



# Deploying OSPF for ISPs

**ISP/IXP Workshops**

# Agenda

- **OSPF Design in SP Networks**
- **Adding Networks in OSPF**
- **OSPF in IOS**

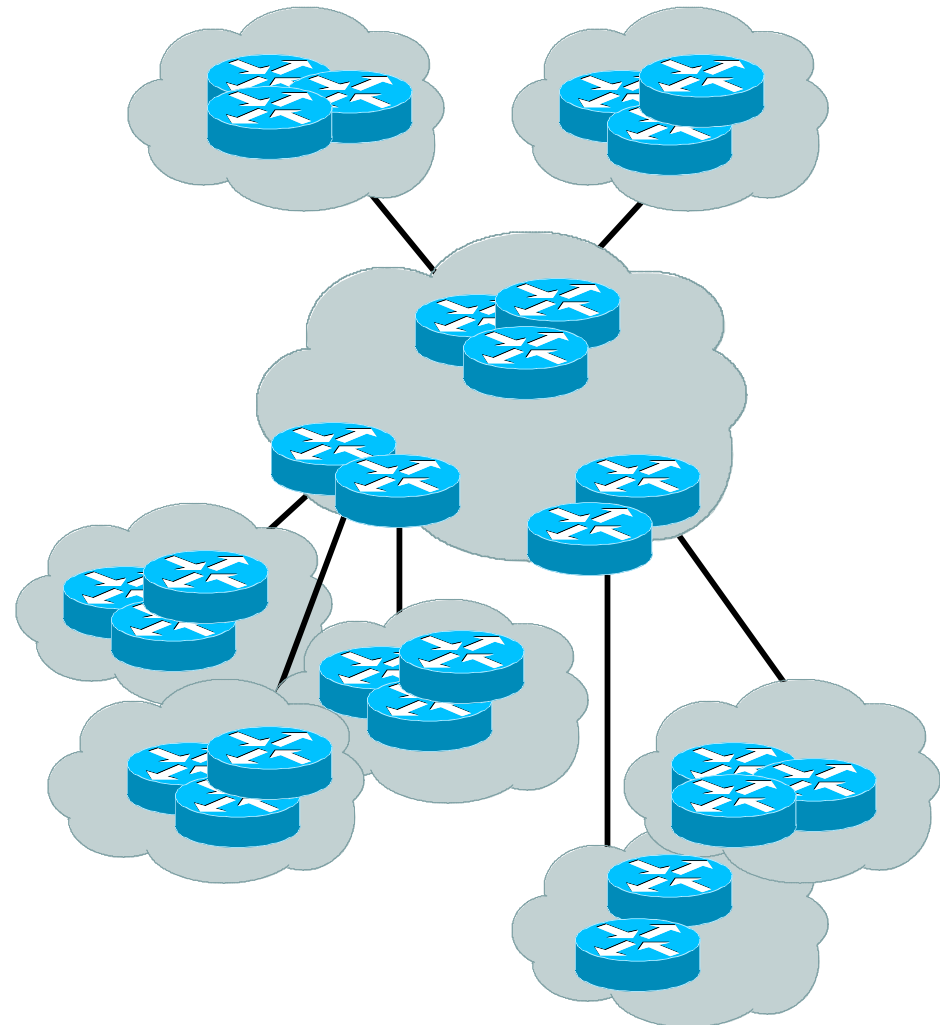


# OSPF Design

**As applicable to Service Provider Networks**

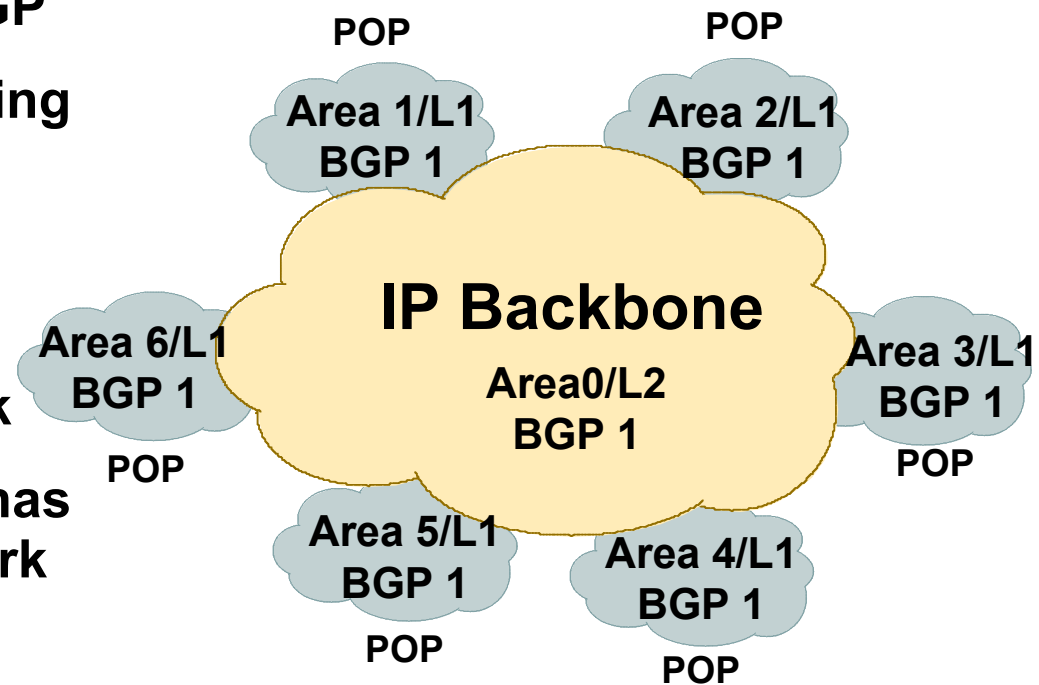
# Service Providers

- **SP networks are divided into PoPs**
- **Transit routing information is carried via BGP**
- **IGP is used to carry next hop only**
- **Optimal path to the next hop is critical**



