Asia Pacific Network Information Centre

APNIC

Welcome!

APNIC Internet Routing Registry Tutorial

29 July 2004, Kathmandu, Nepal

In conjunction with SANOG IV

Introduction

- Presenters
  - PART I
    - Champika Wijayatunga champika@apnic.net
  - PART II
    - Gaurab Raj Upadhaya gaurab@lahai.net

Objectives

- To provide an introduction to the APNIC Routing Registry
  - Explain concepts of the global RR
  - Outline the benefits of the APNIC Routing Registry
- NOT to:
  - Teach routing
  - Explain Internet resource policy and procedures
  - Provide advise on network configuration

Internet Routing Registry

Overview

Overview

- APNIC database recap
- What is IRR?
- Why use an IRR?
- APNIC database and the IRR
- Using RPSL in practice
- Using the Routing Registry
  - Overview of IRRToolSet
- Benefit of using IRR

APNIC Database Recap

APNIC Database

- Public network management database
  - APNIC whois database contains:
    - Internet resource information and contact details
  - APNIC Routing Registry (RR) contains:
    - routing information
- APNIC RR is part of IRR
  - Distributed databases that mirror each other

Database Object

- An object is a set of attributes and values
- Each attribute of an object...
  - Has a value
  - Has a specific syntax
  - Is mandatory or optional
  - Is single- or multi-valued
- Some attributes ...
  - Are primary (unique) keys
  - Are lookup keys for queries
  - Are inverse keys for queries
- Object “templates” illustrate this structure

Person Object Example

- Person objects contain contact information

Attributes	Values
person:	Ky Xander
address:	ExampleNet Service Provider
address:	2 Pandora St Boxville
address:	Wallis and Futuna Islands
country:	WF
phone:	+680-368-0844
fax-no:	+680-367-1797
e-mail:	kxander@example.com
nic-hdl:	KX17-AP
mnt-by:	MAINT-ENET-WF
changed:	kxander@example.com 20020731
source:	APNIC

Inetnum Object Example

– Contain IP address allocations / assignments

```
inetnum:        202.51.64.0 - 202.51.95.255
netname:        CCNEP-NP-AP
descr:          Communication & Communicate Nepal Ltd
descr:          VSAT Service Provider, Kathmandu
country:        NP
admin-c:        AS75-AP
tech-c:         AS75-AP
mnt-by:         APNIC-HM
mnt-lower:      MAINT-NP-ARUN
remarks:        -+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
remarks:        This object can only be updated by APNIC
remarks:        hostmasters.To update this object, please contact
remarks:        APNIC hostmasters and include your organisation's
remarks:        account name in the subject line.
remarks:        -+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
changed:        hostmaster@apnic.net 20010205
status:         ALLOCATED PORTABLE
source:         APNIC
```

Querying Whois db

- Unix
  - Whois –h whois.apnic.net <lookup key>
    - E.g. whois –h whois.apnic.net AS2000
- Whois web interface
  - http://www.apnic.net/apnic-bin/whois.pl
- Keys for querying
  - Primary key, other lookup keys
    - E.g. EX91-AP
  - Inverse key “-i {attribute} {value}”
    - E.g. whois -i mnt-by MAINT-EXAMPLE-AP
- APNIC whois db query options:
  - http://www.apnic.net/db/search/all-options.html

Advanced database queries

– Flags used for inetnum queries

None find exact match
- l find one level less specific matches
- L find all less specific matches
- m find first level more specific matches
- M find all More specific matches
- x find exact match (if no match, nothing)
- d enables use of flags for reverse domains
- r turn off recursive lookups

Please see “APNIC Whois Database queries” card for more details in your folder.

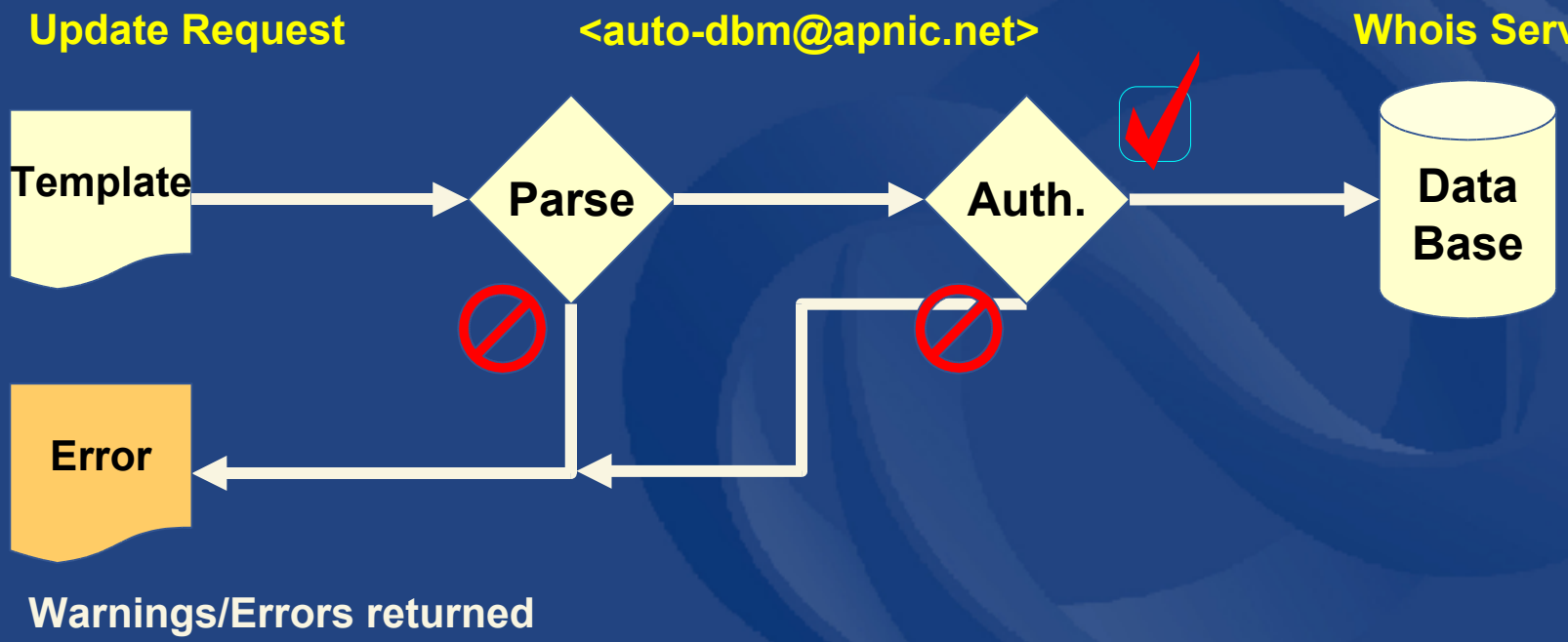
Database Update Process

- Update transactions
  - Create a new object
  - Change an object
  - Delete an object

- Updates are submitted by email
  - E-mail to <auto-dbm@apnic.net>
- Email message contains template representing new or updated object

Database Update Process

- Email requests to <auto-dbm@apnic.net>
- Each request contains an object template



Database Protection

- Authorisation
  - "mnt-by" references a mntner object
    - Can be found in all database objects
    - "mnt-by" should be used with every object!
- Authentication
  - Updates to an object must pass authentication rule specified by its maintainer object

Authentication Methods

- ‘auth’ attribute
  - Crypt-PW
    - Crypt (Unix) password encryption
    - Use web page to create your maintainer
  - PGP – GNUPG
    - Strong authentication
    - Requires PGP keys
  - MD5
    - Available

Hierarchical Authorisation

- ‘mnt-by’ attribute
  - Can be used to protect any object
  - Changes to protected object must satisfy authentication rules of ‘mntner’ object.
- ‘mnt-lower’ attribute
  - Also references mntner object
  - Hierarchical authorisation for inetnum & domain objects
  - The creation of child objects must satisfy this mntner
  - Protects against unauthorised updates to an allocated range - highly recommended!

Prerequisite for updating objects

- Create person objects for contacts
  - To provide contact info in other objects
- Create a mntner object
  - To provide protection of objects
- Protect your person object

What is an IRR?

