Open Shortest Path First - OSPF

IP Routing

Jim Binkley

Outline

- overview
- theory
  - database, sub-protocols, metrics/SPF, areas, LSAs
- protocol headers
- LSA formats
- security
- summary and study questions

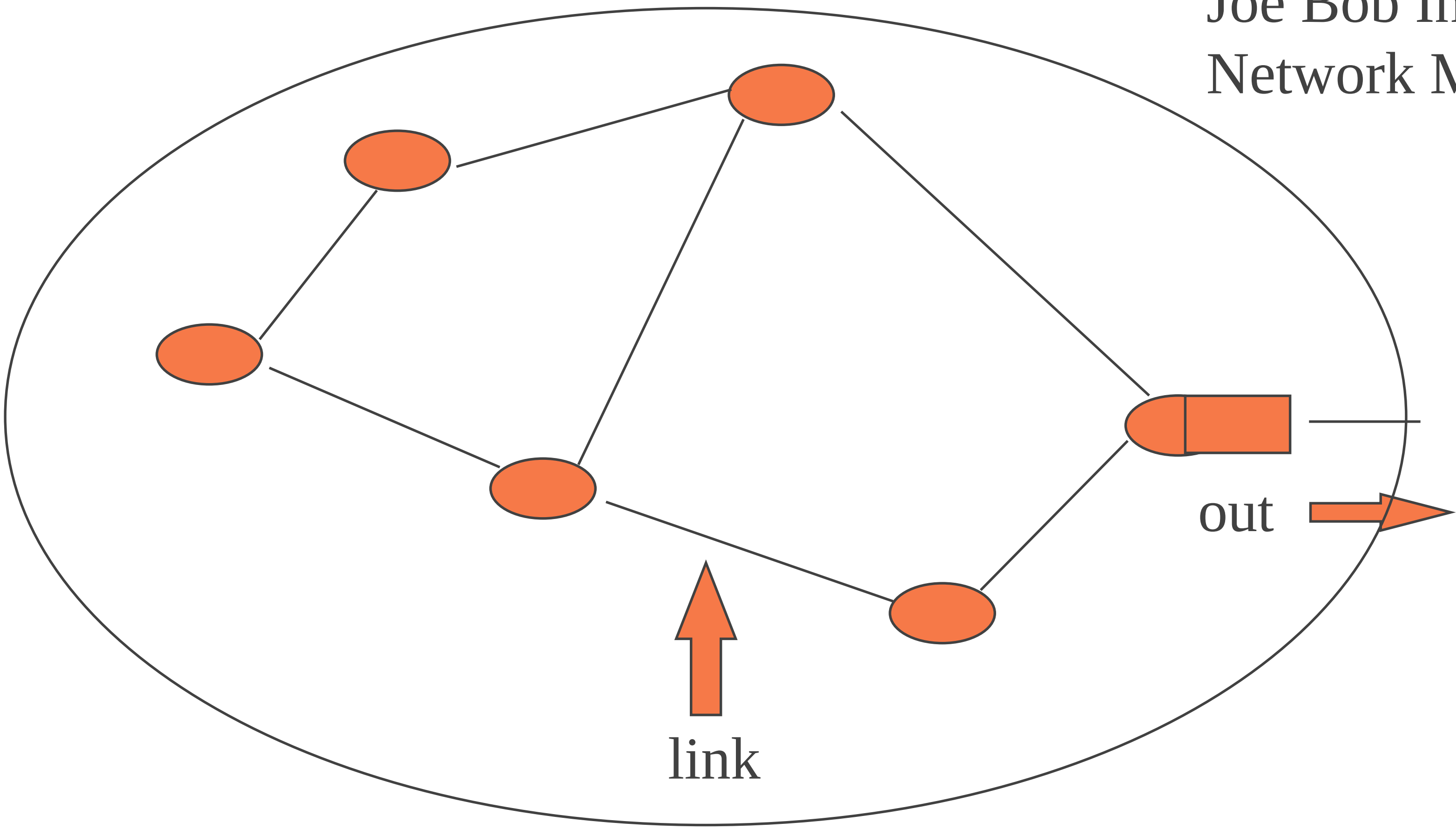
divide routing world into 3 parts

topology	IETF	ISO/OSI
same “link” or wire	none, intra-link?	none, intra-link?
enterprise or campus	Interior Gateway Protocol - IGP	intra-domain routing protocol
between enterprises	Exterior Gateway Protocol - EGP	inter-domain

protocols acc. to topology

topology	IETF	ISO/OSI
intra-link	ARP	ES-IS
intra-domain	RIP, RIP(2), **OSPF**	IS-IS
inter-domain	EGP, BGP(4)	IDRP

the Interior - RIP or OSPF



Bibliography

- RFCs of interest: (others exist, e.g., MIB)
  - **J. Moy, OSPF Version 2, 2328, 1998**
  - 2154, OSPF with Digital Signatures (experimental)
  - 2740, OSPF for IPv6, R. Coltun, et. all, 1999
- books:
  - Moy, OSPF
  - Huitema, Routing in the Internet, c. 6
    - » "Why Is OSPF So Complex?"

