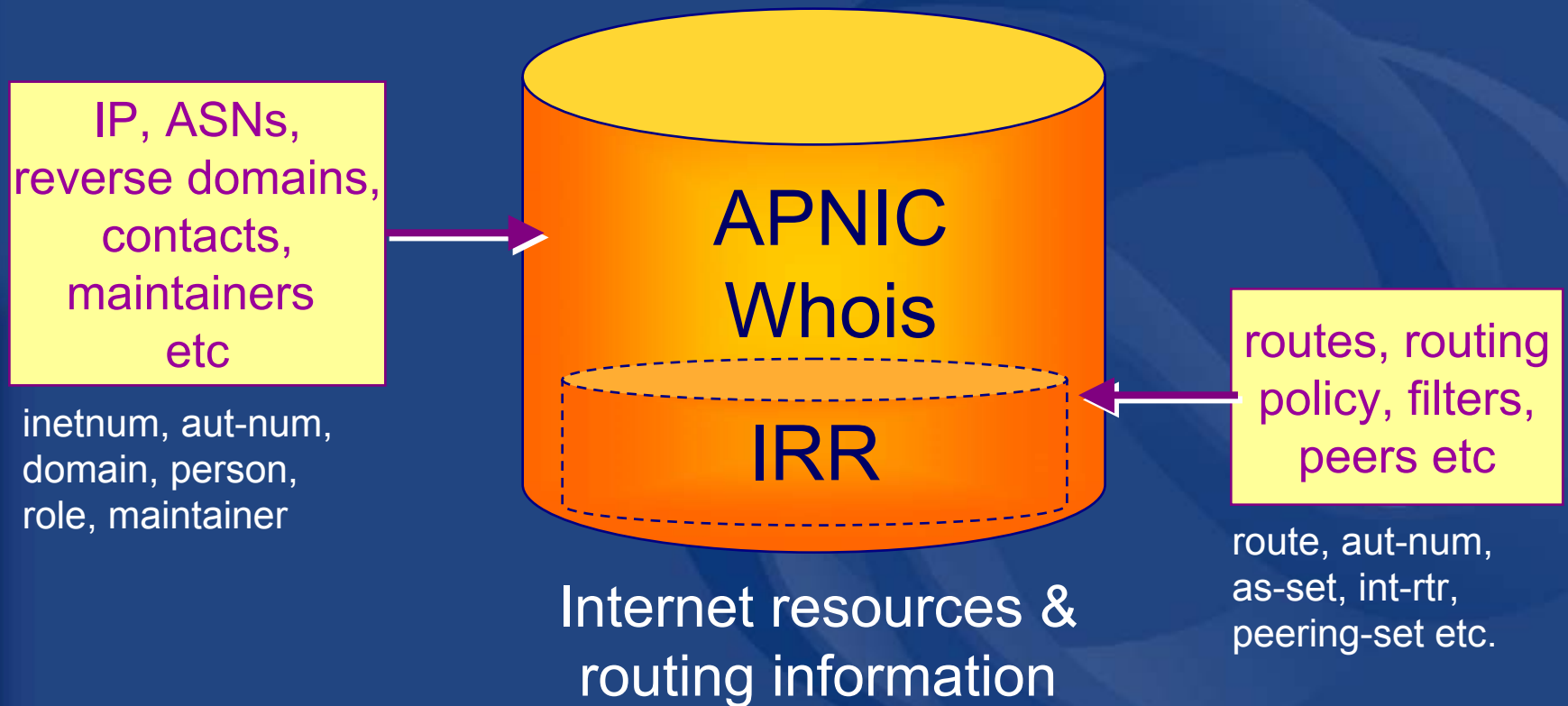
- Route filtering
 - Peering networks
 - A provider and its customer
- Network troubleshooting
 - Easier to locate routing problems outside your network
- Router configuration
 - By using IRRToolSet
 - <ftp.ripe.net/tools/IRRToolSet>
- Global view of routing
 - A global view of routing policy improves the integrity of Internet's routing as a whole.

APNIC Database & the IRR

- APNIC whois Database
 - Two databases in one
- Public Network Management Database
 - “whois” info about networks & contact persons
 - IP addresses, AS numbers etc
- Routing Registry
 - contains routing information
 - routing policy, routes, filters, peers etc.
 - APNIC RR is part of the global IRR

Integration of Whois and IRR

- Integrated APNIC Whois Database & Internet Routing Registry



RPSL

- Routing Policy Specification Language
 - Object oriented language
 - Based on RIPE-181
 - Structured whois objects
- Higher level of abstraction than access lists
- Describes things interesting to routing policy:
 - Routes, AS Numbers ...
 - Relationships between BGP peers
 - Management responsibility
- Relevant RFCs
 - Routing Policy Specification Language
 - Routing Policy System Security
 - Using RPSL in Practice



RFC
2622



RFC
2725



RFC
2650

IRR objects

- **route**
 - Specifies interAS routes
- **aut-num**
 - Represents an AS. Used to describe external routing policy
- **inet-rtr**
 - Represents a router
- **peering-set**
 - Defines a set of peerings
- **route-set**
 - Defines a set of routes
- **as-set**
 - Defines a set of **aut-num** objects
- **rtr-set**
 - Defines a set of routers
- **filter-set**
 - Defines a set of routes that are matched by its filter

www.apnic.net/db/ref/db-objects.html

Inter-related IRR objects

aut-num: AS1

...

tech-c: **KX17-AP**
mnt-by: **MAINT-EX**

...

route: 202.0.16/20

origin:

...

mnt-by: **MAINT-EX**

inetnum:

202.0.16 - 202.0.31.255

...

tech-c: **KX17-AP**
mnt-by: **MAINT-EX**

person:

...

nic-hdl: KX17-AP

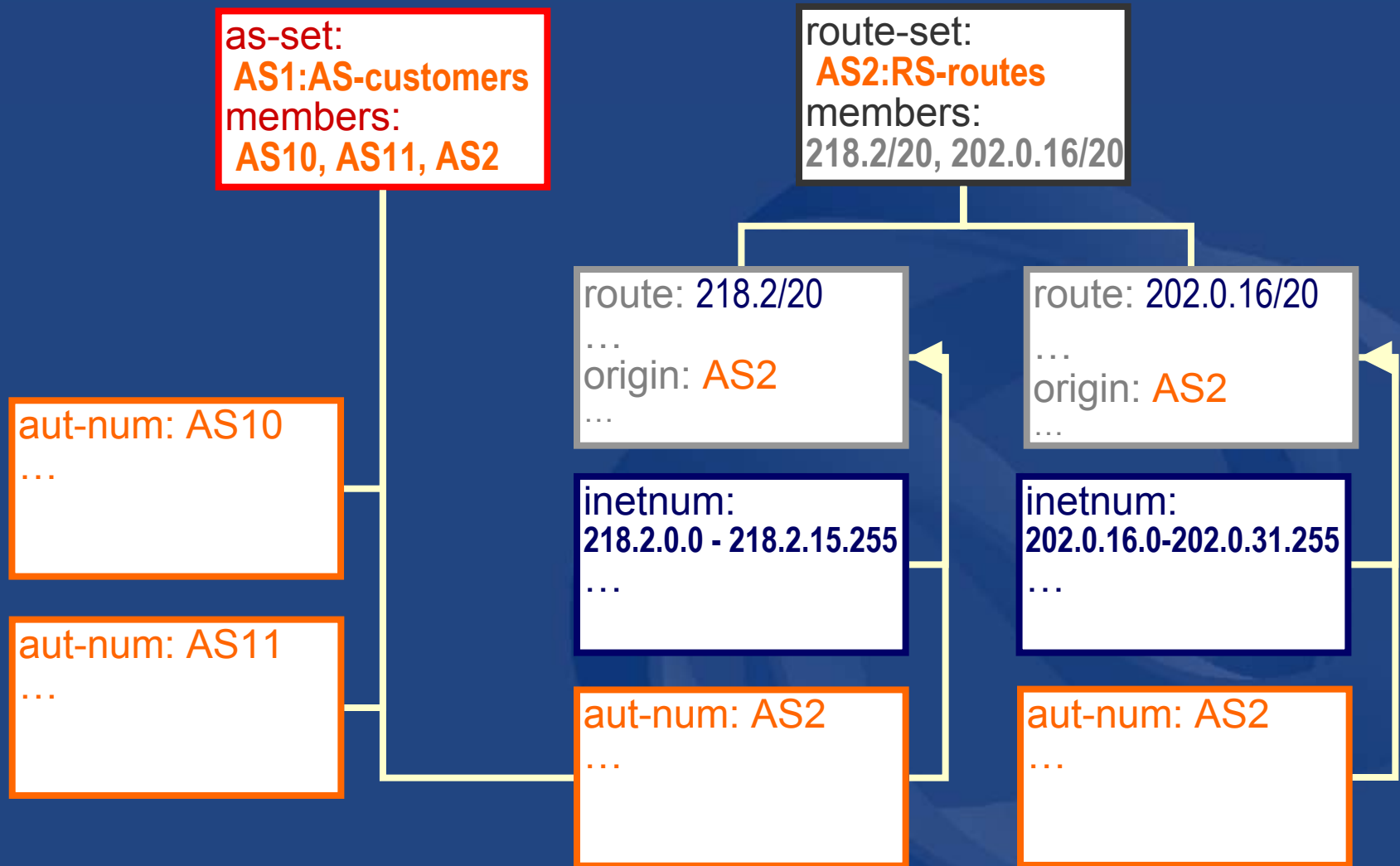
...

mntner: MAINT-EX

...



Inter-related IRR objects

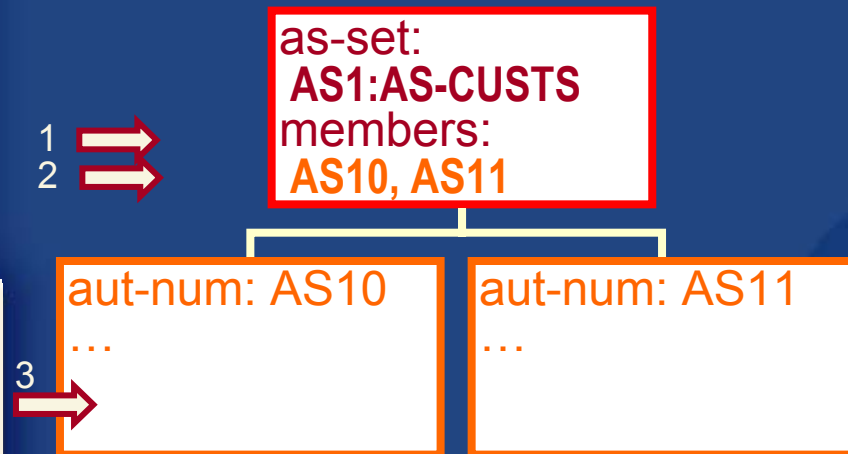


'Set-' objects and their members

- Two ways of referencing members

members

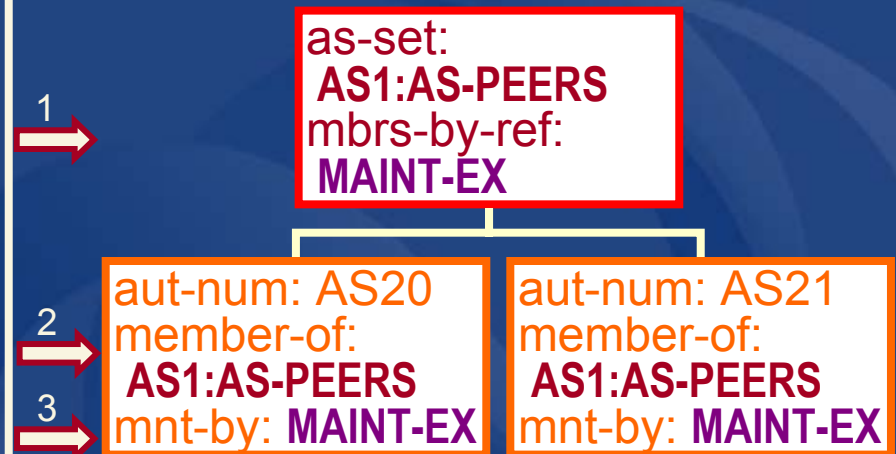
- members specified in the 'set-' object