What is a Routing Registry?

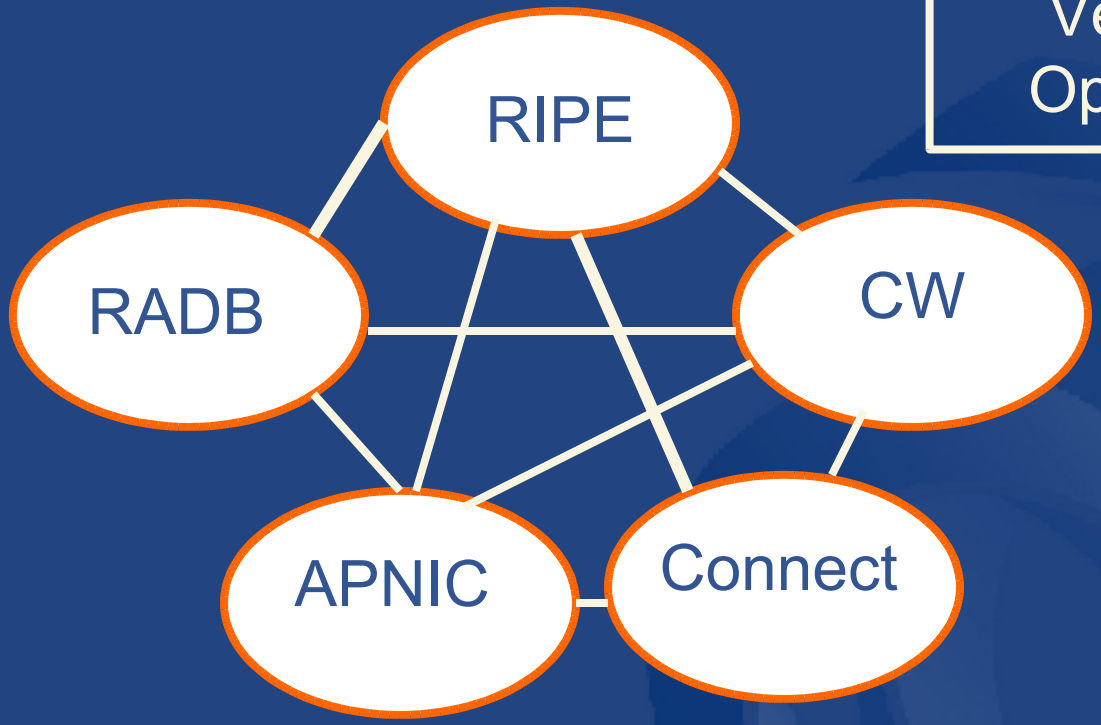
- A repository (database) of Internet routing policy information
  - ASes exchanges routing information via BGP
  - Exterior routing decisions are based on policy based rules
  - However BGP does not provides a mechanism to publish/communicate the policies themselves
  - RR provides this functionality
- Routing policy information is expressed in a series of objects

What is a Routing Registry?

- 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

What is a Routing Registry?

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



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

Overview of Routing Registry functions

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

Why use an IRR?

- Information – if every AS registers its policy and routes….
  - a global view of routing policy could be mapped
    - This global picture has the ability to improve the integrity of global Internet routing
  - Provides LIR/ISP with a mechanism to find all possible paths between any two points in the Internet
- Provides a high level of abstraction

Why use an IRR?

- Router configuration
  - By using IRRToolSet
    - ftp.ripe.net/tools/IRRToolSet
    - Extract information from IRR to create a router readable configuration file
    - Vendor independent
    - Protect against inaccurate routing info distribution
    - Verification of Internet routing
- Network troubleshooting
  - Easier to locate routing problems outside your network

What is Routing Policy?

- Description of the routing relationship between autonomous systems
  - Who are my BGP peers?
    - Customer, peers, upstream
  - What routes are:
    - Originated by each neighbour?
    - Imported from each neighbour?
    - Exported to each neighbour?
    - Preferred when multiple routes exist?
  - What to do if no route exists?
  - What routes to aggregate?

Routing Policy Specification Language

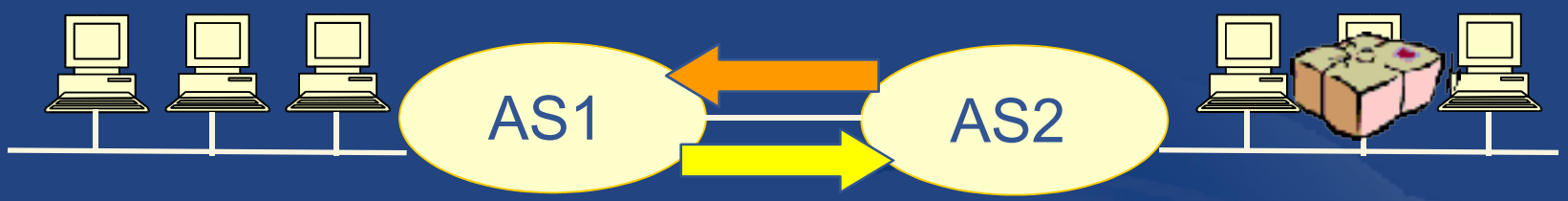
- Derived from RIPE-181
- Introduced with v3 Database
- "New" object specification language
  - more expressive syntax
  - advanced aut-num and routing policy options

- Especially useful in an Internet Routing Registry

Routing Policy Specification Language

- Purpose of RPSL
  - Allows you to specify your routing configuration in the public IRR
    - Allows you to check “Consistency” of policies and announcements
  - Gives the opportunity to consider the policies and configuration of others
  - There are required syntax and semantics which need to be understood before using RPSL

Representation of routing policy



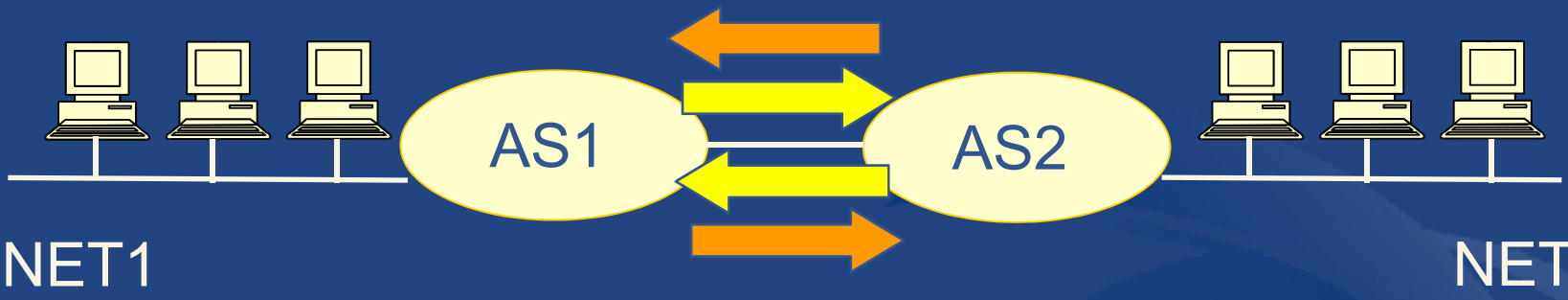
In order for traffic to flow from NET2 to NET1 between AS1 and AS2:

AS1 has to announce NET1 to AS2 via BGP

And AS2 has to accept this information and use it

Resulting in packet flow from NET2 to NET1

Representation of routing policy (cont.)