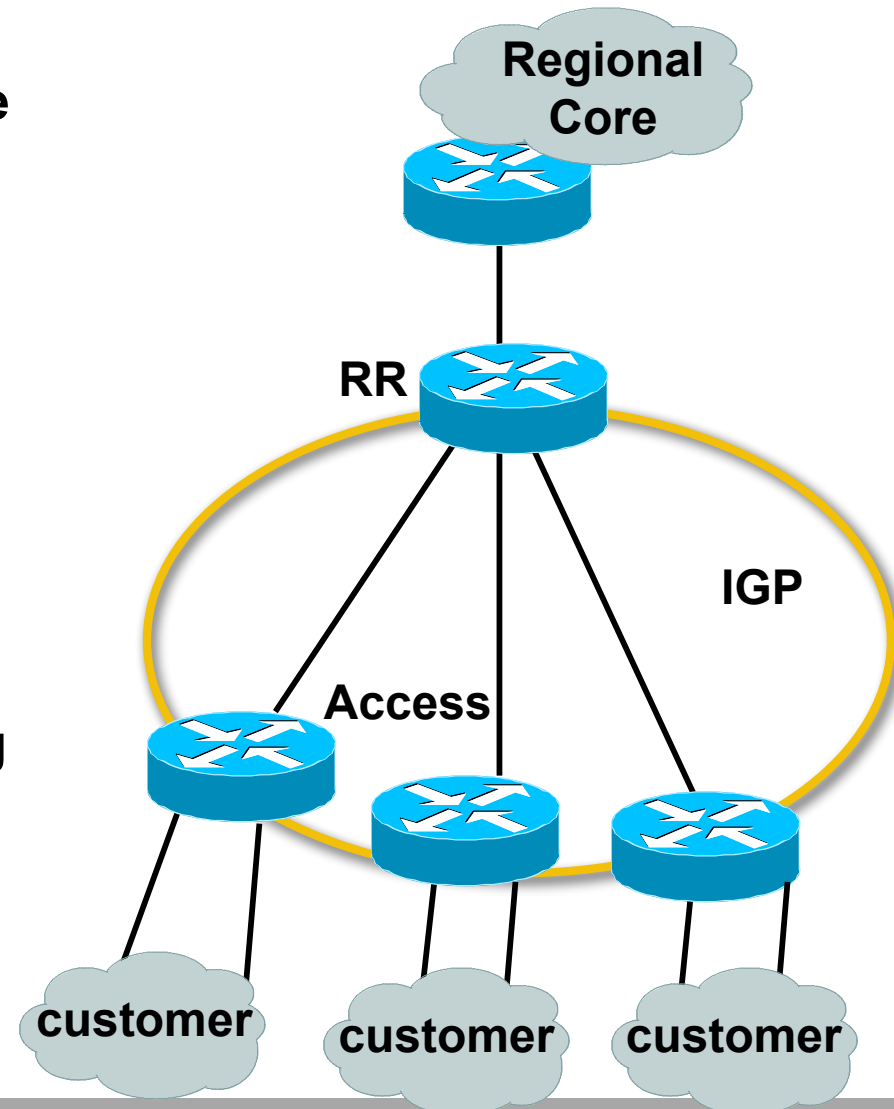
# SP Architecture

- Major routing information is ~190K prefixes via BGP
- Largest known IGP routing table is ~6–7K
- Total of 196K
- 6K/196K ~ 4% of IGP routes in an ISP network
- A very small factor but has a huge impact on network convergence!



# SP Architecture

- You can reduce the IGP size from 6K to approx the number of routers in your network