- 'members' specifies members of the set
- Members added in the 'set-' object
- No need to modify the member object when adding members

mbrs-by-ref

- 'set' specified in the member objects



- 'mbrs-by-ref' specifies the maintainer of the members.
- Members reference the 'set-' object in the 'member-of' attribute
- Members are maintained by the maintainer specified in the 'set-'

Hierarchical authorisation

- **mnt-routes**
 - authenticates *creation* of route objects
 - creation of route objects must pass authentication of mntner referenced in the mnt-routes attribute

- Format:

- mnt-routes : <mntner>

In:

inetnum

,

aut-num

and

route

objects

Authorisation mechanism

```
inetnum:      202.137.181.0 - 202.137.185.255
netname:      SPARKYNET-WF
descr:        SparkyNet Service Provider
...
mnt-by:      MAINT-APNIC-AP
mnt-lower:   MAINT-SPARKYNET
mnt-routes:  MAINT-SPARKYNET-WF
```

This object can only be modified by APNIC

Creation of more specific objects (assignments) within this range has to pass the authentication of MAINT-SPARKYNET

Creation of route objects matching/within this range has to pass the authentication of MAINT-SPARKYNET-WF

Creating route objects

- Multiple authentication checks:
 - Originating ASN
 - mntner in the mnt-routes is checked
 - If no mnt-routes, mnt-lower is checked
 - If no mnt-lower, mnt-by is checked
 - AND the address space
 - Exact match & less specific route
 - mnt-routes etc
 - Exact match & less specific inetnum
 - mnt-routes etc
 - AND the route object mntner itself
 - The mntner in the mnt-by attribute



aut-num

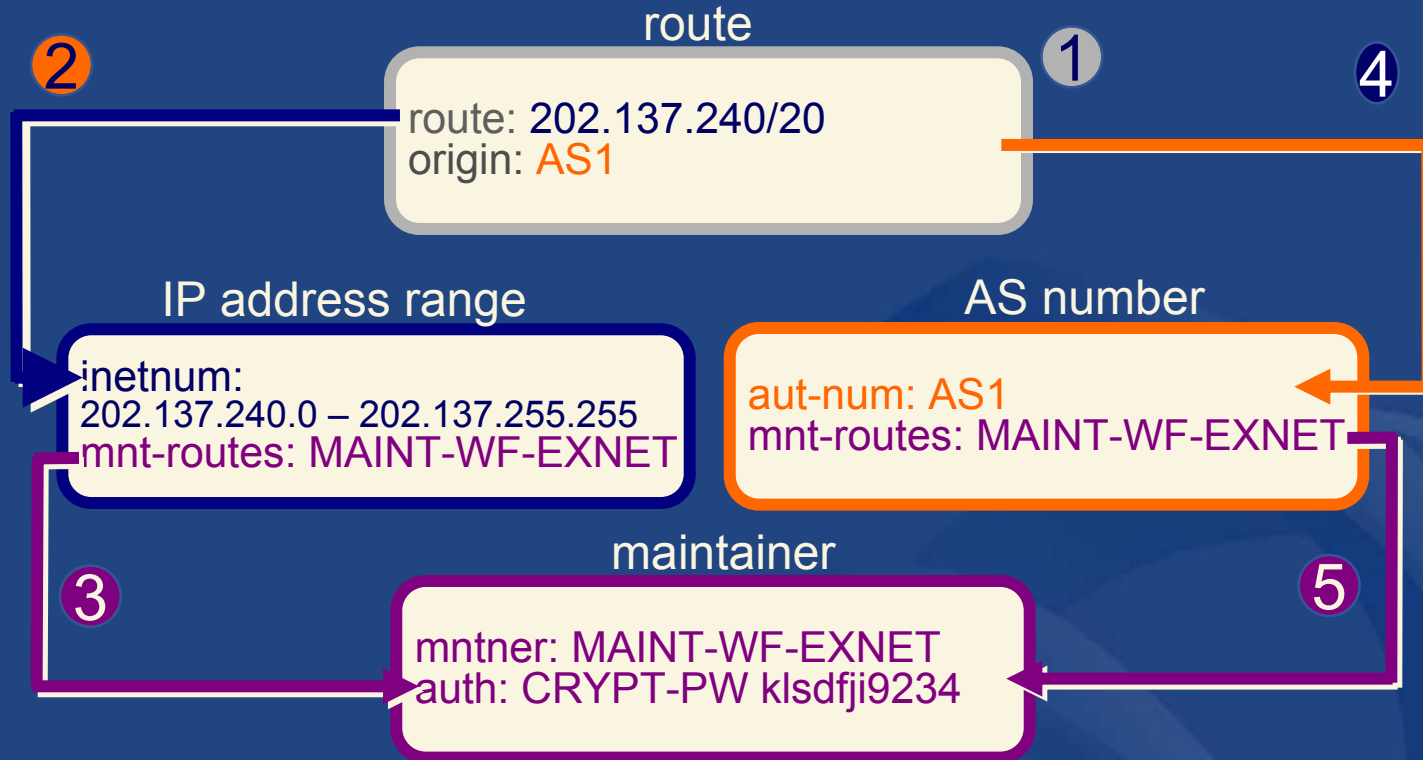
inetnum

route

(encompassing)

route

Creating route objects



1. Create route object and submit to APNIC RR database
2. Db checks inetnum obj matching/encompassing IP range in route obj
3. Route obj creation must pass auth of mntner specified in inetnum *mnt-routes* attribute.
4. Db checks aut-num obj corresponding to the ASN in route obj
5. Route obj creation must pass auth of mntner specified in aut-num *mnt-routes* attribute.

Useful IRR queries

- *What routes are originating from my AS?*
 - **whois -i origin <ASN>**
 - route objects with matching origin
- *What routers does my AS operate?*
 - **whois -i local-as <ASN>**
 - inet-rtr objects with a matching local-as
- *What objects are protecting “route space” with my maintainer?*
 - **whois -i mnt-routes <mntner>**
 - aut-num, inetnum & route objects with matching mnt-routes

(always specify host. e.g. ‘whois -h whois.apnic.net’)

Useful IRR queries (cont'd)

- *What '-set objects' are the objects protected by this maintainer a member of?*
 - **whois -i mbrs-by-ref <mntner>**
 - set objects (as-set, route-set and rtr-set) with matching mbrs-by-ref
- *What other objects are members of this '-set object'?*
 - **whois -i member-of <set name>**
 - Objects with a matching member-of
 - provided the membership claim is validated by the mbrs-by-ref of the set.



Recap attributes of some objects

Inetnum, aut-num and route object

Inetnum object

- Review of some attributes
 - inetnum:
 - Specifies a range of IPv4 that inetnum object represents
 - netname:
 - The name of a range of IP address space
 - status:
 - Specifies the status of the address range represented by inetnum object
 - mnt-by:
 - Specifies the identifier of a registered mntner object for authorisation of updating the object
 - mnt-lower:
 - Specifies the identifier of a registered mntner object to provide hierarchical authorisation

Inetnum object example

- Specifies IP allocations & assignments

```
inetnum:          202.36.0.0 - 202.37.255.255
netname:          NZGATE-NZ
descr:           NZ Gate National Service Provider
descr:           Administered by Telecom New Zealand Ltd
descr:           New Zealand
country:         NZ
admin-c:         DBK1-AP
tech-c:          KS61-AP
tech-c:          KS61-AP
remarks:         service provider
notify:          dbmon@apnic.net
mnt-by:          APNIC-HM
changed:         ARRON@WAIKATO.AC.NZ 19950612
changed:         hostmaster@apnic.net 20011004
changed:         hm-change@apnic.net 20020722
status:          ALLOCATED PORTABLE
source:          APNIC
```

Aut-num object

- Review of some attributes
 - aut-num:
 - ASN, an “AS” string followed by the number
 - member-of:
 - Identify as-set object
 - import:
 - Specifies an import policy expression
 - export:
 - Specifies an export policy expression
 - mnt-lower:
 - Specifies the identifier of a registered mntner object to provide hierarchical authorisation
 - mnt-routes:
 - Determines authorisation for the creation of route objects
 - mnt-by:
 - Specifies the identifier of a registered mntner object for authorisation of updating the object

Aut-num object: import attribute

- Import

```
from <peering-1> [action <action-1>]
```

```
....
```

```
from <peering-N> [action <action-N>]
```

```
accept <filter>
```

- <peering-x> can be ASN or as-set
- Set of routes matched by filter
 - Imported from all peers in peerings
- While importing routes at <peering-x>
 - <action-x> is done
- Example
 - med=0; community.append (3561:10); pref=30

Aut-num object: export attribute

- Export

```
to <peering-1> [action <action-1>]
```

```
....
```

```
to <peering-N> [action <action-N>]
```

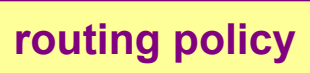
```
announce <filter>
```

- Set of routes matched by filter
 - Exported to all peers in peerings
- While exporting routes at <peering-x>
 - <action-x> is done
- Note: use semicolon (;) after each action specification (not mentioned in the RFC)

Aut-num object example

– Describes an Autonomous System

```
aut-num: AS17914
as-name: ASN-2DAY-NZ-AP
descr: 2Day Internet Limited
country: NZ
import: from AS17914:AS-TRANSIT action pref=100; accept ANY
import: from AS17914:AS-PEERS action pref=120; accept PeerAS
export: to AS17914:AS-TRANSIT announce AS17914:AS-CUSTOMERS
export: to AS17914:AS-PEERS announce AS17914:AS-CUSTOMERS
admin-c: PM5-NZ
tech-c: JA39
remarks: 2day.com peers at the Auckland Peering Exchange
mnt-by: MAINT-2DAY-NZ
changed: jabley@automagic.org 20021104
source: APNIC
```



Route object

- Review of some attributes
 - route
 - The address prefix of the route.
 - origin
 - Specifies the AS that originates the route.
 - member-of
 - Identifies a set object that his object wants to be a memembr of.
 - mnt-by
 - Specifies a registered mntner object used for authorisaton
 - mnt-lower
 - Specifies a registered mntner object used for hierarchical authorisation.
 - mnt-routes
 - References a mntner object which is used in determining authorisation for the creation of route objects.



Route object example

- Each interAS route originated by an autonomous system

```
route:          202.37.240.0/23
descr:         route originating from 2day.com
origin:        AS17914
mnt-by:        MAINT-2DAY-NZ
changed:       jabley@automagic.org 20021220
source:        APNIC
notify:        noc@2day.com
```



APNIC

Asia Pacific Network Information Centre

Routing Policy

What is a Routing Policy?

- Exchange of routing information between Autonomous Systems



- Usually policies are not configured for each network separately
 - Configured for groups of networks



Why define a Routing Policy?

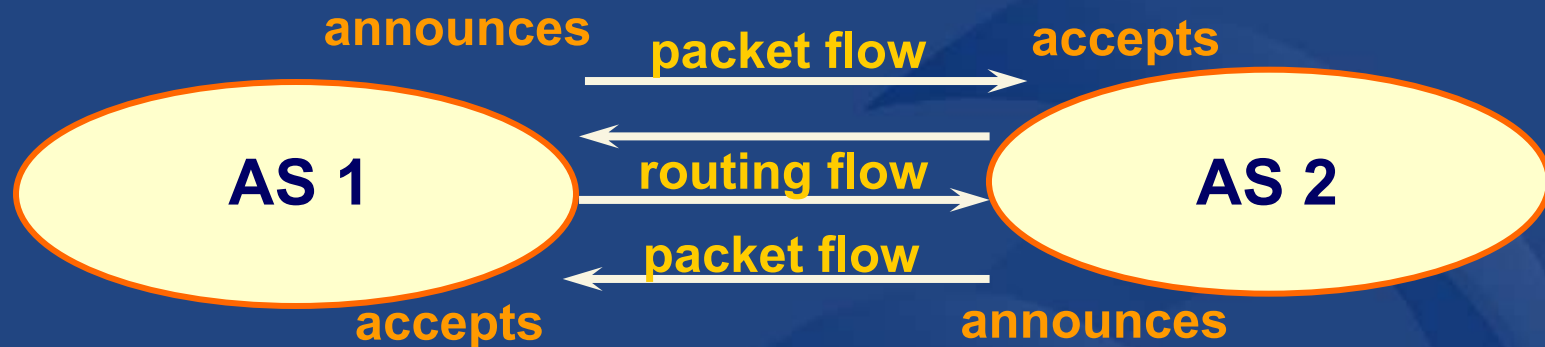
- Documentation
- Consistency across your AS
 - routers / implementations
- Scalability
- Provides routing security
 - Can peer originate the route?
 - Can peer act as transit for the route?