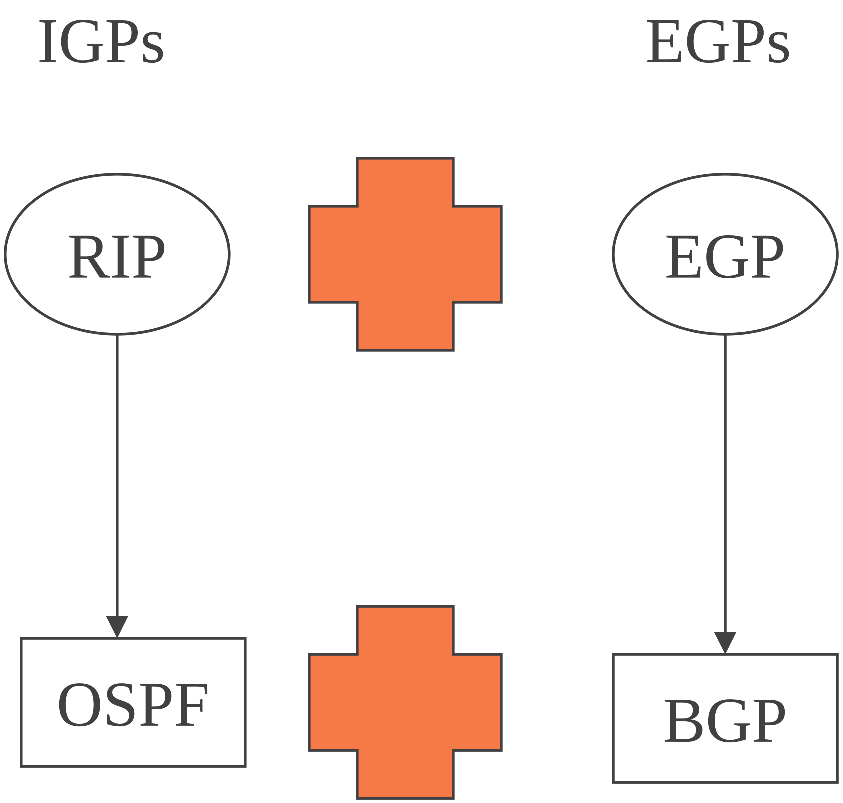
History (also Herstory)

- Link-State protocols developed early on in history of ARPANET (late 70's) (1st DV, then LSP by BBN)
  - distributed map idea
  - reaction against DV ideas (or at least RIP)
- ISO protocol suite developed IS-IS
  - IETF attitude was IS-IS == 0, not totally fair to ISO work
  - OSPF IETF IS-IS cousins and IS-IS predecessor
- Perlman suggested how to make flooding robust
- OSPF v1 formulated, but not deployed
  - problems with distributed link-state database
- v2, RFC 1247, 1991, note v1 didn't happen

herstory, cont. (IS-IS is used)

- Moy in RFC 2328:
  "A link state algorithm has also been proposed for use as an ISO routing protocol. ... The OSPF Working Group of the IETF has extended this work in developing the OSPF protocol".
- note that due to existence of a good vendor implementation of IS-IS that speaks IP, there exist AS out there that use IS-IS with IP addresses
  - as opposed to CNLP ISO addresses (20 byte var. length)
- IDPR - link-state EGP ... contention exists about whether it might replace BGP? not hop by hop, sophisticated policy routing possible

pictorial routing evolutionary history (started with NSFNET)



not everybody sees it this way

add CIDR in 90s, therefore BGPv3 to BGPv4

if you don’t do OSPF, what other choices are there in IGP land?

- IS-IS (aka Integrated IS-IS), on Ciscos
- EIGRP (DV++) from Cisco
- RIP (v2 hopefully)
  – v1 doesn’t speak CIDR
  – Cisco’s IGRP (view as RIP++)
- static routes of course
- are IGPs ever used as EGPs?
  – do layering violations occur in network stacks?

OSPF terminology (from RFC)

- **AS** - autonomous system, assume a group of centrally managed routers under one administrative control (has IP EGP meaning too of course)
  - aka routing domain
- an AS runs an **IGP**
- **Router ID -** 32 bit number assigned to each router running OSPF (guess which #?)f
  - must uniquely id router

terms

- **network -** IP number/netmask pair; therefore subnet (or supernet)
- **networks come in several kinds** acc. to OSPF
  - broadcast or not (come back to this)
- **interface**
  - on a router, aka port, aka link but let’s reserve that for the “wire”
- **neighbor routers**
  - two routers with a common link, formerly common network however (distinction is important)