- This will bring really fast convergence
- Optimise where you must and summarise where you can
- Stops unnecessary flapping

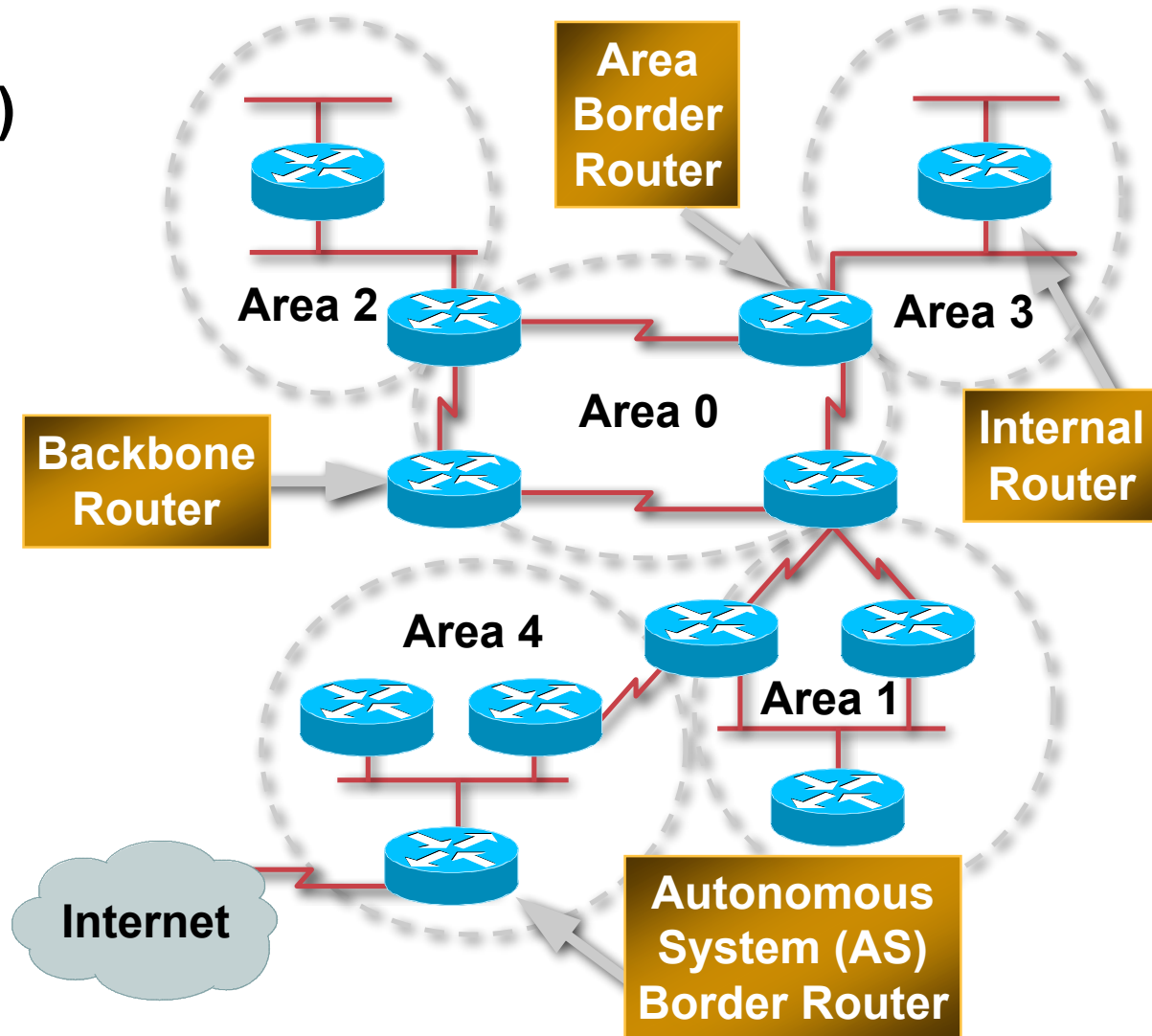


# OSPF Areas and Rules

Backbone area (0) must be present

All other areas must have connection to backbone

Backbone must be contiguous  
Do **NOT** partition area (0)