How define a Routing Policy?

- Who are my BGP neighbours?
 - (customers/ peers/ upstreams)
- What routes are:
 - Originated by each neighbour?
 - Imported from each neighbour?
 - Exported to each neighbour?
 - Preferred when multiple routes exist?
 - How are they treated (modified routing parameters?)

Defining the Routing Policy

- Routing and packet flows

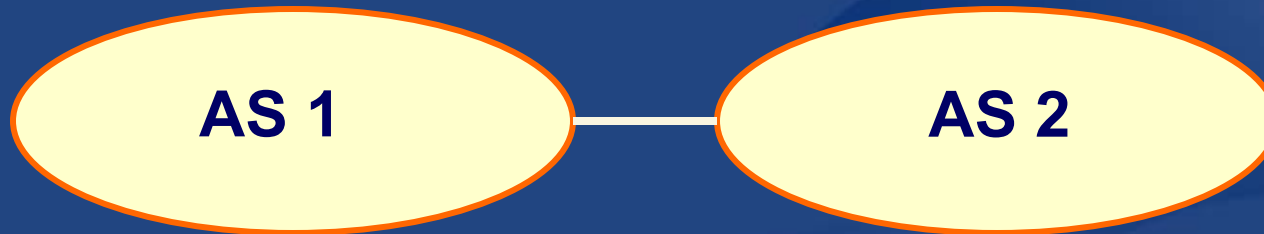


For AS1 and AS2 networks to communicate

- AS1 must announce to AS2
- AS2 must accept from AS1
- AS2 must announce to AS1
- AS1 must accept from AS2

Defining the Routing Policy

Basic concept



*“action pref” - the lower the value,
the preferred the route*

aut-num: AS1

...

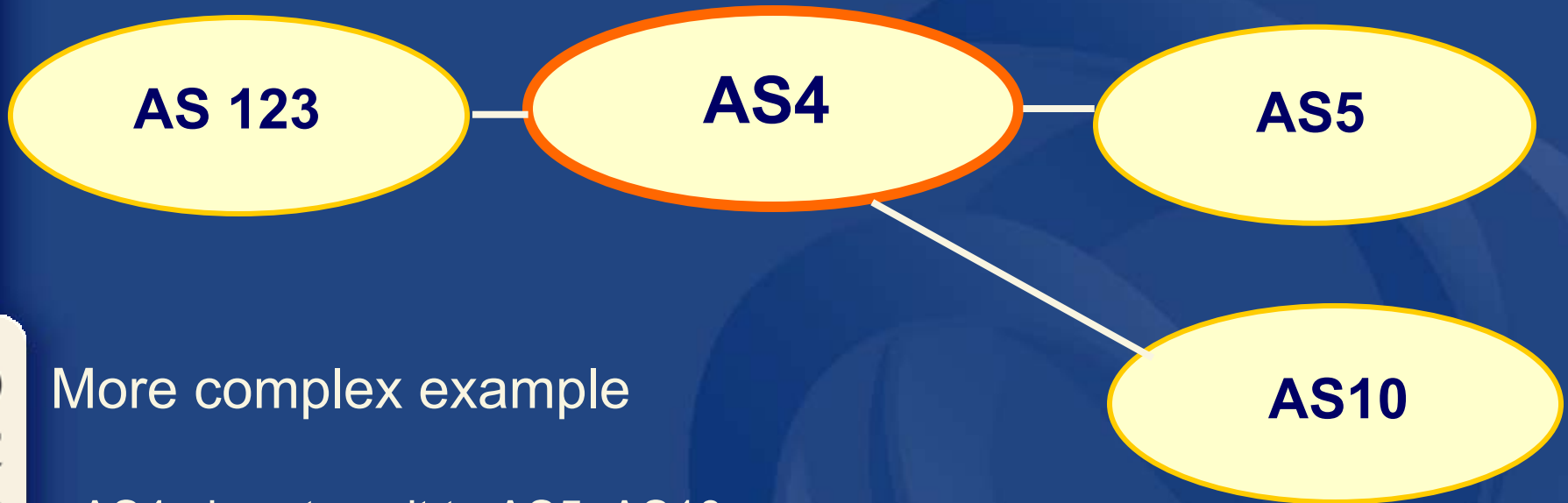
**import: from AS2
action pref=100;
accept AS2
export: to AS2 announce AS1**

aut-num: AS2

...

**import: from AS1
action pref=100;
accept AS1
export: to AS1 announce AS2**

Defining the Routing Policy



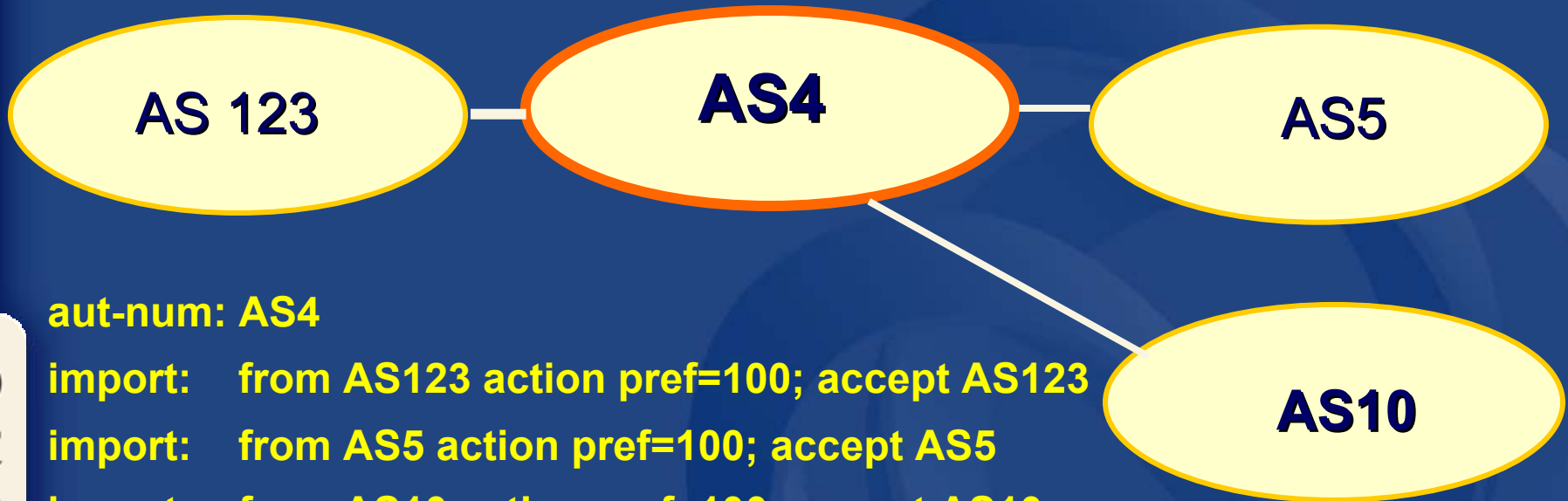
More complex example

- AS4 gives transit to AS5, AS10
- AS4 gives local routes to AS123

Defining the Routing Policy

Let's express import and export attributes for AS4!

- AS4 gives transit to AS5, AS10
- AS4 gives local routes to AS123



aut-num: AS4

import: from AS123 action pref=100; accept AS123

import: from AS5 action pref=100; accept AS5

import: from AS10 action pref=100; accept AS10

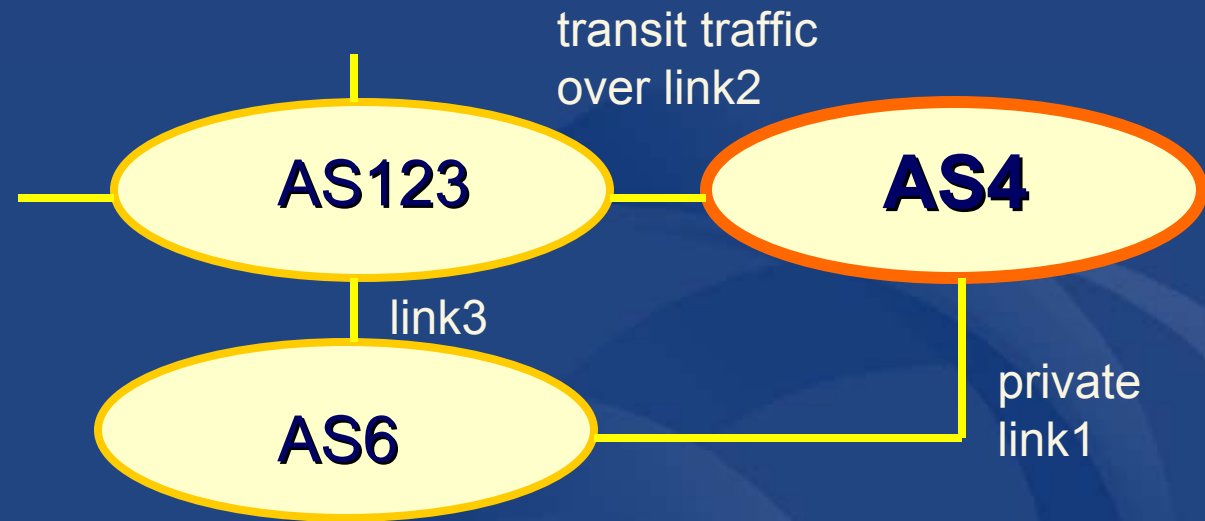
export: to AS123 announce AS4

export: to AS5 announce AS4 AS10

export: to AS10 announce AS4 AS5

Not a path

Defining the Routing Policy

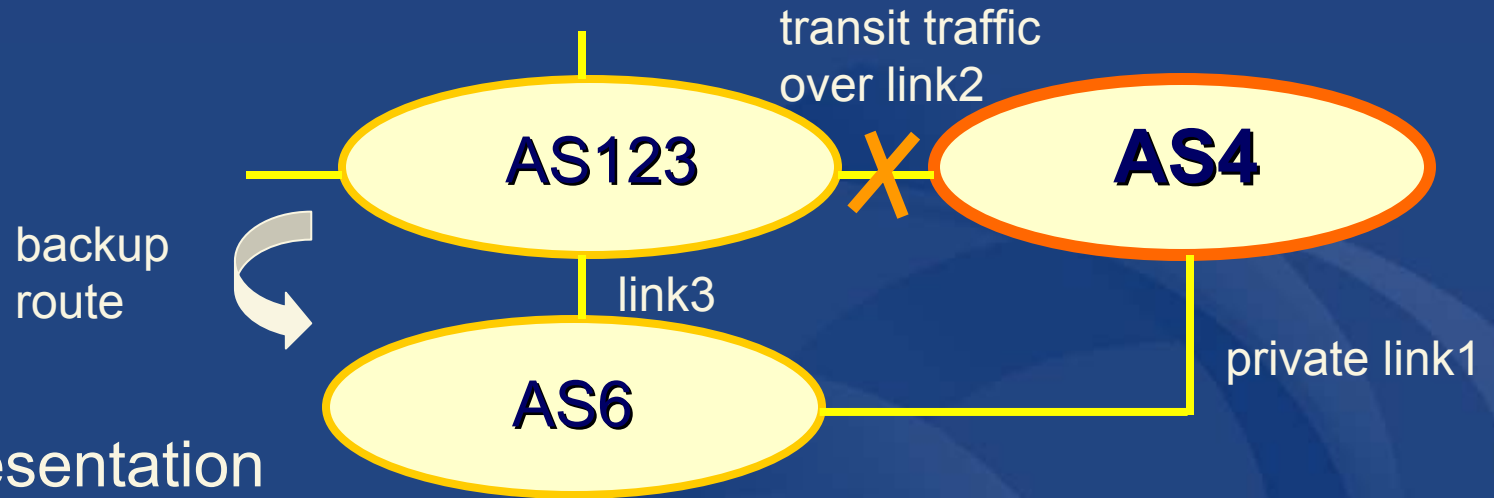


More complex example

- AS4 and AS6 private link1
- AS4 and AS123 main transit link2
- backup all traffic over link1 and link3 in event of link2 failure