terms

- **adjacency -** a relationship formed between two neighbors for exchanging/sync of LSA database info on interface reboot
  - not all neighbors form adjacencies
  - optimization here basically for broadcast networks (which have DRs and BDRs)
- **designated (and backup designated) router**
  - broadcast net with 2 neighbors has elected DR that generates LSA for that net
  - reduces numbers of adjacencies, therefore domain more scaleable, less routing overhead

more terms

- **area** - OSPF supports optional hierarchy
  - more or less a set of routers directly exchanging LSAs
  - LSA flooding limited within area
  - 2 level hierarchy, area 0 at top, and other areas (with area number, say 51 (of course)) underneath
- **LSA** - link state advertisement, describes routers (routes) with a given link, LSAs are
- **flooded** - which is how distributed map is created
- **hello protocol** - how routers on a given network determine set of routers, and build LSA

even more terms

- LSP - ISO for LSA - OSPF says advertisement
  - packet as opposed to advertisement
- areas may be **transit** or **stub**
  - transit means pkts cross area but do not originate from area
- more terms
  - set of LSAs (LSAs have types)
    - » example: **AS-external LSA**
    - » can potentially add new ones to grow OSPF functionality
  - routers have OSPF functions as well
    - » example: **ASBR**

OSPF network types

- layer 3 does not want to be layer 2 specific
  – and layer 2 can be weird and wonderful
  – especially the telco layer 2s
  – therefore OSPF has several link models
  – this model effects exactly how
    » hello works (neighbor discovery)
    » database adjacency synchronization
    » how the link is represented in LSA terms

network models include

- **broadcast subnets** (DR)
- **point to point subnets** (e.g., no DR)
  - only 2 routers, 1 wire
- **NBMA**, non broadcast, multiple access
  - all routers must be fully meshed
- **point to multipoint**
- **virtual links** (later, part of area discussion)
  - regard as virtual point to point

details:

- broadcast
  - e.g., ethernet, network can do broadcast
  - hello will elect DRs
  - the network itself is an element in the LS database
- NBMA - similar to broadcast
  - must be fully meshed (all Rs have link to other Rs)
  - network that is not bcast capable; e.g., ATM
  - emulation of broadcast is done (therefore DR)
  - MAY do with frame-relay, PVC, but painful

details

- point to point
  - no point (apologies) in DR
- point to multipoint
  - e.g., used with frame-relay, PVCs ...
  - treated as set of point to point links, no DR
  - auto-discovery of neighbors MAY be possible

OSPF features include

- areas - hierarchy can be introduced to make more scalable
  - fundamental point is to limit reach of inter-area LSA flooding (can't cross from one area to another)
- equal-cost-multipath
  - if equal cost metric paths to a destination, traffic can be round-robined
- on broadcast network, multicast used as optimization
- area internals can be summarized with summary LSA (aggregation) with net/mask
- routing traffic can be authenticated
- external routes can be injected and/or tagged

features cont.

- CIDR is supported (of course)
  - aggregation
  - host route possible, mask is all 1s
  - default possible of course
- several kinds of areas including stub and NSSA (not so stubby)
- multicast routing LSAs exist (MOSPF)
- note TOS (type of service) (different metrics) feature exists NO MORE

basic ideas - review

- "tell the world about your neighbors"
- distributed map is key idea
- 1st - determine neighbors on link
  - Link State determined by hello packets
- 2nd - reliable flooding of Link-State info
  - to all routers, hence they have the complete map
- 3rd - use Dijkstra SPF to determine shortest path from self to all networks via metric

however OSPF is more complex

- DRs introduce (or prevent?) complexity
  - an optimization, to drive N**2 to O(N)
- really 3 protocols + SPF calculation
  - hello which does DR election as well as neighbor discovery (and adjacency determination)
  - database xchange (bringing up adjacencies)
  - flooding of LSAs, which is **RELIABLE**
- the strange question of OSPF & metrics
- plus > 1 kind of LSA packet with many fields

theory overview

- LSA database
- flooding/sequence numbers
- hello/bringing up adjacencies
- metrics/Shortest Path First calculation
- areas/types of routers
- types of LSAs