In order for traffic to flow towards from NET1 to NET2:

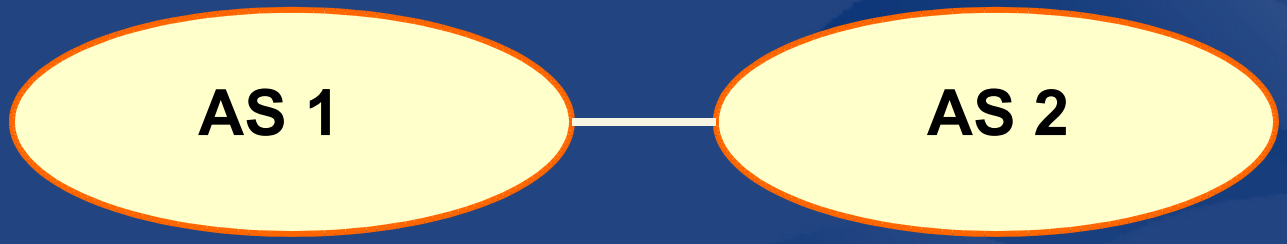
AS2 must announce NET2 to AS1

And AS1 has to accept this information and use it

Resulting in packet flow from NET 1 to NET2

Representation of routing policy

Basic concept

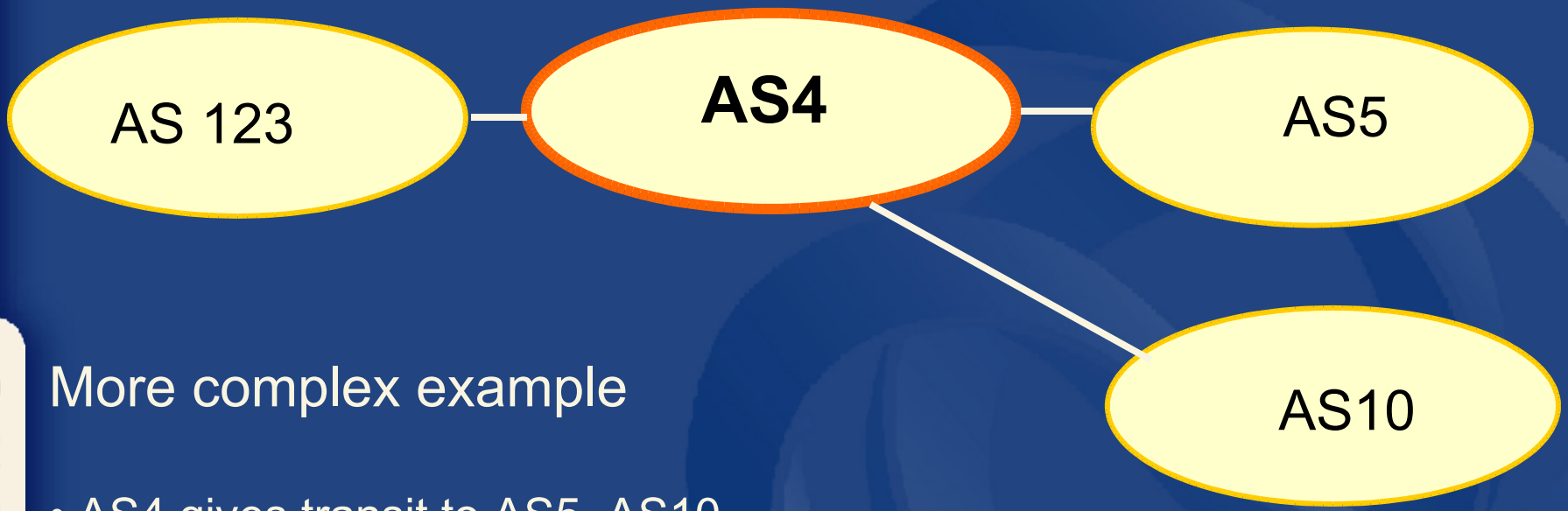


"action pref" - the lower the value, the preferred the route

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

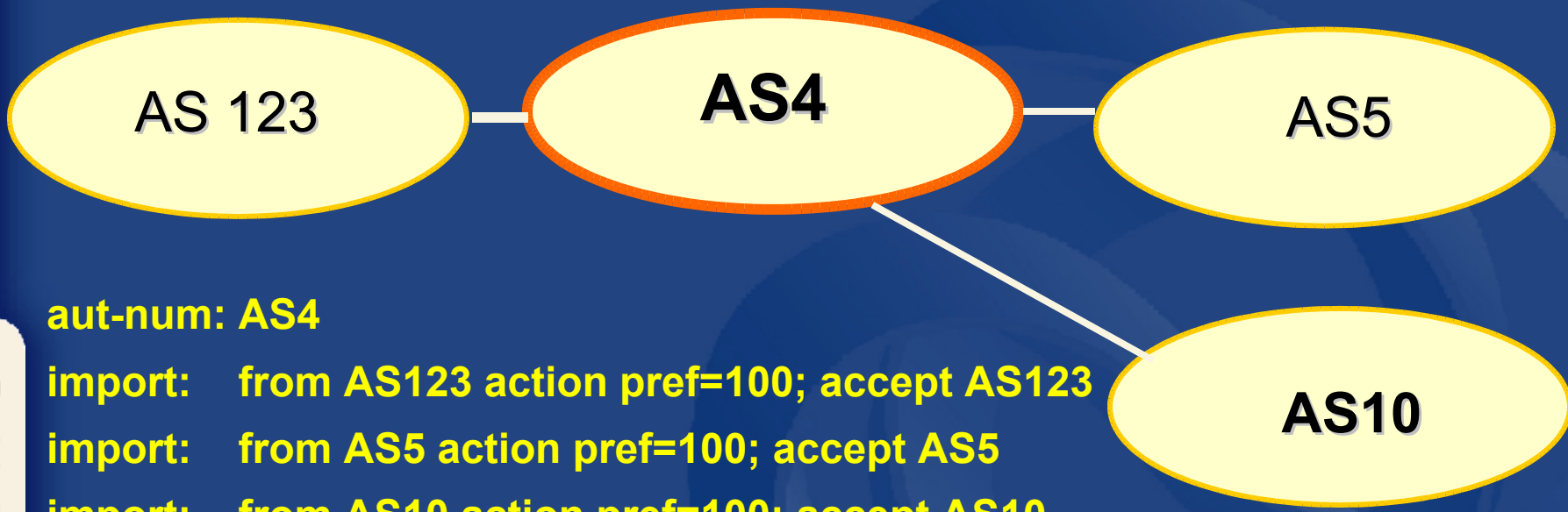
Representation of routing policy



More complex example

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

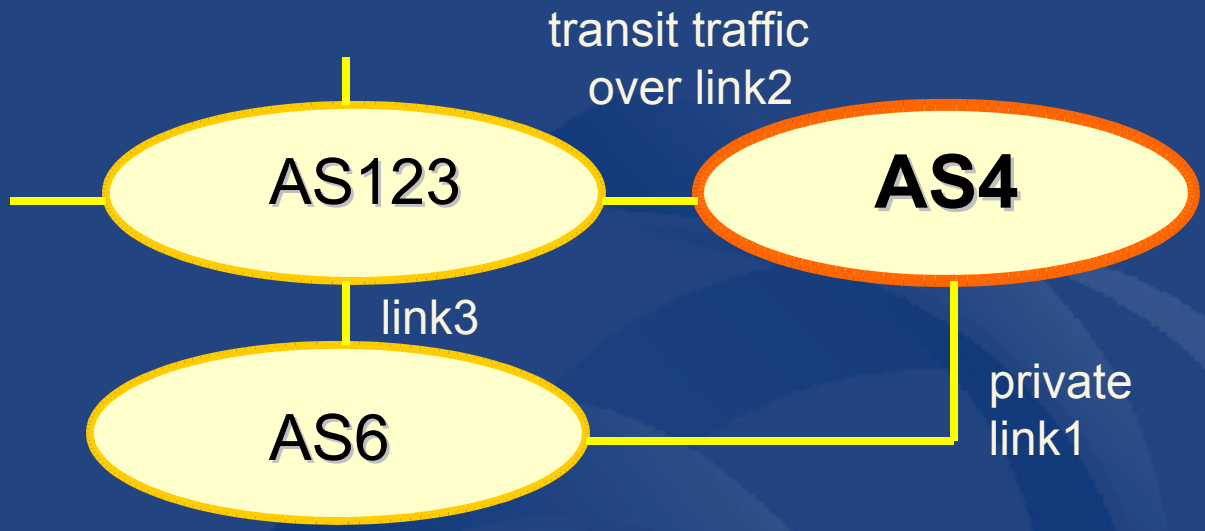
Representation of routing policy




```
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

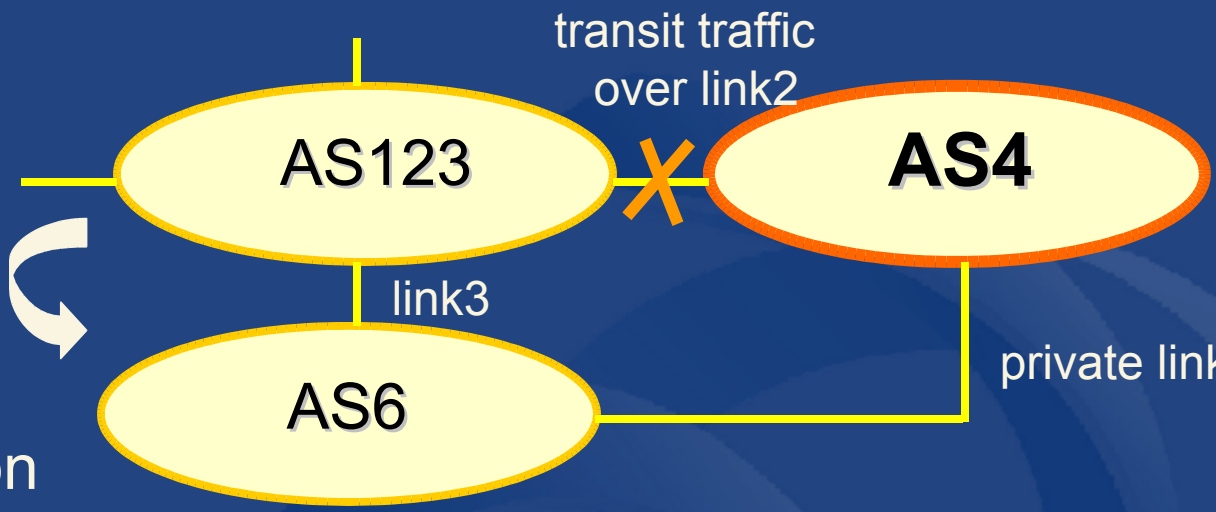
Representation of routing policy



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

Representation of routing policy



AS representation