Defining the Routing Policy

Let's express import and export attributes for AS4!



aut-num: AS4

import: from AS123 action pref=100; accept ANY ← full routing received

import: from AS6 action pref=50; accept AS6

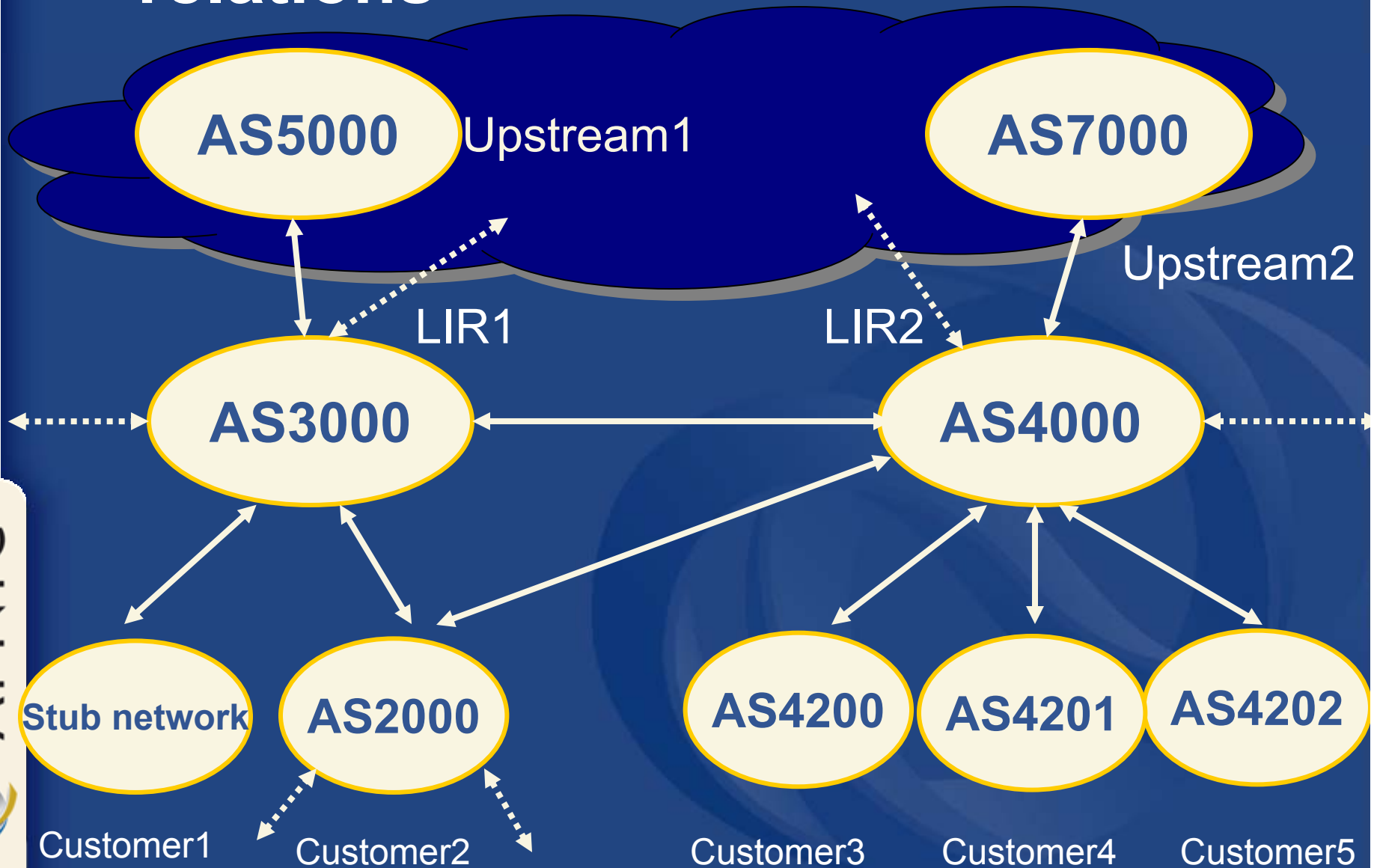
import: from AS6 action pref=200; accept ANY