LS database - theory

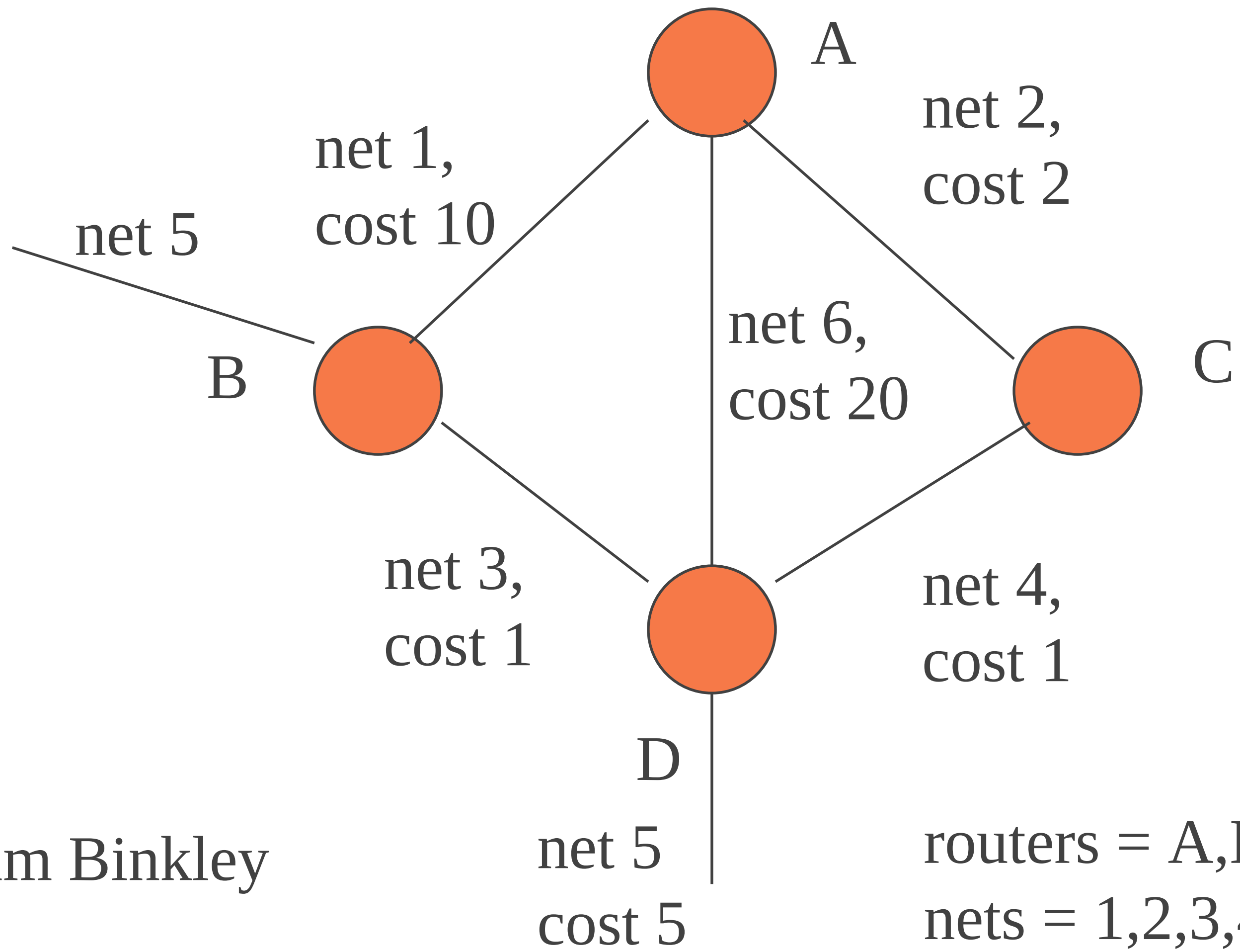
- assume point to point for following discussion
  - note with broadcast net, networks themselves are LS database entries
- the LS database consists of a set of LSAs flooded around the IGP domain
- each LSA has a cost (metric) associated with it, for now assume metric function is additive and f(x) is good when low (could be good when high)
- thus the LS database represents a directed graph for the IGP routing domain

and this point

- LSA has originator (one router with unique router ID)
- every other router in domain stores LSA in its LSA database
  - thus all have the same view
  - this is not quite totally true, as areas exist to contain LSA flooding
  - therefore true for routers in same area