```
aut-num:    AS4
import:  from AS123 action pref=100; accept  ANY
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
```

full routing received

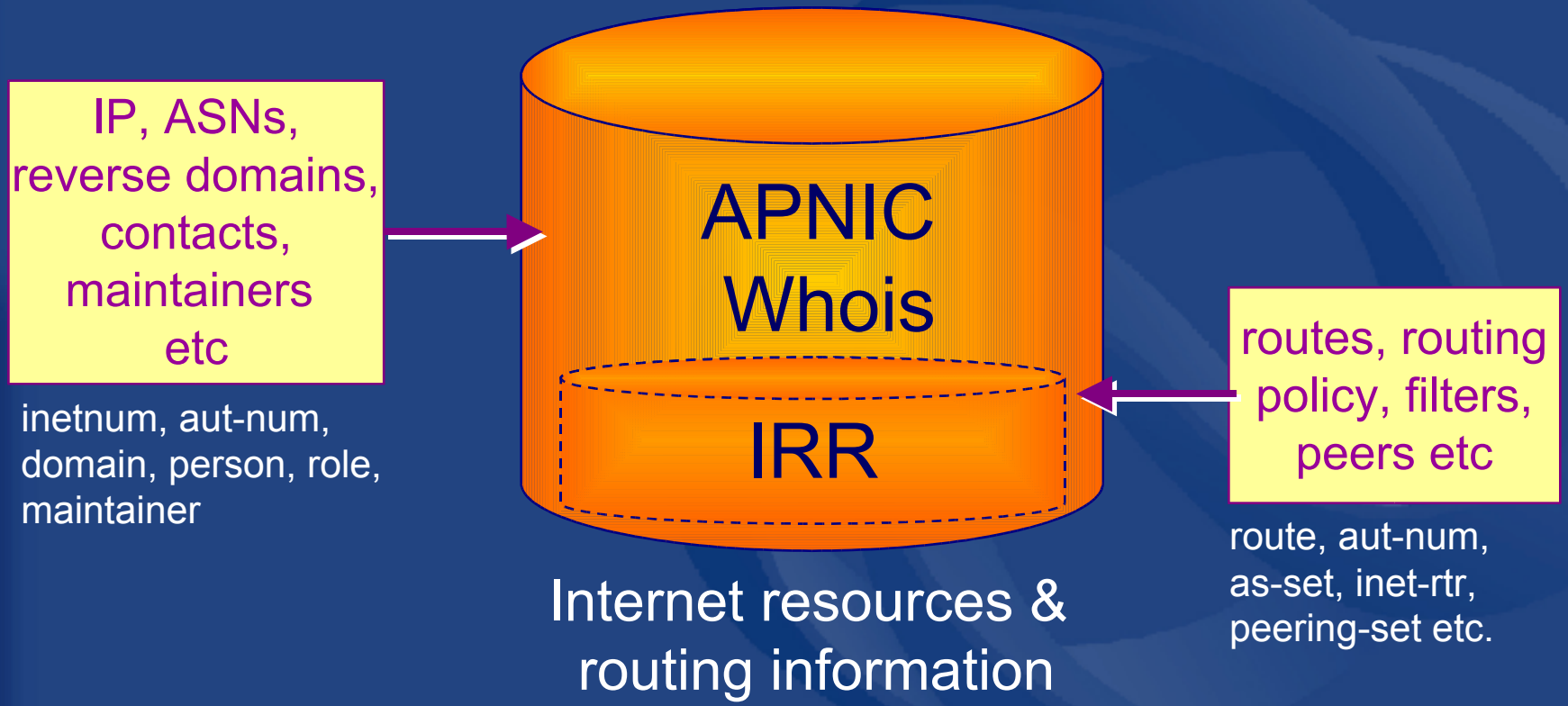
higher cost for backup route

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



RR objects review

- Aut-num object

Attribute	Value	Type
aut-num	<as-number>	mandatory, single-valued, class key
as-name	<object-name>	mandatory, single-valued
member-of	List of <as-set-name>	optional, multi-value
import	see next slide	optional, multi-value
export	see next slide	optional, multi-value

RR objects review

- route object

Attribute	Value	Type
route	Prefix of the InterAS route	mandatory, single-valued, class key
origin	<AS-number> originates the route	mandatory, single-valued
member-of	List of <route-set-name>	optional, multi-value
mnt-routes	Explained later	optional, multi-value

RR object review

- As-set object

Attribute	Value	Type
as-set	<object-name>	mandatory, single-valued, class key
members	List of <as-numbers> or <as-set-names>	optional, multi-value
Mbrs-by-ref	List of <mntner-names>	optional, multi-value

- As-set attribute starts with “as-”

RR object review

- Route-set object

Attribute	Value	Type
route-set	<object-name>	mandatory, single-valued, class key
members	List of <address-prefix-range> or <route-set-name><range-operator>	optional, multi-value
Mbrs-by-ref	List of <mntner-names>	optional, multi-value

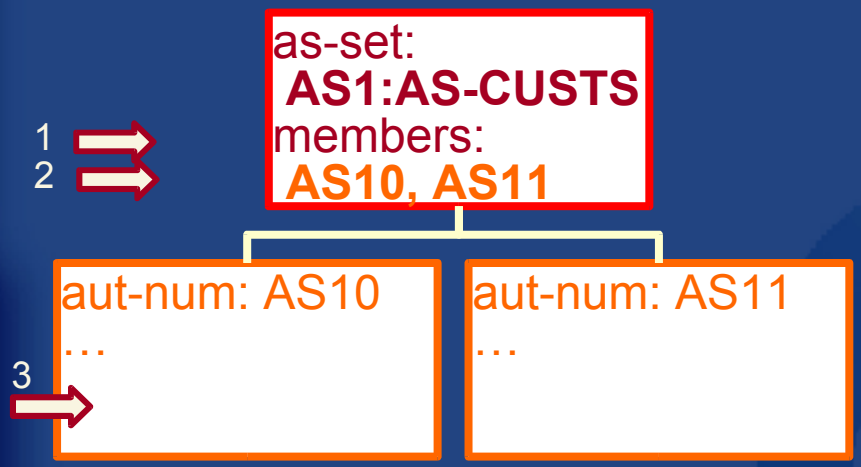