# OSPF Design: Addressing

- **OSPF Design and Addressing go together**

**Objective is to keep the Link State Database *lean***

**Create an address hierarchy to match the topology**

**Use separate Address Blocks for network infrastructure, customer interfaces, customers, etc.**



# OSPF Design: Areas

- **Examine physical topology**
  - Is it meshed or hub-and-spoke?
- **Use areas and summarisation**
  - This reduces overhead and LSA counts
  - (but watch next-hop for iBGP when summarising)
- **Don't bother with the various stub areas**
  - No benefits for ISPs, causes problems for iBGP
- **Push the creation of a backbone**
  - Reduces mesh and promotes hierarchy

# OSPF Design: Areas

- **One SPF per area, flooding done per area**  
Watch out for overloading ABRs
- **Avoid externals in OSPF**  
External LSAs flood through entire network
- **Different types of areas do different flooding**
  - Normal areas
  - Stub areas
  - Totally stubby (stub no-summary)
  - Not so stubby areas (NSSA)

# OSPF Design: Summary

- **Think Redundancy**

  - **Dual Links out of each area – using metrics (cost) for traffic engineering**

- **Too much redundancy...**

  - **Dual links to backbone in stub areas must be the same cost – other wise sub-optimal routing will result**

  - **Too Much Redundancy in the backbone area without good summarization will effect convergence in the area 0**



# OSPF for Service Providers

## Adding Networks

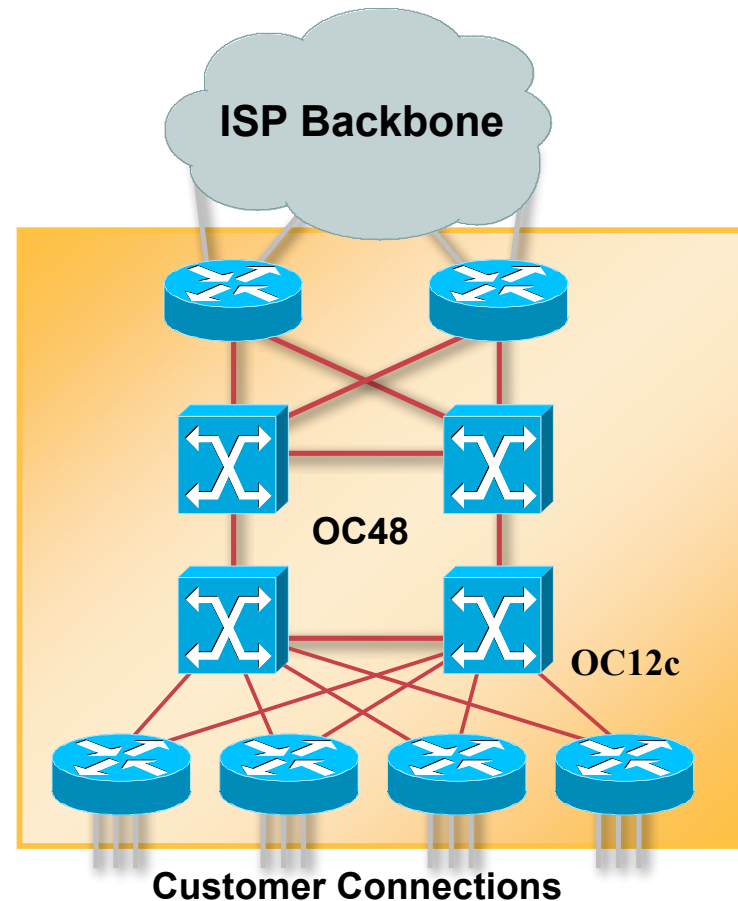
# OSPF: Adding Networks

- **BCP – Individual OSPF Network statement for each infrastructure link**

Have separate IP address blocks for *infrastructure* and *customer links*

Use *IP Unnumbered* Interfaces or *iBGP next-hop-self* for customer /30 point-to-point links