theory - the LS database

consider the following set of routers + nets



routers = A,B,C,D
nets = 1,2,3,4,5 (external), 6

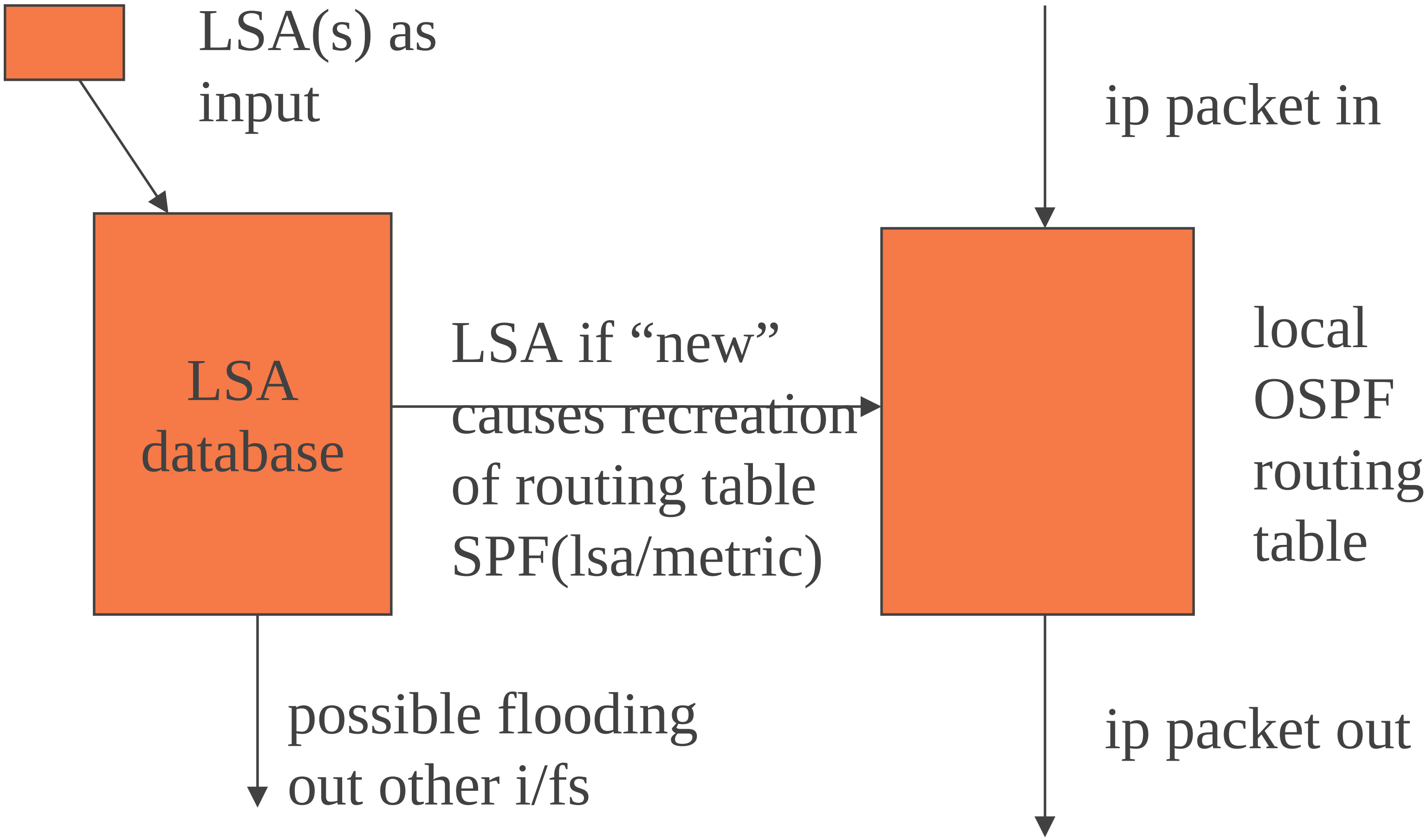
when state == CONVERGED

- each router has database with all LS records
- assume LS records are per net; e.g., A has:
  - A to B, net 1, cost 10
  - A to C, net 2, cost 2
  - B to D, net 3, cost 1
  - C to D, net 4, cost 1
  - D, net 5, cost 5
  - A to D, net 6, cost 20
- A can therefore calculate using SPF a routing table that is f(metric assumption, database)

A’s resulting routing table

- to B via C, cost is what?
  - what happens if C goes down?
- to C via net2, cost 2
  - what happens if A’s port to C blows up?
- to D via C, cost is 3
- to net 5 (outside), via B, cost 8
  - could have more than one way to outside
  - external routes may have different weights

there exists a LSA database, and there exists a routing table



Jim Binkley

flooding

- note that routers or interfaces may fail
  - interface UP or DOWN
    - » a router can determine its own link has failed
    - » or a neighbor may determine that a router has disappeared
    - » these events can drive LSA generation
- note that interfaces have a state machine associated with them
  - complicated by DR election, adjacencies, hw knowledge events (link is down)

flooding algorithm basics

- flooding is reliable per link
- if A/C net fails, A will notify other two links
- B e.g., will tell D but will NOT tell A (don’t send it back thru input i/f)
- B will add message to its DB and recompute routing table iff
- LSA is more recent, not corrupt, known type
- updates would cross from B to/from D, but D would not in turn then forward the pkt to A

flooding mechanics

- protocol includes per link ACK
  - resend until ACK heard therefore reliable
  - ACK is optimized in several ways and e.g., not sent when updates cross
  - recv may delay in hopes that ACK (may be unicast or multicast) may include multiple ACKs
- we need checksum/sequence # pair as well
  - sequence number must have “overflow” technique

checksum/sequence

- all OSPF packets include checksum and other robustness features in face of errors, hdr has IP csum, LSA has csum too
- OSPF does not use spanning tree, but floods which is inherently redundant
- router might accidentally delete LSA, therefore **originator** must refresh LSA on 30 minute basis
- pkt discarded if csum fails, checksum not altered by others, (LSA csum excludes age field)
- 3 tuple for freshness (csum, sequence number, age #)
- every router increments age, hence like IP TTL
  - discard at MaxAge

freshness, robustness, etc.