- Route-set attribute starts with “rs-”

‘Set-’ objects and their members

- Two ways of referencing members

members

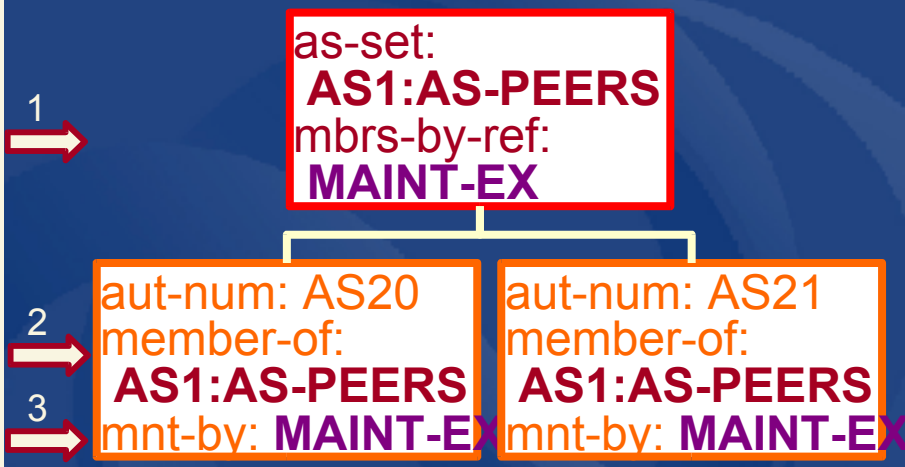
- members specified in the ‘set-’ object



1. ‘members’ specifies members of the set
2. Members added in the ‘set-’ object
3. No need to modify the member object when adding members

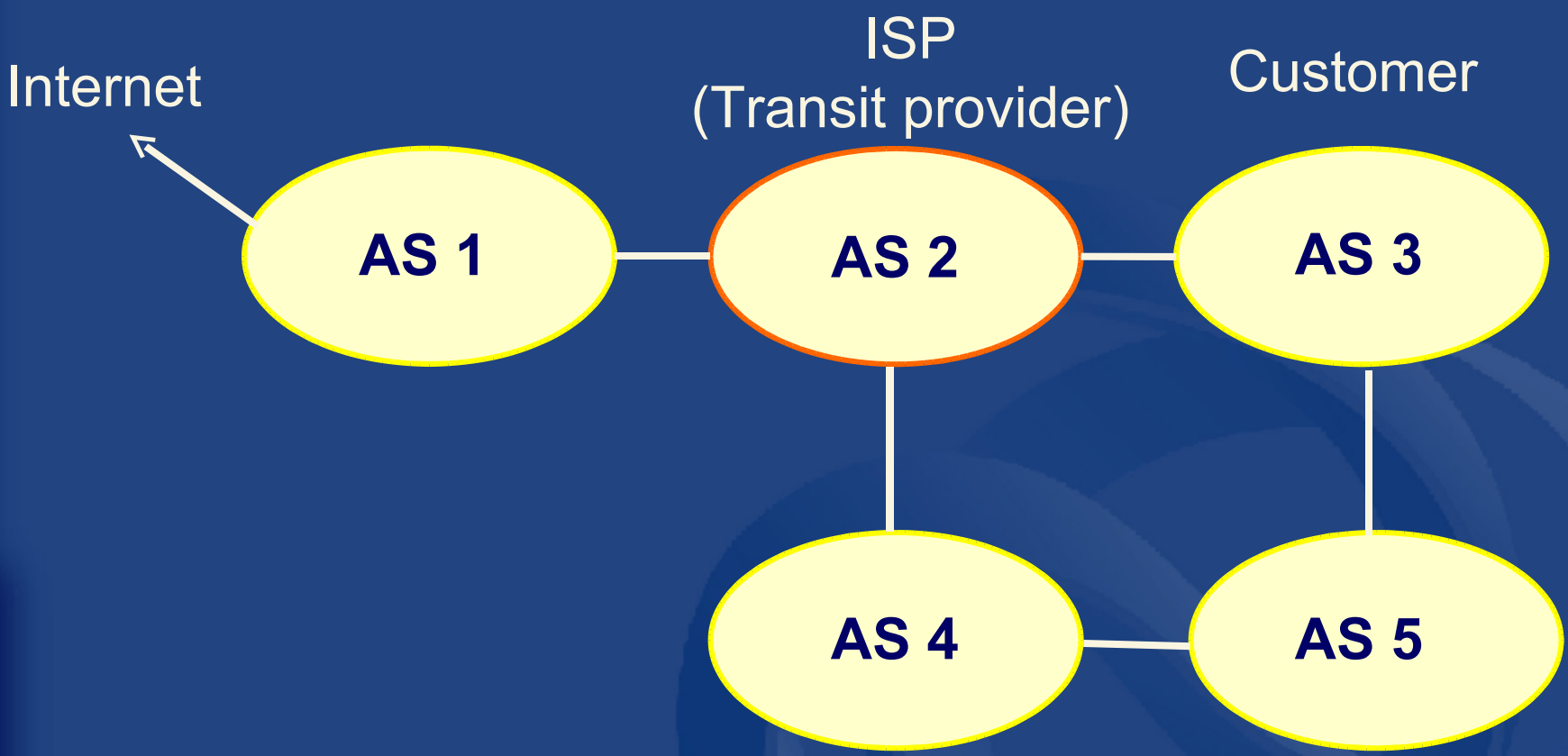
mbrs–by–ref

- ‘set’ specified in the member objects



1. ‘mbrs-by-ref’ specifies the maintainer of the members.
2. Members reference the ‘set-’ object in the ‘member-of’ attribute
3. Members are maintained by the maintainer specified in the ‘set-’

Common peering policies



- Peering policies of an AS
  - Registered in an aut-num object

Common peering policies

```
aut-num:        AS2
as-name:        SAMPLE-NET
dsescr:         Sample AS
import:         from AS1 accept ANY
import:         from AS3 accept <^AS3+$>
export:         to AS3 announce ANY
export:         to AS1 announce AS2 AS3
admin-c:        SN36-AP
tech-c:         MF53-AP
mtn-by:         MAINT-SAMPLE-AP
changed:        sample@sample.net
```

ISP customer – transit provider policies

- Policy for AS3 and AS4 in the AS2 aut-num object

```
aut-num:    AS2
import:     from AS1 accept ANY
import:     from AS3 accept <^AS3+$>
import:     from AS4 accept <^AS4+$>
export:     to AS3 announce ANY
export:     to AS4 announce ANY
export:     to AS1 announce AS2 AS3 AS4
```

AS-set object

- Describe the customers of AS2