export: to AS6 announce AS4

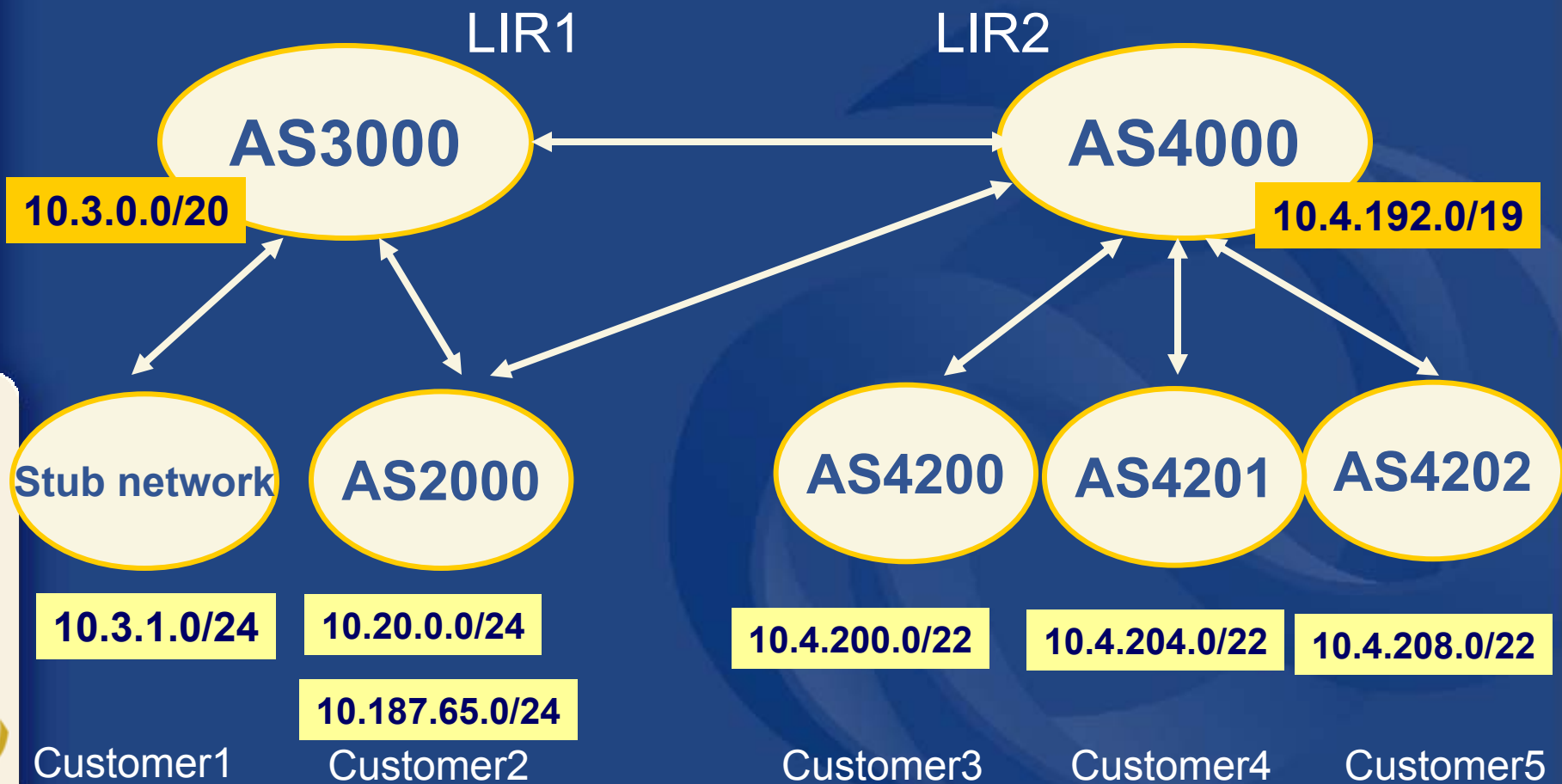
export: to AS123 announce AS4

higher cost for backup route

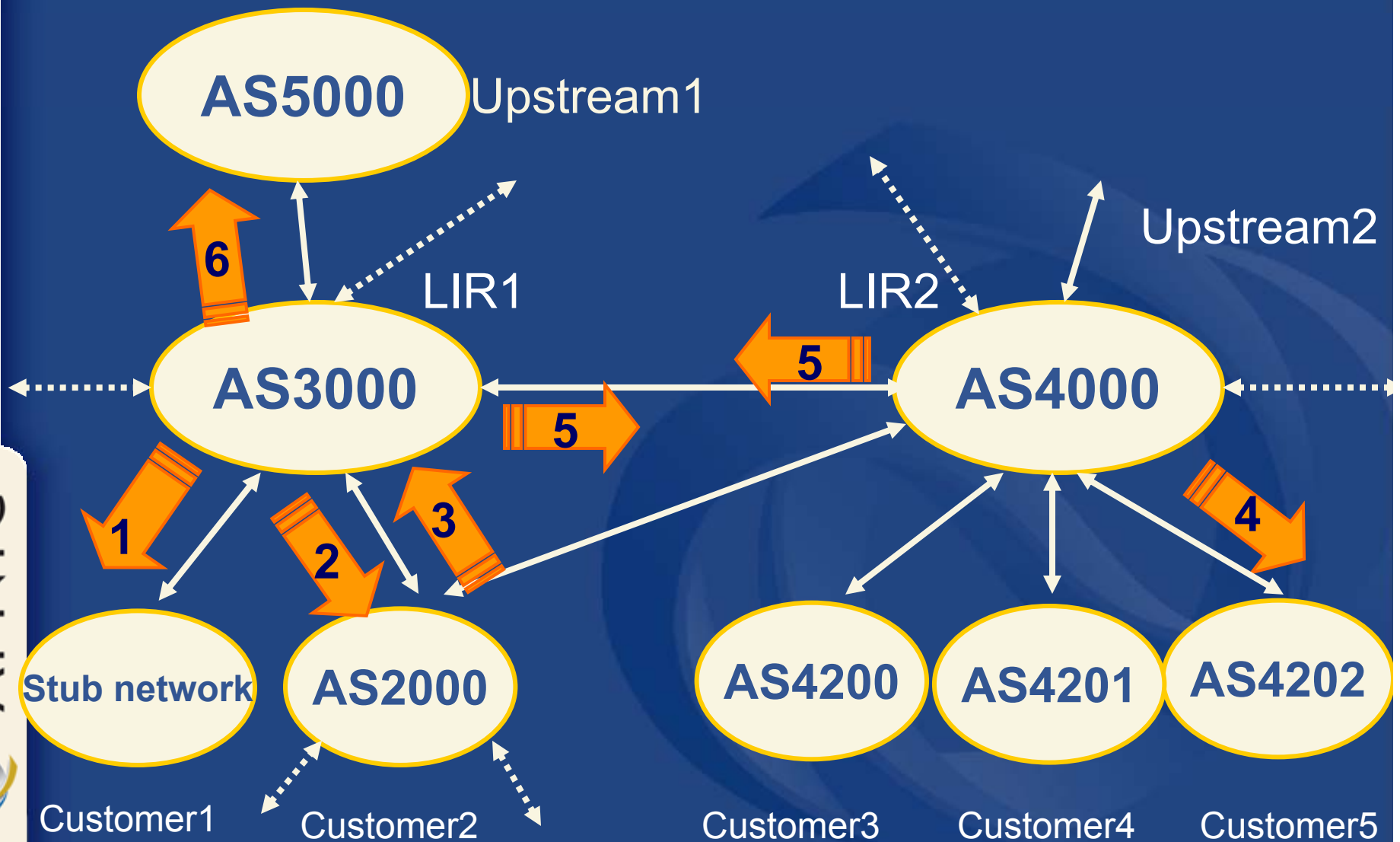
Experimental setup: AS relations



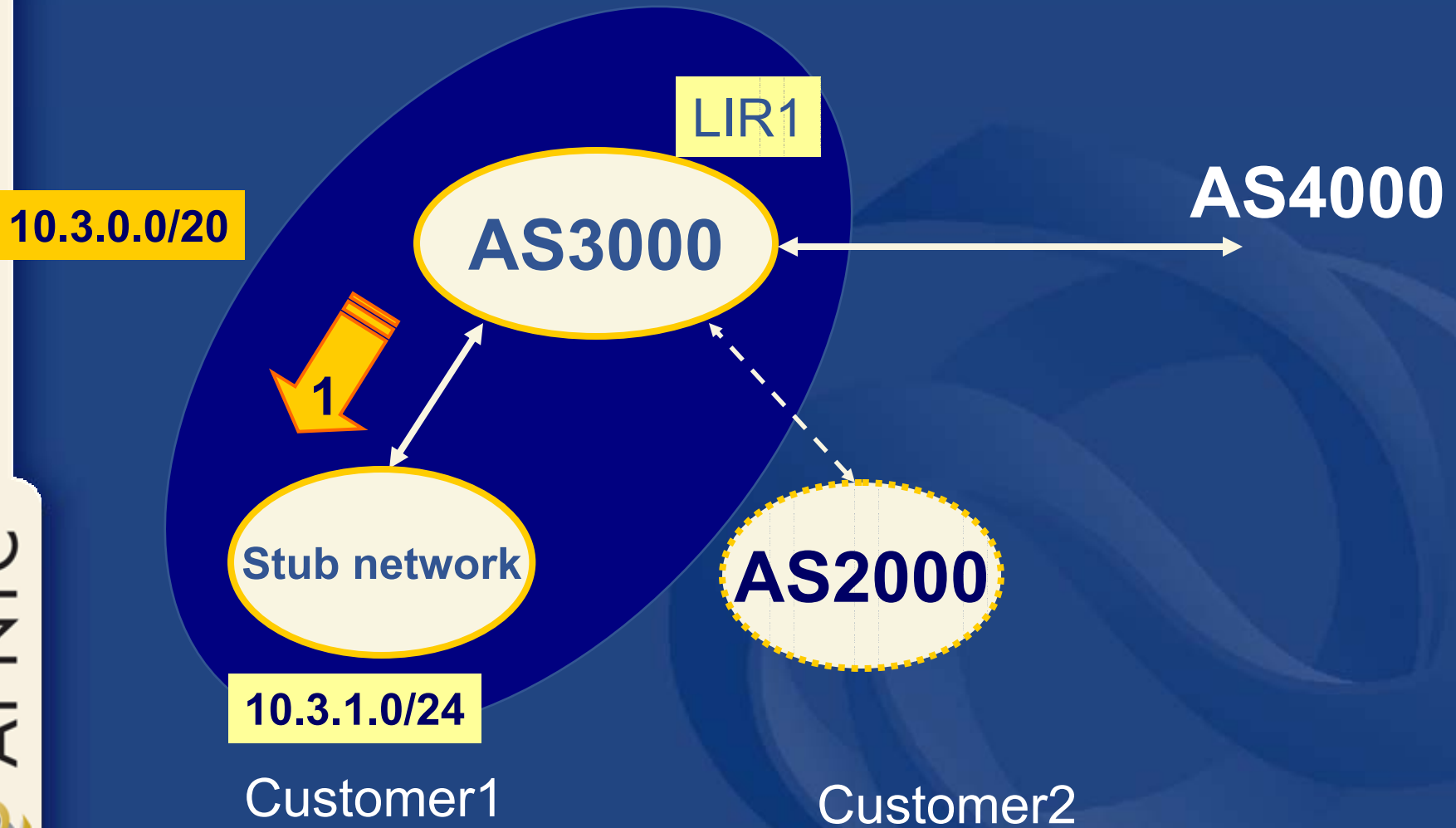
AS relations, including allocations & assignments



Case studies, overview

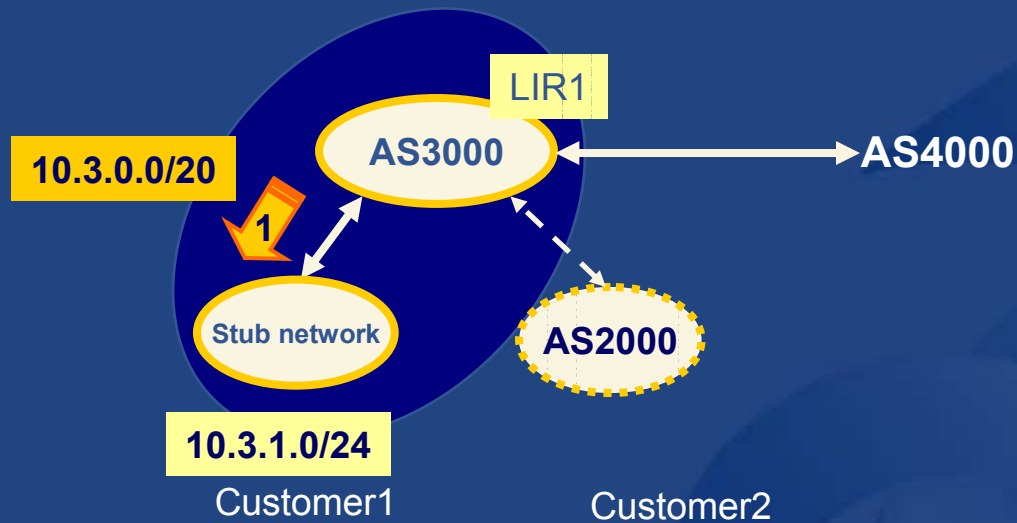


Case1: Static end-user set-up



Exercise 1: Static end-user set-up

Express import and export attributes for AS3000



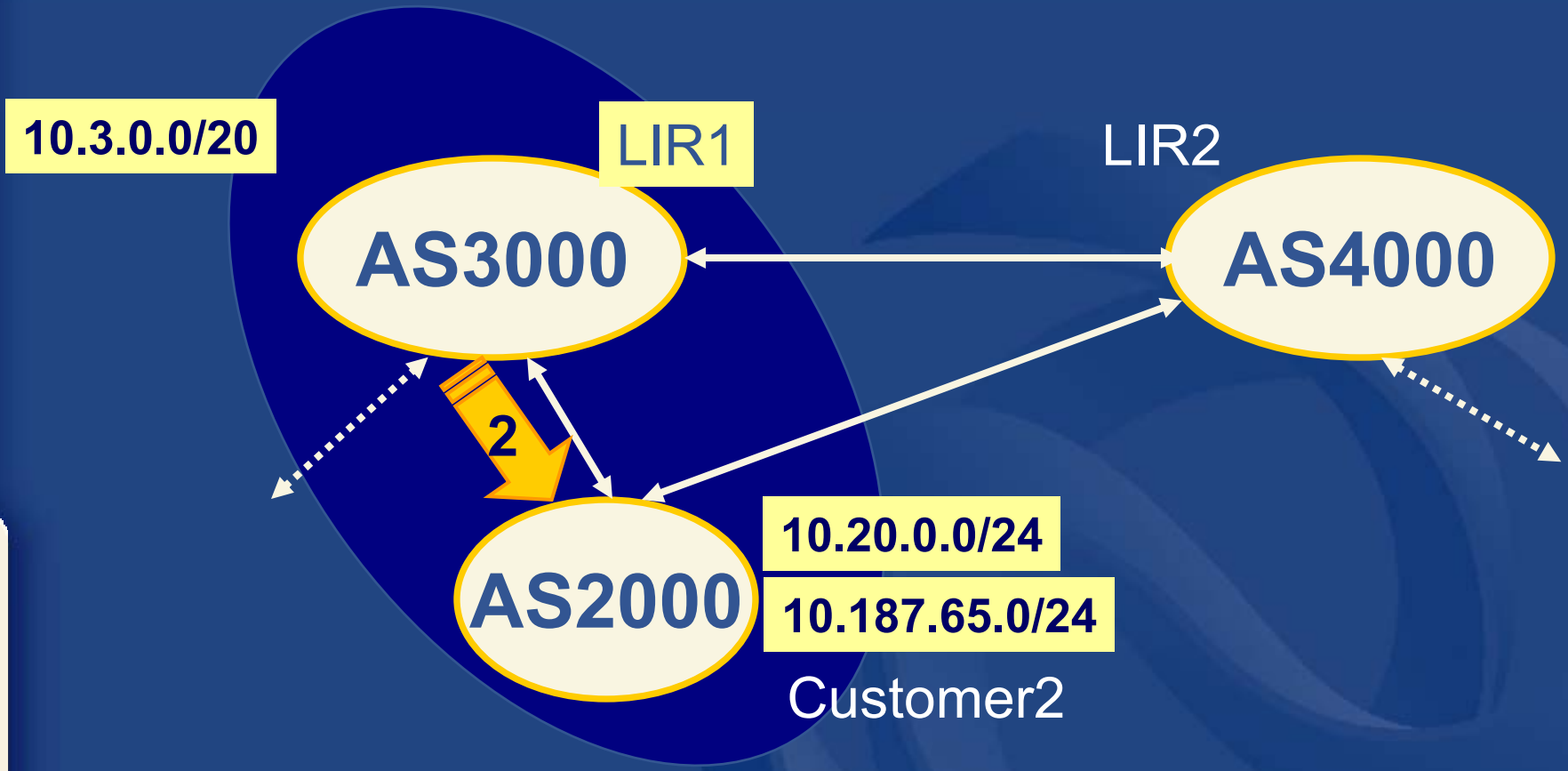
aut-num: AS3000

import: protocol STATIC into BGP4 from AS3000 accept {10.3.1.0/24}

export: to AS4000 announce AS3000

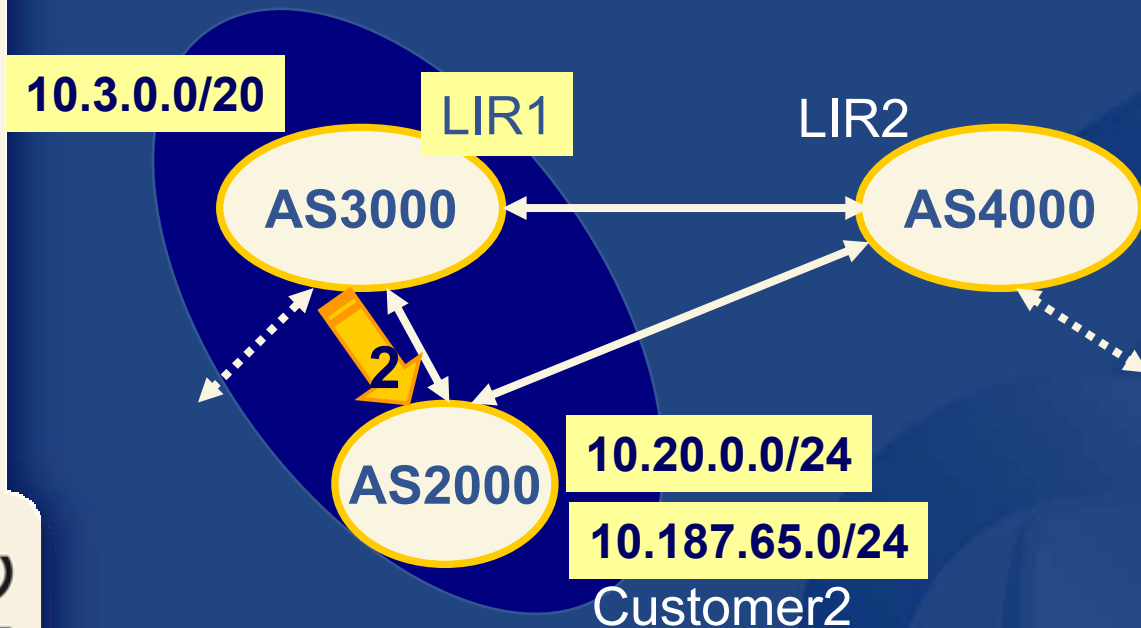
[...]