- rate limit LSA origination, at most 1 per 5 secs
- router periodically verifies LSA csums in DB. guards against internal memory failures
- originator sends (checksum, seq+1, age=0)
- if stored in other R db, age is incremented as it passes through, and over time by timeout function
- if 1 hour passes, and no resend, then LSA is tossed (why wait 30 minutes?)
- sequence space WRAP is velly tricky ...

sequence space wrap

- in ARPANET, LS protocol had famous sequence # failure
  - in theory Sn+1 > Sn, but unfortunately S1 > S2 > S3 > S1 happened
  - entire network had to be power-cycled
- v1 had lollipop algorithm
  - calculation still felt to be problematic
- therefore v2 **does not wrap** ...

v2 sequence idea

- we have reliable flooding, therefore originator reliably REMOVES LSA from domain, and regenerates it at wrap time
- S0 is InitialSequenceNumber, max negative, in hex 0x800000001,
- increment by one until 0x7fffffff, but 1st
- flood deletion with S(max), then send S0
- in theory, 600 years of time ... but errors could occur

hello/bringing up adjacencies

- hello is neighbor discovery packet
- therefore has these functions
  - link operational (peers exist)
  - elect Designated R and BDR on broadcast links
- hello sent at default 10 seconds
- on write sent to 224.0.0.5 (all-SPF-routers)
- list of neighbors are included (i can hear you)
  - basically this is an ACK, link must be bi-directional
- routerDeadInterval, 40 seconds - must hear from neighbor within this time, else route around

hello, cont.

- decide link is operational iff
  - other guy has you in its hello
  - if pt/pt, that is enough
  - if broadcast, must wait for DR election
- election algorithm ideas:
  - priority field and IP address used as discriminators
  - highest priority wins, if > 1 with same priority, highest IP wins
  - always keep DR and BDR, if DR fails, BDR is DR

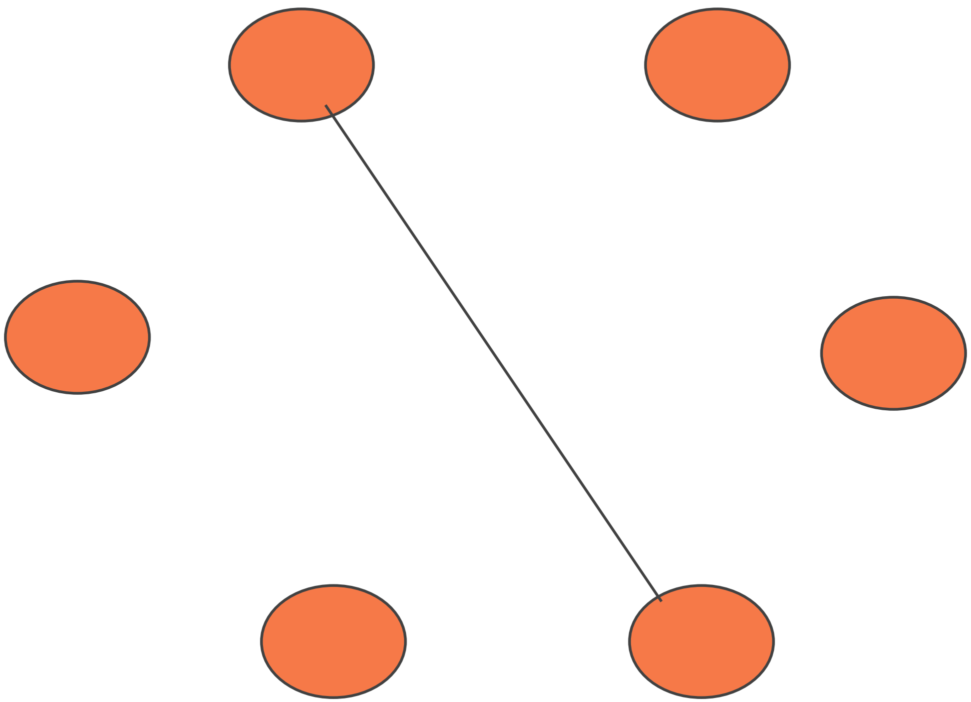
election algorithm roughly

- if more than one BDR, choose based on 1. priority/2. high IP address is tiebreaker
- if no backup, choose based on priority/IP
- if > 1 DR, choose based on priority/IP
- if no DRs, and BDR, promote BDR
- key idea: DRs and BDRs must do database exchange with all other routers on subnet
  - **non DR is adjacent to DR**

how many relationships on this bcast net?