```
as-set:     AS2:AS-CUSTOMERS
members:    AS3 AS4
changed:    sample@sample.net
source:     APNIC
```

Aut-num object referring as-set object

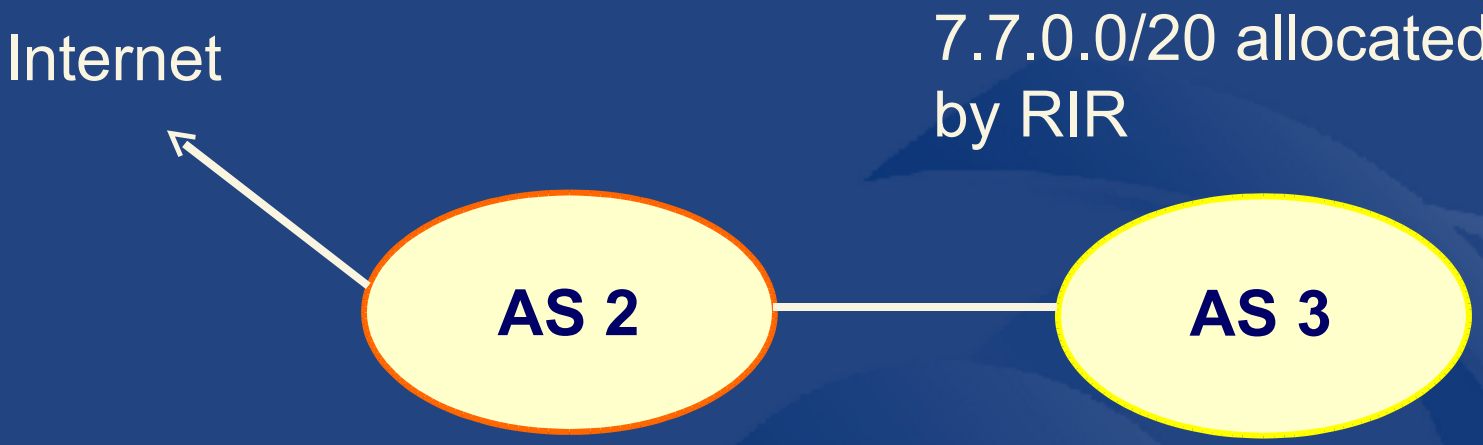
```
aut-num:    AS2
import:     from AS1 accept ANY
import:     from AS2:AS-CUSTOMERS accept
            <^AS2:AS-CUSTOMERS+$>
export:     to AS2:AS-CUSTOMERS announce ANY
export:     to AS1 announce AS2 AS2:AS-
            CUSTOMERS
```

```
aut-num:    AS1
import:     from AS2 accept <^AS2+AS2:AS-
            CUSTOMERS+$>
export:     ……
```

Express filtering policy

- To limit the routes one accepts from a peer
  - To prevent the improper use of unassigned address space
  - To prevent malicious use of another organisation's address space

Filtering policy



AS3 wants to announce part or all of 7.7.0.0/20 on the global Internet.

AS2 wants to be certain that it only accepts announcements from AS3 for address space that has been properly allocated to AS3.

Aut-num object with filtering policy

```
aut-num:    AS2
import:     from AS3 accept { 7.7.0.0/20^20-24 }
.......
```

For an ISP with a growing or changing customer base, this mechanism will not scale well.

Route-set object can be used.

Route-set

route-set:	AS2:RS-ROUTES:AS3
members:	7.7.0.0/20^20-24
changed:	sample@sample.net
source:	APNIC

Specifies the set of routes that will be accepted from a given customer

Set names are constructed hierarchically:

AS2 : RS-ROUTES : AS3

AS2 indicates whose sets these are

AS3 indicates peer AS

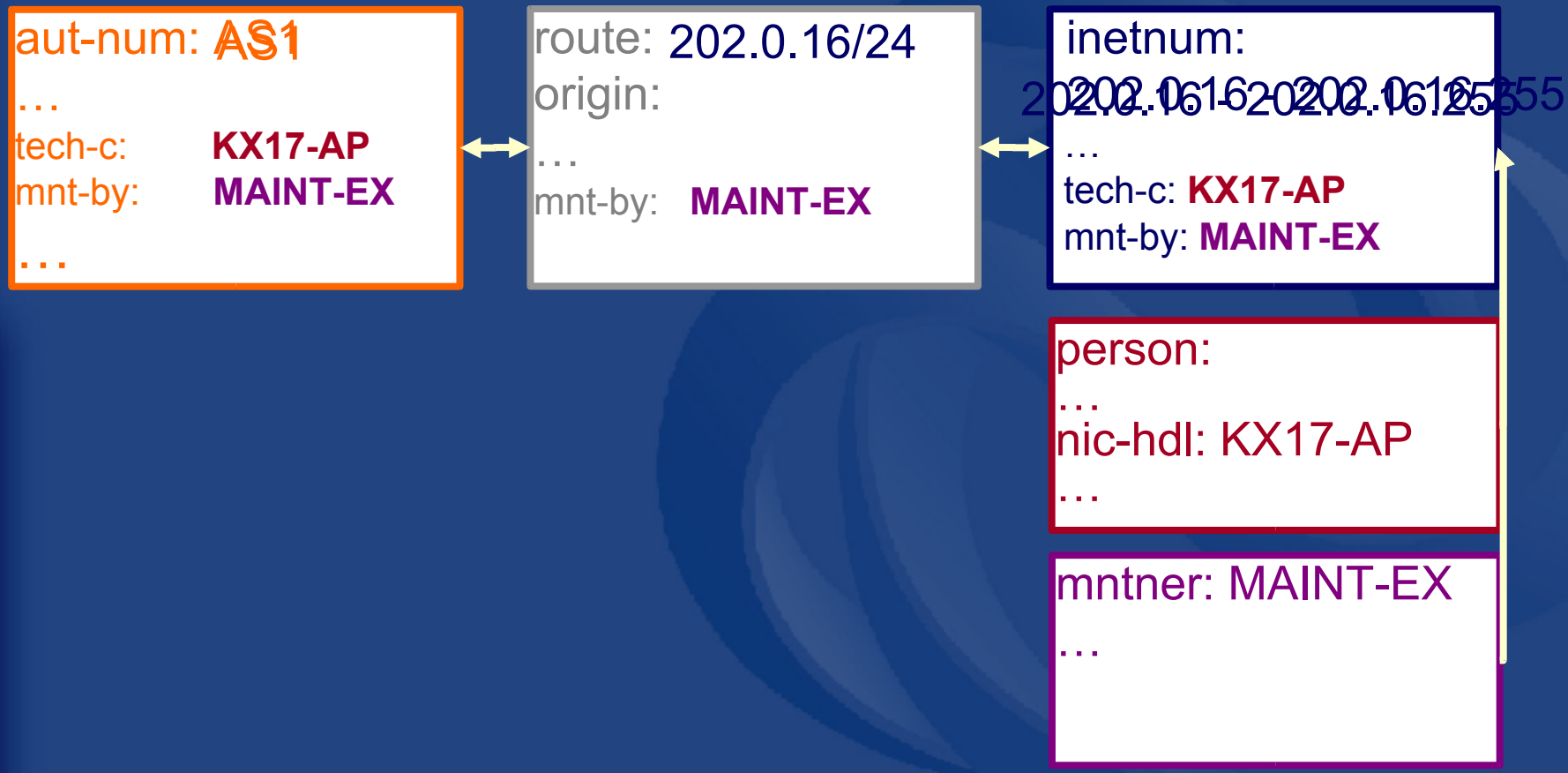
Filter configuration using route-set – AS2