Case 2: Multi-homed customer - provider set-up



Exercise 2: Multi-homed customer - provider set-up

Express import and export attributes for AS3000



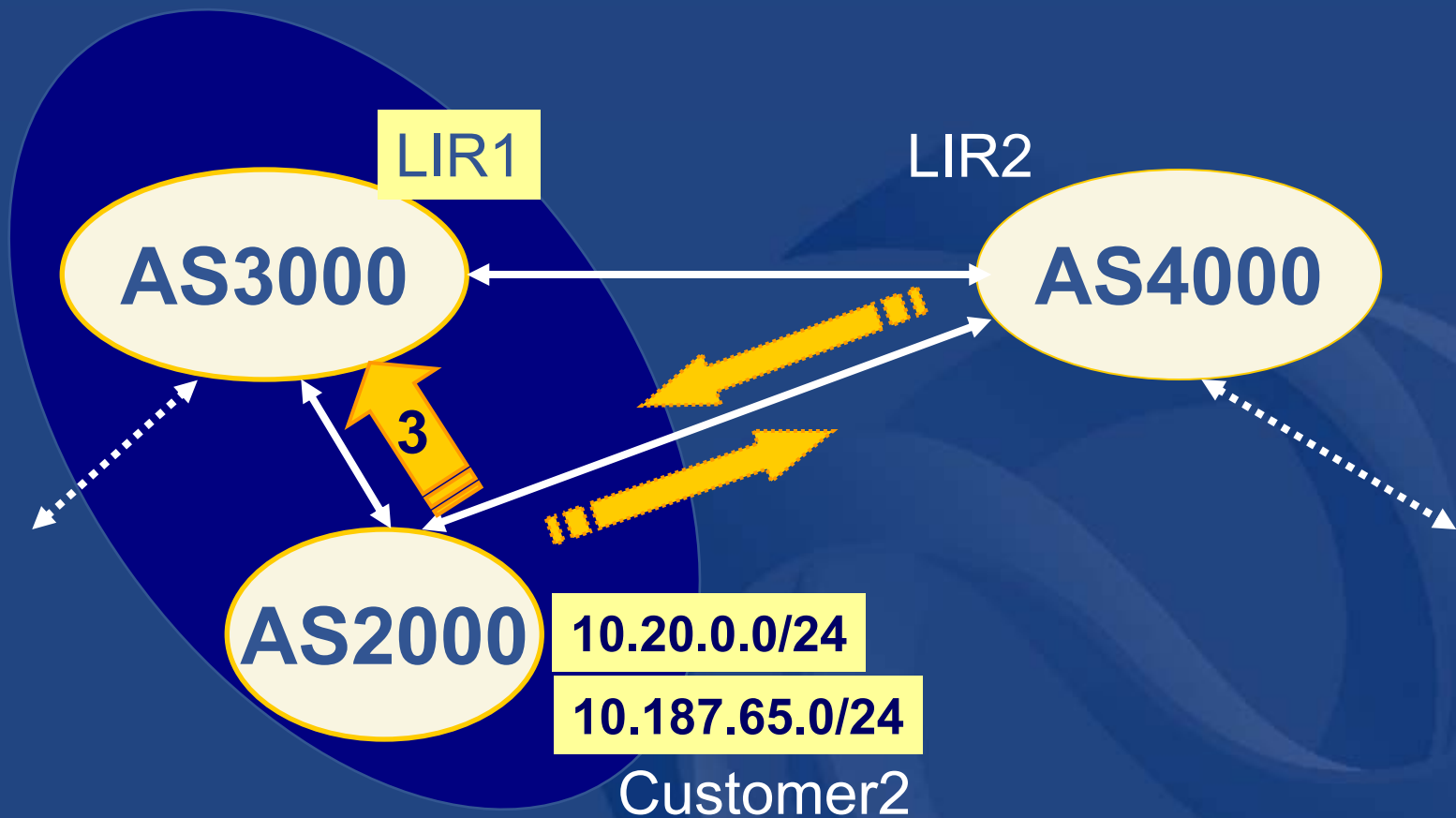
```
aut-num: AS3000
import: from AS2000 accept AS2000
export: to AS2000 announce any
[...]
```

Review Case 2: BGP customers, - provider aut-num

```
aut-num: AS3000
import: from AS2000 accept AS2000
export: to AS2000 announce ANY
[...]
```

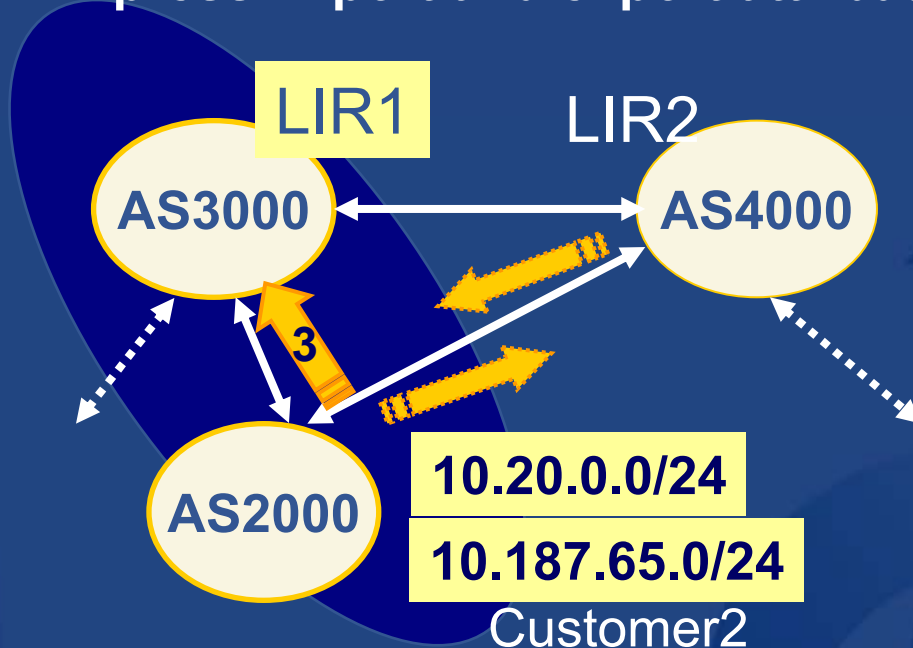
- The simplest policy is strict customer/provider relationship
 - Customer sends its routes to provider
 - Customer accepts everything the provider sends

Case 3: Multi-homed customer - customer set-up



Exercise 3-1: Not- Full Multi-homed customer - customer set-up

Express import and export attributes for AS2000



```
aut-num: AS2000
import: from AS3000 accept ANY
export: to AS3000 announce AS2000
import: from AS4000 accept AS4000
export: to AS4000 announce AS2000
```

[...]

Review Case 3.1: Not Full Multihoming - customer aut-num

- DB objects:

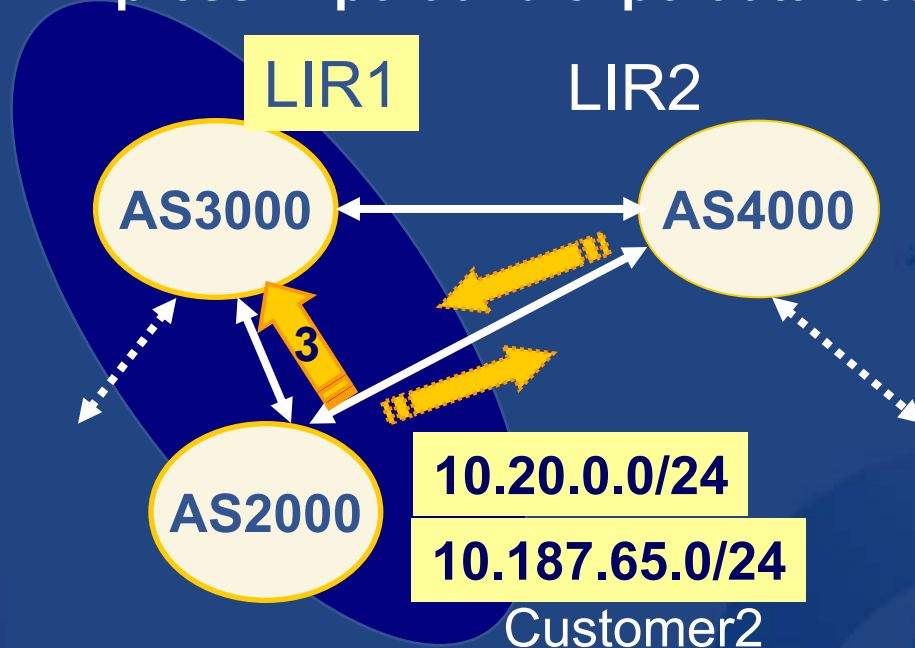
```
aut-num: AS2000
import:  from AS3000 accept ANY
export:  to AS3000 announce AS2000
import:  from AS4000 accept AS4000
export:  to AS4000 announce AS2000
[...]
```

```
route: 10.20.0.0/24
origin: AS2000
[...]
```

```
route: 10.187.65.0/24
origin: AS2000
[...]
```

Exercise 3-2: Full Multi-homed customer - customer set-up

Express import and export attributes for AS2000



```

aut-num: AS2000
import: from AS3000 action pref=50; accept ANY
export: to AS3000 announce AS2000
import: from AS4000 action pref=100; accept AS4000
export: to AS4000 announce AS2000
[...]
```

Review Case 3.2: Full Multihoming - customer aut-num

- Introducing policy, setting the “pref” value
 - lower the “pref”, the preferred the route

```
aut-num: AS2000
import:      from AS3000 action pref=50; accept ANY
export:      to AS3000 announce AS2000
import:      from AS4000 action pref=100; accept ANY
export:      to AS4000 announce AS2000
```



APNIC

Asia Pacific Network Information Centre

Using the Routing Registry

Routing policy, the IRRToolSet &
APNIC RR Benefits

IRRToolSet

- Set of tools developed for using the Internet Routing Registry
 - Started as RAToolSet
- Now maintained by RIPE NCC:
 - <http://www.ripe.net/db/irrtoolset/>
 - Download:
<ftp://ftp.ripe.net/tools/IRRToolSet/>
 - Installation needs: lex, yacc and C++ compiler