OSPF should only carry infrastructure routes in an ISP's network



# OSPF: Adding Networks Method One

- **redistribute connected subnets**

**Works for all connected interfaces on the router but sends networks as external type-2s – which are not summarized**

```
router ospf 100
  redistribute connected subnets
```

- **Do NOT do this!**

**Because:**

**Type-2 LSAs flood through entire network**

**These LSAs are not all useful for determining paths through backbone; simply take up space**

# OSPF: Adding Networks Method Two

- **Specific network statements**

**Every active interface with a configured IP address needs an OSPF network statement**

**Interface that will have no OSPF neighbours needs *passive-interface* to disable OSPF Hello's**

**That is: all interfaces connecting to devices outside the ISP backbone (i.e. customers, peers, etc)**

```
router ospf 100  
  
network 192.168.1.1 0.0.0.3 area 51  
  
network 192.168.1.5 0.0.0.3 area 51  
  
passive interface Serial 1/0
```

# OSPF: Adding Networks Method Three

- **Network statements – wildcard mask**

**Every active interface with configured IP address covered by wildcard mask used in OSPF network statement**

**Interfaces covered by wildcard mask but having no OSPF neighbours need *passive-interface* (or use *passive-interface default* and then activate the interfaces which will have OSPF neighbours)**

```
router ospf 100
  network 192.168.1.0 0.0.0.255 area 51
  passive-interface default
  no passive interface POS 4/0
```



# OSPF: Adding Networks Recommendations

- **Don't ever use Method 1**
- **Method 2 doesn't scale too well when router has a large number of interfaces but only a few with OSPF neighbours**
  - solution is to use Method 3 with “no passive” on interfaces with OSPF neighbours
- **Method 2 is fine for core/infrastructure routers**
- **Method 3 is preferred for aggregation routers**
  - Or use iBGP next-hop-self
  - Or even ip unnumbered on external point-to-point links

# OSPF: Adding Networks

## Example One

- Aggregation router with large number of leased line customers and just two links to the core network:

```
interface loopback 0
  ip address 192.168.255.1 255.255.255.255
interface POS 0/0
  ip address 192.168.10.1 255.255.255.252
interface POS 1/0
  ip address 192.168.10.5 255.255.255.252
interface serial 2/0:0 ...
  ip unnumbered loopback 0
! Customers connect here ^^^^^^
router ospf 100
  network 192.168.255.1 0.0.0.0 area 51
  network 192.168.10.0 0.0.0.3 area 51
  network 192.168.10.4 0.0.0.3 area 51
  passive-interface default
  no passive interface POS 0/0
  no passive interface POS 1/0
```

# OSPF: Adding Networks Example Two

- **Core router with only links to other core routers (as core routers do!):**

```
interface loopback 0
  ip address 192.168.255.1 255.255.255.255
interface POS 0/0
  ip address 192.168.10.129 255.255.255.252
interface POS 1/0
  ip address 192.168.10.133 255.255.255.252
interface POS 2/0
  ip address 192.168.10.137 255.255.255.252
interface POS 2/1
  ip address 192.168.10.141 255.255.255.252
router ospf 100
  network 192.168.255.1 0.0.0.0 area 0
  network 192.168.10.128 0.0.0.3 area 0
  network 192.168.10.132 0.0.0.3 area 0
  network 192.168.10.136 0.0.0.3 area 0
  network 192.168.10.140 0.0.0.3 area 0
  passive interface loopback 0
```