```
import:     from AS1 accept ANY
import:     from AS3 accept AS2:RS-ROUTES:AS3
import:     from AS4 accept AS2:RS-ROUTES:AS4
export:     to AS2:AS-CUSTOMERS announce ANY
export:     to AS1 announce AS2 AS2:AS-CUSTOMERS
```

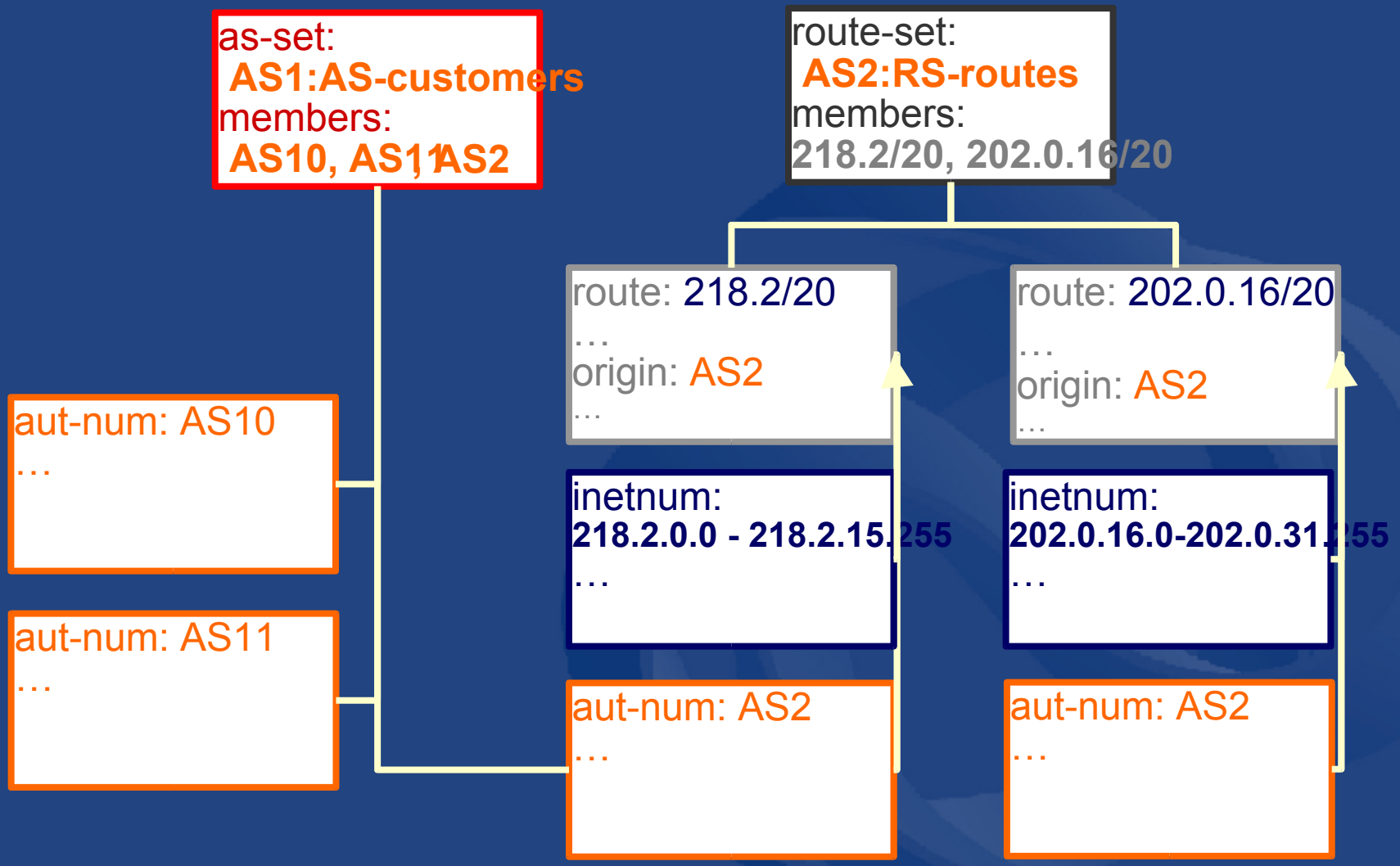
RPSL allows the peer's AS number to be replaced by the keyword PeerAS

```
import:  from AS2:AS-CUSTOMERS accept
         AS2:RS-ROUTES:PeerAS
```

Inter-related IRR objects



Inter-related IRR objects



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 - 202.137.196.255
netname:      SPARKYNET-WF
descr:        SparkyNet Service Provider
…
mnt-by:       APNIC-HM
mnt-lower:    MAINT-SPARKYNET1-WF
mnt-routes:   MAINT-SPARKYNET2-WF
```

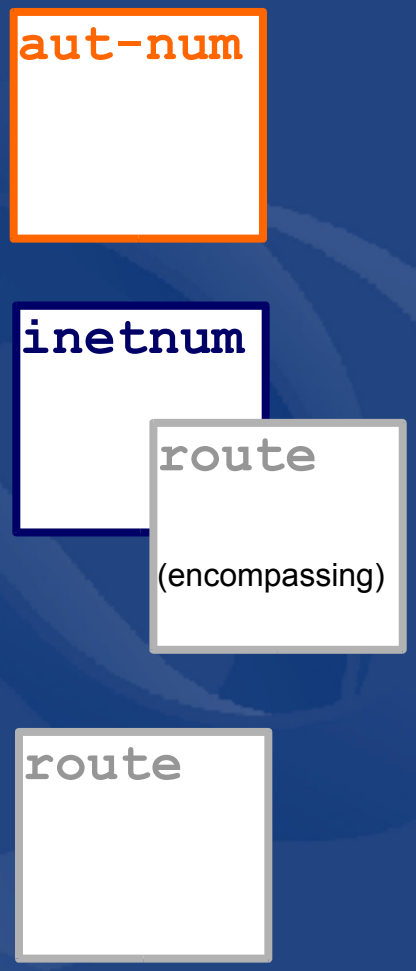
This object can only be modified by APNIC

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

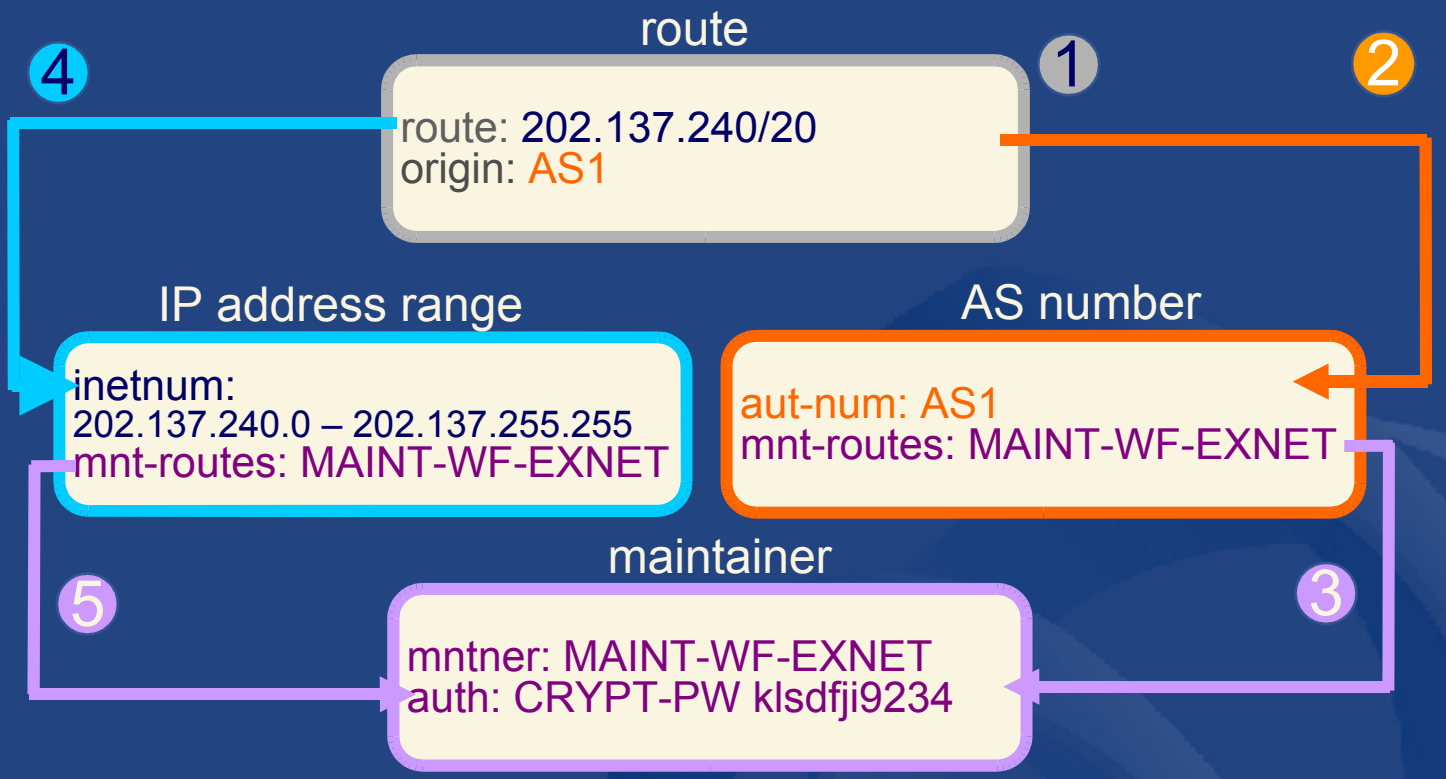
Creation of route objects matching/within this range has to pass the authentication of MAINT-SPARKYNET-WF