6 routers, N * (N-1) / 2

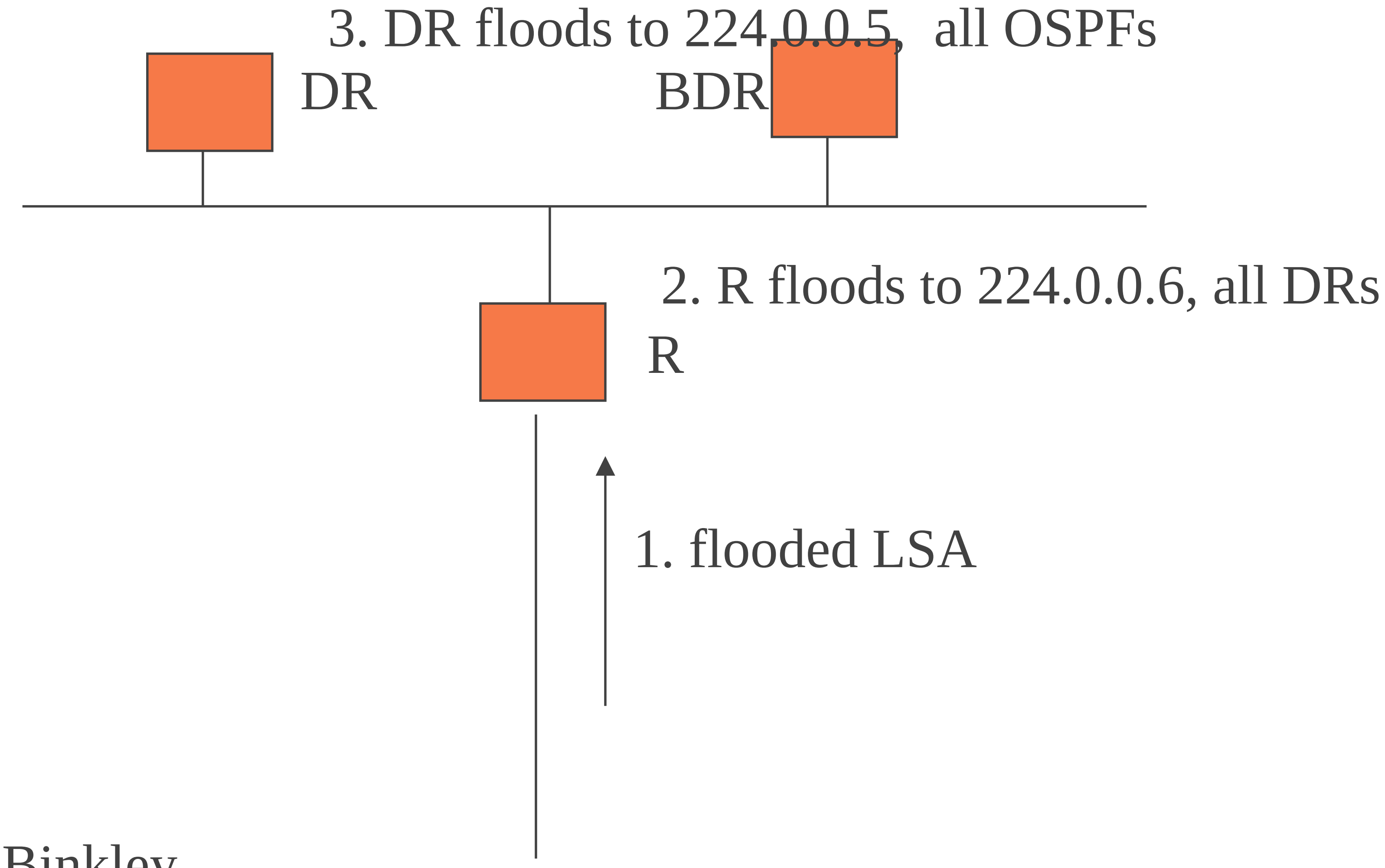
N = 6



DR points/are these

- non DR routers keep LSA databases in sync with DR using
  - database exchange (I booted, give me all you got)
  - reliable flooding
  - single point of failure, therefore BDR is hot standby
  - routers must sync with BDR too
  - this makes complexity linear

flooding with DRs then



Jim Binkley

database sync

- could come from LSA flooding alone
- we MUST keep routers in sync with LSA maps
- else we risk routing loops, black holes
- optimization: at boot, exchange map with adjacent router, or do this at partition fixup
- call this **database exchange**

aka

- **bringing up adjacencies ...**
- one of 3 sub-protocols in OSPF
- 1. hello
- 2. bringing up adjacencies (db exchange)
- 3. reliable flooding (fun with LSAs)

database exchange

- basically adjacent peers exchange headers only, determine if LSA needed
  - then ask for new LSA and get it
  - database description exchange resembles TFTP, only one outstanding, must be ACKed
- database exchange done after hello sync
- always done with pt/pt, on broadcast done with router to DR (e.g.), not 2 non-DRs

exchange protocol idea - overview

- 1. at top level, 1st 2-way exchange of hellos
  - hello from you must have me in it
- 2. then we have reliable exchange of database description
  - Master/Slave role with ACKS
  - note ACKs can have LSAS for slave
- 3. then each router sends Link State Request for LSAs that are new
  - gets back Link State Update with LSAs