# OSPF: Adding Networks Summary

- **Key Theme when selecting a technique: Keep the Link State Database Lean**

**Increases Stability**

**Reduces the amount of information in the Link State Advertisements (LSAs)**

**Speeds Convergence Time**

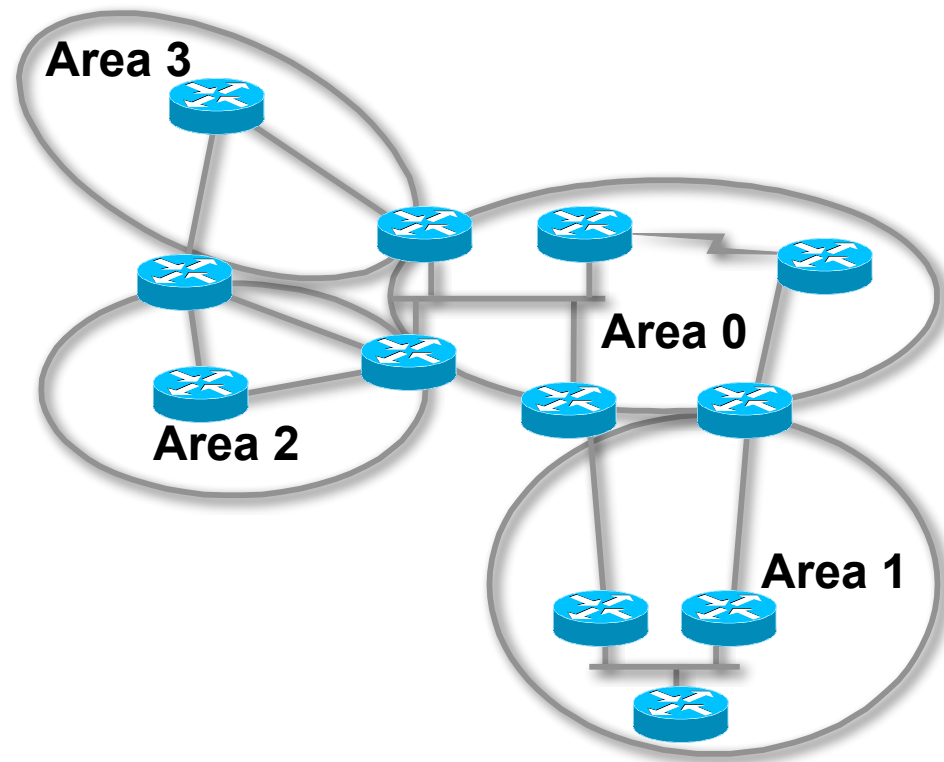


# OSPF in IOS

**Useful features for ISPs**

# Areas

- **An area is stored as a 32-bit field:**
  - Defined in IP address format
  - Can also be defined using single decimal value (i.e., Area 0.0.0.0, or Area 0)
- **0.0.0.0 reserved for the backbone area**



# Logging Adjacency Changes

- **The router will generate a log message whenever an OSPF neighbour changes state**

- **Syntax:**

```
[no] [ospf] log-adjacency-changes
```

**(OSPF keyword is optional, depending on IOS version)**

- **Example of a typical log message:**

```
%OSPF-5-ADJCHG: Process 1, Nbr 223.127.255.223 on  
Ethernet0 from LOADING to FULL, Loading Done
```

# Number of State Changes

- **The number of state transitions is available via SNMP (ospfNbrEvents) and the CLI:**

```
show ip ospf neighbor [type number]  
[neighbor-id] [detail]
```

**Detail—(Optional) Displays all neighbours given in detail (list all neighbours). When specified, neighbour state transition counters are displayed per interface or neighbour ID**



# State Changes (Continued)

- To reset OSPF-related statistics, use the **clear ip ospf counters EXEC** command. At this point **neighbor** is the only available option; it will reset neighbour state transition counters per interface or neighbour id

```
clear ip ospf counters [neighbor [<type number>]  
[neighbor-id]]
```

# Router ID

- If the loopback interface exists and has an IP address, that is used as the router ID in routing protocols – **stability!**
- If the loopback interface does not exist, or has no IP address, the router ID is the highest IP address configured – **danger!**
- OSPF sub command to manually set the Router ID:

```
router-id <ip address>
```

# Cost & Reference Bandwidth

- **Bandwidth used in Metric calculation**

$$\text{Cost} = 10^8 / \text{bandwidth}$$

**Not useful for interface bandwidths > 100 Mbps**

- **Syntax:**

```
ospf auto-cost reference-bandwidth <reference-bw>
```

- **Default reference bandwidth still 100 Mbps for backward compatibility**
- **Most ISPs simply choose to develop their own cost strategy and apply to each interface type**

# Cost: Example Strategy

<b>40GE</b>	<b>40Gbps</b>	<b>cost = 1</b>
<b>10GE/OC192</b>	<b>10Gbps</b>	<b>cost = 5</b>
<b>OC48</b>	<b>2.5Gbps</b>	<b>cost = 10</b>
<b>GigEthernet</b>	<b>1Gbps</b>	<b>cost = 20</b>
<b>OC12</b>	<b>622Mbps</b>	<b>cost = 50</b>
<b>OC3</b>	<b>155Mbps</b>	<b>cost = 100</b>
<b>FastEthernet</b>	<b>100Mbps</b>	<b>cost = 500</b>
<b>Ethernet</b>	<b>10Mbps</b>	<b>cost = 1000</b>
<b>E1</b>	<b>2Mbps</b>	<b>cost = 5000</b>