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



Creating route objects



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

Asia Pacific Network Information Centre

APNIC

Using the Routing Registry

Overview of the IRRToolSet

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

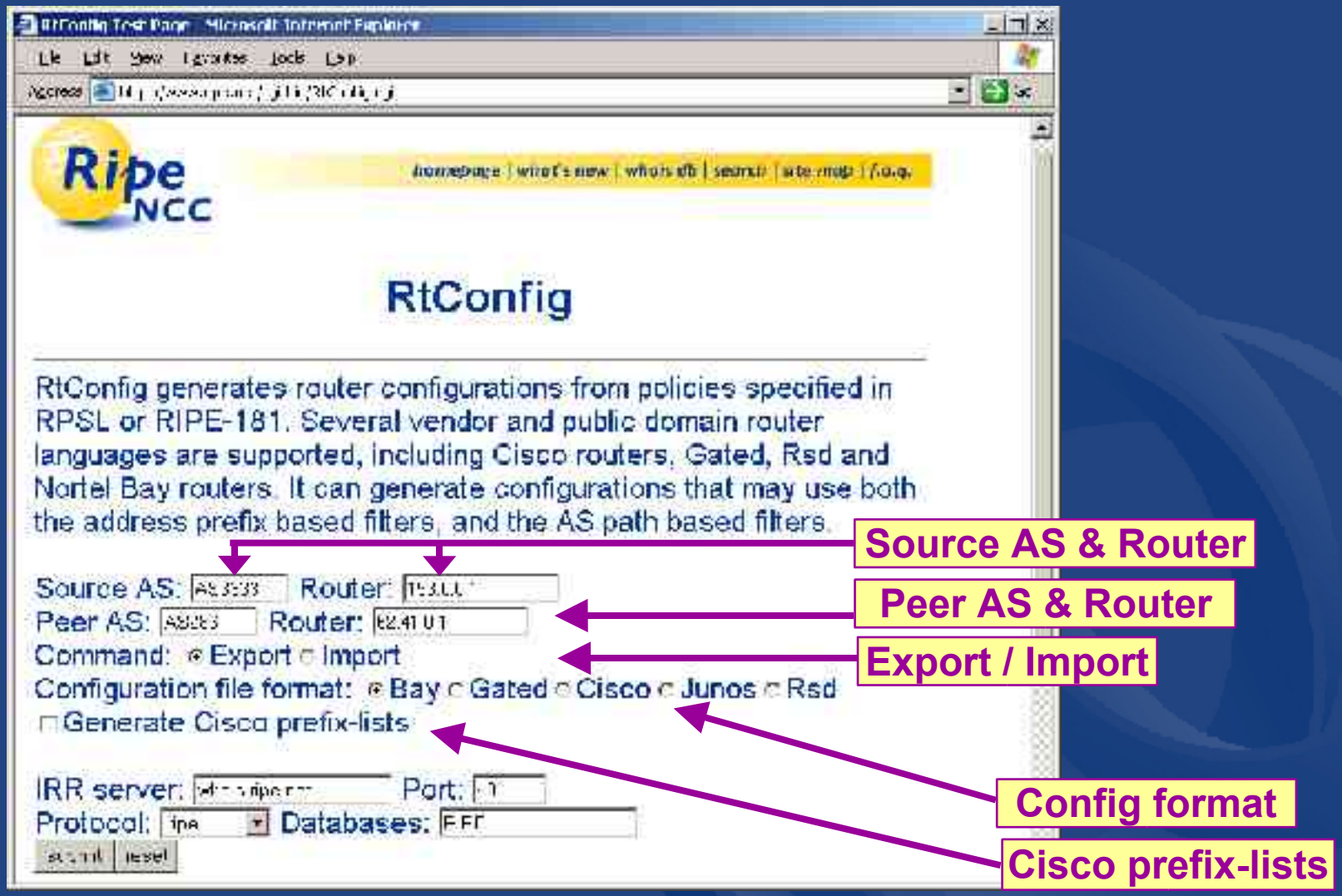
Use of RPSL - RtConfig

- 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 :-)

RtConfig – web prototype



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

Benefit of using IRR

Using the Routing Registry

Costs

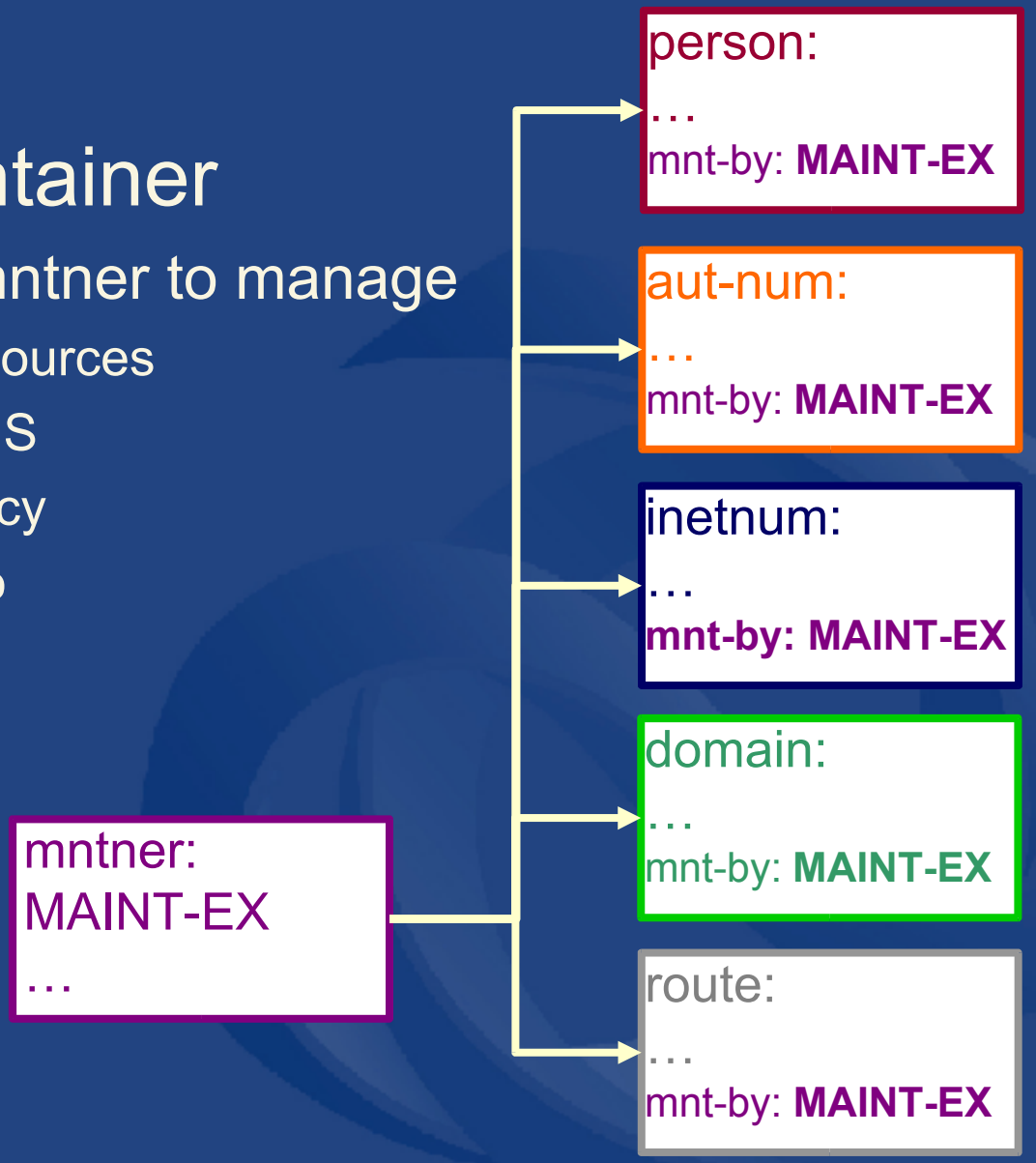
- Requires some initial planning
- Takes some time to define & register policy
- Need to maintain data in RR

Benefits

- 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)

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!

Questions ?

Thank you!