Use of RPSL - RtConfig

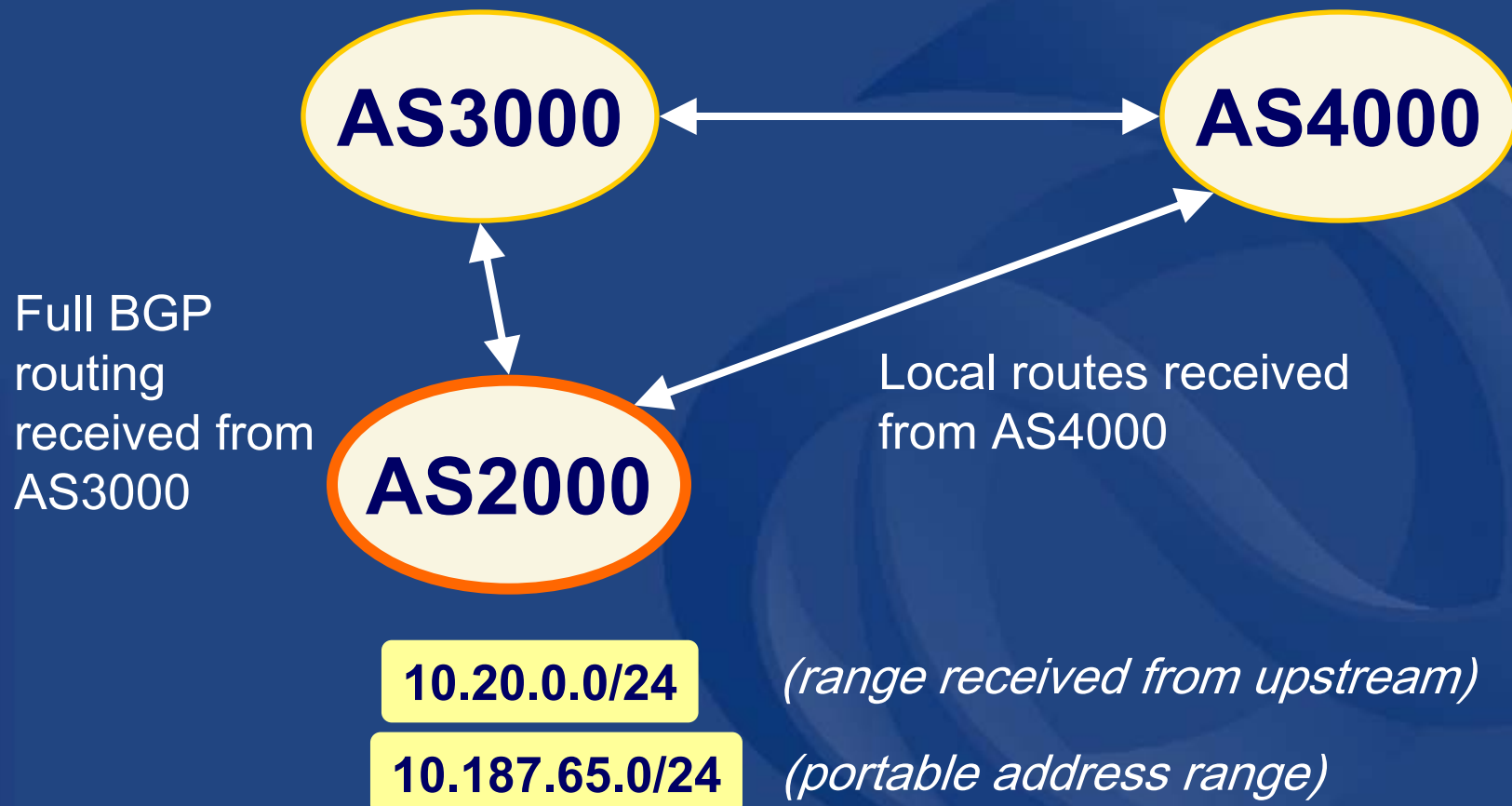
- RtConfig v4
 - part of IRRToolSet
- Reads policy from IRR (aut-num, route & -set objects) and generates router configuration
 - vendor specific:
 - Cisco, Bay's BCC, Juniper's Junos and Gated/RSd
 - Creates route-map and AS path filters
 - Can also create ingress / egress filters
 - (documentation says Cisco only)

Why use IRR and RtConfig?

- Benefits of RtConfig
 - Avoid filter errors (typos)
 - Expertise encoded in the tools that generate the policy rather than engineer configuring peering session
 - Filters consistent with documented policy
 - (need to get policy correct though)
 - Engineers don't need to understand filter rules
 - it just works :-)

Using RtConfig - Case scenario

Not fully multi-homing



Using RtConfig – IRR objects

```
aut-num: AS2000
import:  from AS3000 accept ANY
export:  to AS3000 announce AS2000
import:  from AS4000 accept AS4000
export:  to AS4000 announce AS2000
[...]
```

← *full BPG routing*

← *local routes*

```
route: 10.20.0.0/24
origin: AS2000
[...]
```

```
route: 10.187.65.0/24
origin: AS2000
[...]
```

RtConfig commands

```
@RtConfig set cisco_map_name = "AS%d-IMPORT"
@RtConfig import AS2000_10.20.0.3 AS3000 10.3.15.2
!
@RtConfig set cisco_map_name = "AS%d-IMPORT"
@RtConfig import AS2000_10.20.0.4 AS4000 10.4.192.2
!
```


RtConfig output (import)

```
no route-map AS3000-IMPORT
!
route-map AS3000-IMPORT permit 10
!
router bgp 2000
neighbor 10.0.1.3 route-map AS3000-IMPORT in
!
!
no ip prefix-list p1134
ip prefix-list p1134 permit 10.4.192.0/19
ip prefix-list p1134 deny 0.0.0.0/0 le 32
!
no route-map AS4000-IMPORT
!
route-map AS4000-IMPORT permit 10
  match ip address prefix-list p1134
exit
!
router bgp 2000
neighbor 10.0.1.4 route-map AS4000-IMPORT in
```

RtConfig – web prototype

RtConfig Test Page - Microsoft Internet Explorer

Address <http://www.ripe.net/cgi-bin/RtConfig.cgi> Go

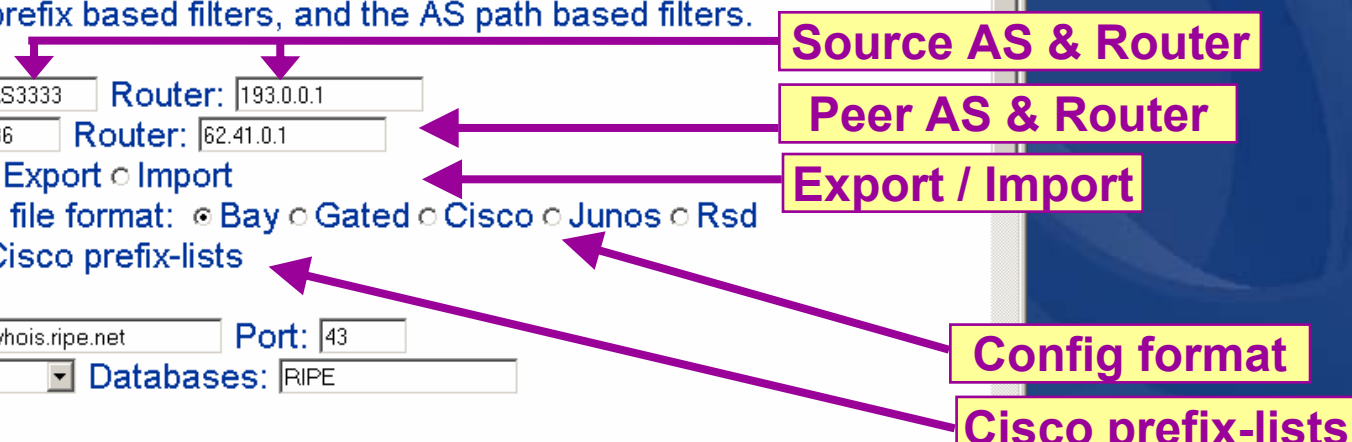
 [homepage](#) | [what's new](#) | [whois db](#) | [search](#) | [site map](#) | [f.a.q.](#)

RtConfig

RtConfig generates router configurations from policies specified in RPSL or RIPE-181. Several vendor and public domain router languages are supported, including Cisco routers, Gated, Rsd and Nortel Bay routers. It can generate configurations that may use both the address prefix based filters, and the AS path based filters.

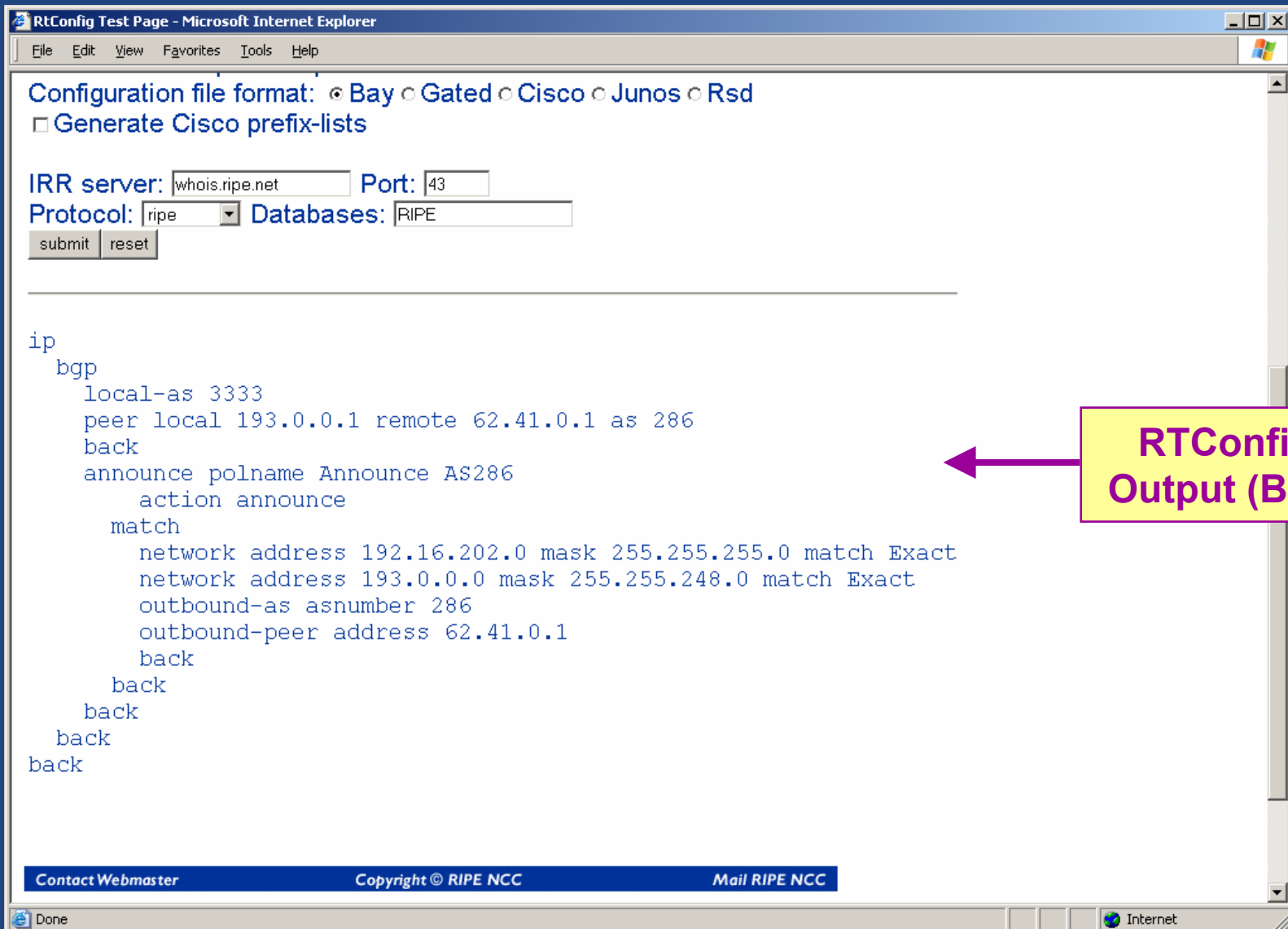
Source AS: Router:
Peer AS: Router:
Command: Export Import
Configuration file format: Bay Gated Cisco Junos Rsd
 Generate Cisco prefix-lists

IRR server: Port:
Protocol: Databases:



<http://www.ripe.net/cgi-bin/RtConfig.cgi>

RtConfig – web output



The screenshot shows the RtConfig web interface in Microsoft Internet Explorer. The configuration file format is set to Bay. The IRR server is whois.ripe.net, the port is 43, the protocol is ripe, and the database is RIPE. The configuration output is as follows:

```
ip
  bgp
    local-as 3333
    peer local 193.0.0.1 remote 62.41.0.1 as 286
    back
    announce polname Announce AS286
      action announce
      match
        network address 192.16.202.0 mask 255.255.255.0 match Exact
        network address 193.0.0.0 mask 255.255.248.0 match Exact
      outbound-as asnumber 286
      outbound-peer address 62.41.0.1
    back
  back
back
back
back
back
```

A yellow box with a purple arrow points to the configuration output, labeled "RTConfig Output (Bay)".