# Clear/Restart

- **OSPF clear commands**

If no process ID is given, all OSPF processes on the router are assumed

- `clear ip ospf [pid] redistribution`

This command clears redistribution based on OSPF routing process ID

- `clear ip ospf [pid] counters`

This command clears counters based on OSPF routing process ID

- `clear ip ospf [pid] process`

This command will restart the specified OSPF process. It attempts to keep the old router-id, except in cases where a new router-id was configured or an old user configured router-id was removed. Since this command can potentially cause a network churn, a user confirmation is required before performing any action

# Use OSPF Authentication

- **Use authentication; too many people overlook this basic feature**
- **When using authentication, use the MD5 feature**

```
area <area-id> authentication message-digest  
(whole area)
```

```
ip ospf message-digest-key 1 md5 <key>
```

- **Authentication can be selectively disabled per interface with:**

```
ip ospf authentication null
```

# Tuning OSPF (1)

- **Hello/Dead Timers**

*ip ospf hello-interval 3 (default 10)*

*ip ospf dead-interval 15 (default is 4x hello)*

**This allows for faster network awareness of a failure, and can result in faster reconvergence, but requires more router CPU and generates more overhead**

- **LSA Pacing**

*timers lsa-group-pacing 300 (default 240)*

**This is a great feature; allows grouping and pacing of LSA updates at configured interval; reduces overall network and router impact**

# Tuning OSPF (2)

- **DR/BDR Selection**

*ip ospf priority 100 (default 1)*

This feature should be in use in your OSPF network; forcibly set your DR and BDR per segment so that they are known; choose your most powerful, or most idle routers; try to keep the DR/BDR limited to one segment each

- **OSPF Internal Timers**

*timers spf 2 8 (default is 5 and 10)*

Allows you to adjust SPF characteristics; first number sets wait time from topology change to SPF run; second is hold-down between SPF runs; **BE CAREFUL WITH THIS COMMAND**; if you're not sure when to use it, it means you don't need it; default is 95% effective



# Tuning OSPF (3)

- **OSPF startup**

*max-metric router-lsa on-startup wait-for-bgp*

**Avoids blackholing traffic on router restart**

**Causes OSPF to announce its prefixes with highest possible metric until iBGP is up and running**

**When iBGP is running, OSPF metrics return to normal, make the path valid**

**(Introduced by CSCdp82156 in 12.0S, 12.2 and 12.2S)**

**(Equivalent to ISIS setting over-load-bit)**

# Tuning OSPF (4)

- **LSA filtering/interface blocking**

*Per interface:*

*ip ospf database-filter all out (no options)*

*Per neighbor:*

*neighbor 1.1.1.1 database-filter all out (no options)*

**OSPFs router will flood an LSA out all interfaces except the receiving one; LSA filtering can be useful in cases where such flooding unnecessary (i.e., NBMA networks), where the DR/BDR can handle flooding chores**

*area <area-id> filter-list <acl>*

**Filters out specific Type 3 LSAs at ABRs**

- **Improper use can result in routing loops and black-holes that can be very difficult to troubleshoot**



# Deploying OSPF for ISPs

**ISP/IXP Workshops**



# OSPF Command Summary

# Redistributing Routes into OSPF

```
ROUTER OSPF <pid#x>  
  REDISTRIBUTE {protocol} <as#y>  
    <metric>  
    <metric-type (1 or 2)>  
    <tag>  
    <subnets>
```

# Router Sub-commands

```
NETWORK <n.n.n.n> <mask> AREA <area-id>  
AREA <area-id> STUB {no-summary}  
AREA <area-id> AUTHENTICATION  
AREA <area-id> DEFAULT_COST <cost>  
AREA <area-id> VIRTUAL-LINK <router-id>...  
AREA <area-id> RANGE <address mask>
```

# Interface Subcommands

```
IP OSPF COST <cost>
```

```
IP OSPF PRIORITY <8-bit-number>
```

```
IP OSPF HELLO-INTERVAL <number-of-seconds>
```

```
IP OSPF DEAD-INTERVAL <number-of-seconds>
```

```
IP OSPF AUTHENTICATION-KEY <8-bytes-of-password>
```