Contact Webmaster Copyright © RIPE NCC Mail RIPE NCC

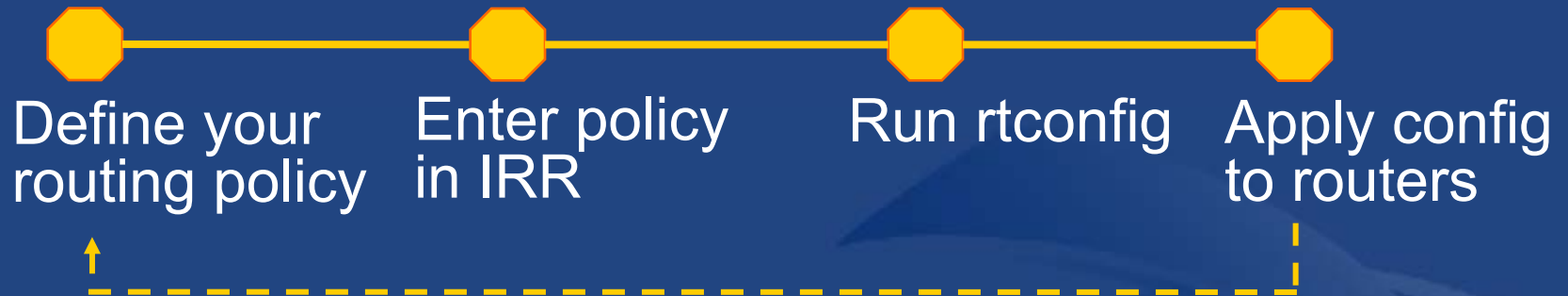
The rest of the IRRToolSet

- peval
 - (Lightweight) policy evaluation tool
- prtraceroute
 - Prints the route packets take - including policy information (as registered in RR)
- aoe (aut-num object editor)
 - Displays the aut-num object for the specified AS
- roe
 - Creates the “route” object (based on BGP dump and routes in aut-num objects)

The rest of the IRRToolSet

- `prpath`
 - enumerates possible paths between two ASes
- `CIDRAdvisor`
 - suggests safe aggregates per AS
- *rpslcheck*
 - syntax checks objects for IRR

Using the Routing Registry



Disadvantages

- Requires some initial planning
- Takes some time to define & register policy
- Need to maintain data in RR

Advantages

- You have a clear idea of your routing policy
- Consistent config over the whole network
- Less manual maintenance in the long run

Benefits of APNIC RR

- Single maintainer
 - Use same mntner to manage
 - internet resources
 - reverse DNS
 - routing policy
 - contact info
 - etc

(Single person object can also be used)

mntner:
MAINT-EX
...

person:
...
mnt-by: MAINT-EX

aut-num:
...
mnt-by: MAINT-EX

inetnum:
...
mnt-by: MAINT-EX

domain:
...
mnt-by: MAINT-EX

route:
...
mnt-by: MAINT-EX

Benefits of APNIC RR

- APNIC able to assert resources for a registered route within APNIC ranges.

```
inetnum:      221.0.0.0 - 221.3.127.255
netname:      CNCGROUP-SD
descr:        CNCGROUP Shandong province network
country:      CN
admin-c:      CH455-AP
tech-c:       XZ14-AP
mnt-by:       APNIC-HM
mnt-lower:    MAINT-CNCGROUP-SD
changed:      hm-chnaged@apnic.net 20021224
status:       ALLOCATED PORTABLE
source:       APNIC
```

Allocation objects
maintained by APNIC

```
mntner:       APNIC-HM
descr:        APNIC Hostmaster - Maintainer
...
```

APNIC RR service scope

- Routing Queries
 - Regular whois clients
 - APNIC whois web interface
 - Special purpose programs such as IRRToolSet
 - <ftp://ftp.ripe.net/tools/IRRToolSet>
- Routing Registration and Maintenance
 - Similar to registration of Internet resources

APNIC RR service scope

- Support
 - APNIC Helpdesk support
- `<helpdesk@apnic.net>`
- Training
 - IRR workshop under development
 - Mirroring
 - APNIC mirrors IRRs within Asia Pacific and major IRRs outside of the region.

Summary

- APNIC RR integrated in APNIC Whois DB
 - whois.apnic.net
 - <auto-dbm@apnic.net>
- IRR benefits
 - Facilitates network troubleshooting
 - Generation of router configuration
 - Provides global view of routing
- APNIC RR benefits
 - Single maintainer (& person obj) for all objects
 - APNIC asserts resources for a registered route
 - Part of the APNIC member service!



APNIC

Asia Pacific Network Information Centre

Questions ?



APNIC

Asia Pacific Network Information Centre

Practical Usage of the RR



Potential Practical Problems

- Policy can easily get very complex and result in even more complex router configuration
- Line limit on cisco AS path filters
 - need to be careful when using as-set
- Nervous about configuring routers from public data?
 - Compare this with anti-virus SW updates!

Next steps

- Tasks for your own AS:
 - Create person and maintainer objects
 - Set up PGP authentication
 - Create aut-num objects for each AS
 - Identify IP prefixes associated with each AS
 - Create route objects in the database
 - Create as-set objects where policy is common
 - Either in the APNIC RR
 - Or in your own routing registry database

References

- RFC 2622 *“Routing Policy Specification Language (RPSL)”*
- RFC 2650 *“Using RPSL in Practice”*
- RFC 2725 *“Routing Policy System Security”*
- APNIC Routing Registry Guide
 - <http://www.apnic.net/services/apnic-rr-guide.html>
- IRRToolSet
 - <http://www.ripe.net/ripencc/pub-services/db/irrtolset/index.html>



APNIC

Asia Pacific Network Information Centre

Questions?

Summary

- The Internet Routing Registry
- APNIC Database
 - RPSL
 - Queries and updates
 - Authentication
- Routing Policy
 - Case studies
- Routing Registry Benefits



Appendix

Object Templates in RPSL

Mntner object template

<code>mntner:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[primary/look-up key]</code>
<code>descr:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>admin-c:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>tech-c:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>upd-to:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>mnt-nfy:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>auth:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>remarks:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[]</code>
<code>notify:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>mnt-by:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>auth-override:</code>	<code>[optional]</code>	<code>[single]</code>	<code>[]</code>
<code>referral-by:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[inverse key]</code>
<code>changed:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>source:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[]</code>

Inetnum object template

inetnum:	[mandatory]	[single]	[primary/look-up key]
netname:	[mandatory]	[single]	[lookup key]
descr:	[mandatory]	[multiple]	[]
country:	[mandatory]	[multiple]	[]
admin-c:	[mandatory]	[multiple]	[inverse key]
tech-c:	[mandatory]	[multiple]	[inverse key]
rev-srv:	[optional]	[multiple]	[inverse key]
status:	[generated]	[single]	[]
remarks:	[optional]	[multiple]	[]
notify:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
mnt-lower:	[optional]	[multiple]	[inverse key]
mnt-routes:	[optional]	[single]	[inverse key]
changed:	[mandatory]	[multiple]	[]
source:	[mandatory]	[single]	[]

Route object template

route:	[mandatory]	[single]	[primary/look-up key]
descr:	[mandatory]	[multiple]	[]
country:	[optional]	[single]	[]
origin:	[mandatory]	[single]	[primary/inverse key]
holes:	[optional]	[multiple]	[]
member-of:	[optional]	[multiple]	[]
inject:	[optional]	[multiple]	[]
aggr-mtd:	[optional]	[single]	[]
aggr-bndry:	[optional]	[single]	[]
export-comps:	[optional]	[single]	[]
components:	[optional]	[single]	[]
remarks:	[optional]	[multiple]	[]
cross-mnt:	[optional]	[multiple]	[inverse key]
cross-nfy:	[optional]	[multiple]	[inverse key]
notify:	[optional]	[multiple]	[inverse key]
mnt-lower:	[optional]	[multiple]	[inverse key]
mnt-routes:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
changed:	[mandatory]	[multiple]	[]
source:	[mandatory]	[single]	[]

Aut-num object template

aut-num:	[mandatory]	[single]	[primary/look-up key]
as-name:	[mandatory]	[single]	[]
descr:	[mandatory]	[multiple]	[]
country:	[optional]	[single]	[]
member-of:	[optional]	[multiple]	[]
import:	[optional]	[multiple]	[]
export:	[optional]	[multiple]	[]
default:	[optional]	[multiple]	[]
remarks:	[optional]	[multiple]	[]
admin-c:	[mandatory]	[multiple]	[inverse key]
tech-c:	[mandatory]	[multiple]	[inverse key]
cross-mnt:	[optional]	[multiple]	[inverse key]
cross-nfy:	[optional]	[multiple]	[inverse key]
notify:	[optional]	[multiple]	[inverse key]
mnt-lower:	[optional]	[multiple]	[inverse key]
mnt-routes:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
changed:	[mandatory]	[multiple]	[]
source:	[mandatory]	[single]	[]



As-set object template

as-set:	[mandatory]	[single]	[primary/look-up key]
descr:	[mandatory]	[multiple]	[]
country:	[optional]	[single]	[]
members:	[optional]	[multiple]	[]
mbrs-by-ref:	[optional]	[multiple]	[inverse key]
remarks:	[optional]	[multiple]	[]
tech-c:	[mandatory]	[multiple]	[inverse key]
admin-c:	[mandatory]	[multiple]	[inverse key]
notify:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
changed:	[mandatory]	[multiple]	[]
source:	[mandatory]	[single]	[]



Route-set object template

route-set:	[mandatory]	[single]	[primary/look-up key]
descr:	[mandatory]	[multiple]	[]
members:	[optional]	[multiple]	[]
mbrs-by-ref:	[optional]	[multiple]	[inverse key]
remarks:	[optional]	[multiple]	[]
tech-c:	[mandatory]	[multiple]	[inverse key]
admin-c:	[mandatory]	[multiple]	[inverse key]
notify:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
changed:	[mandatory]	[multiple]	[]
source:	[mandatory]	[single]	[]



Inet-rtr object template

<code>inet-rtr:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[primary/look-up key]</code>
<code>descr:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>alias:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[]</code>
<code>local-as:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[inverse key]</code>
<code>ifaddr:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[lookup key]</code>
<code>peer:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[]</code>
<code>member-of:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>remarks:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[]</code>
<code>admin-c:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>tech-c:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>notify:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>mnt-by:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>changed:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>source:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[]</code>



Peering-set object template

peering-set:	[mandatory]	[single]	[primary/look-up key]
descr:	[mandatory]	[multiple]	[]
peering:	[mandatory]	[multiple]	[]
remarks:	[optional]	[multiple]	[]
tech-c:	[mandatory]	[multiple]	[inverse key]
admin-c:	[mandatory]	[multiple]	[inverse key]
notify:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
changed:	[mandatory]	[multiple]	[]
source:	[mandatory]	[single]	[]



Filter-set object template

<code>filter-set:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[primary/look-</code>
<code>up key]</code>			
<code>descr:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>filter:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[]</code>
<code>remarks:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[]</code>
<code>tech-c:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>admin-c:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>notify:</code>	<code>[optional]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>mnt-by:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[inverse key]</code>
<code>changed:</code>	<code>[mandatory]</code>	<code>[multiple]</code>	<code>[]</code>
<code>source:</code>	<code>[mandatory]</code>	<code>[single]</code>	<code>[]</code>



Rtr-set object template

rtr-set:	[mandatory]	[single]	[primary/look-up key]
descr:	[mandatory]	[multiple]	
members:	[optional]	[multiple]	
mbrs-by-ref:	[optional]	[multiple]	
remarks:	[optional]	[multiple]	
tech-c:	[mandatory]	[multiple]	[inverse key]
admin-c:	[mandatory]	[multiple]	[inverse key]
notify:	[optional]	[multiple]	[inverse key]
mnt-by:	[mandatory]	[multiple]	[inverse key]
changed:	[mandatory]	[multiple]	
source:	[mandatory]	[single]	