exchange protocol, part 1

- one router decides it is master, sets M bit
  - 2nd router becomes Slave
  - or if tie, and waiting for ACK, and other party claims SHE is master, choose acc. to highest IP
- DD pkt has DD sequence number, contains some number of LSAs (with LSA seqno)
- master sends SEQ N, slave sends DD SEQ N, will include slave LSAs
- this is ACK, if I don't get it, resend
- do this, until all headers exchanged

part 2, exchange LSAs

- send OSPF LSA request, which may include multiple LSAS needed
  - LSA ID includes LSA sequence number
- send OSPF LSA update for LSA that the other party actually wants
  - this is more or less, ordinary flooding, but can obviousally include multiple LSAs of interest

metric/routing table calculation

- OSPF metric theory:
  - assume single metric and not dynamic
  - metric must be integer 1..64k (16 bit LSA field)
  - metric in theory OPAQUE; ideal is that admin decides and might have choices: (implementations!!!)
  - must be additive, smaller the better (acc. to Moy)
  - e.g., might be hop count, delay, mumble mumble
  - OSPF MIB suggests transmission time
  - metric is used in routing table calculation (doh!)

Cisco metric reality

- we weight the numbers to make bigger thruput better
- e.g., if the fastest link is 100BASE ethernet, choose 100,000,000, therefore
- 100BASE ethernet has weight 1
- 10BASE has weight 10
- thus, choose 100BASE over 10BASE
  - RIP can’t do that

Cisco metric reality

link	metric
100mbit	1
10BASE enet	10
T1	65
64k modem	1562

SPF algorithm considerations

- SPF computation initiated by ANY change in LS database
- view result as either:
  - a database of possible paths from self to dest X
    - » we do need equal cost multi path
  - a rooted tree of best paths from you to everybody else
    - » we will think about it this way

E. Dijkstra algorithm

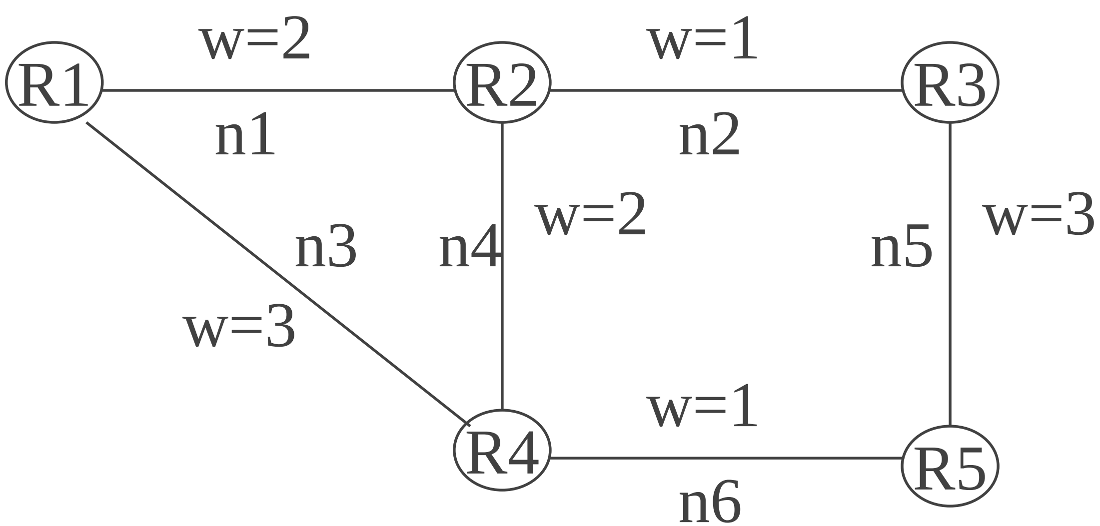
- input: directed graph (the LSA DB) with links having weights
- the SPF algorithm calculates a tree of shortest path (define short as least weight) from self to all others
- we look at each destination once
- we keep a candidate list that is sorted by weight
- we take the best (shortest) value in the candidate and put it in the routing table
- we may modify and resort the candidate list as new LSAs are found (we look at all LSAs)
- IP routing table needs only next hop, LSA tree has all paths

simplified howto

- you have routing table (final output), you have candidate list (working set), you have set of LSAs
- 1. pick one node (directly connected) (start with self)
- 2. place that nodes links in the candidate list
  - always keep sorted by weight
- 3. take best candidate router
  - and put in routing table, go to 2

exercise: perform SPF on this domain

how can we track equal-cost multipath?



e.g., start with R2

e.g., 1st iteration

- pick r2, puts it links in candidate list then
  - to R1, n1,w=2
  - to R3, n2,w=1
  - to R4, n4,w=2
- add R3 to routing table, next hop to n5
- add R3's links to candidate table and sort
  - to R3, n5,w=3 (and mod this weight)
- when add LS to c list, mod weights to reflect path out from R2
- also note ECMP case, w=2 2 times from R2 to n3

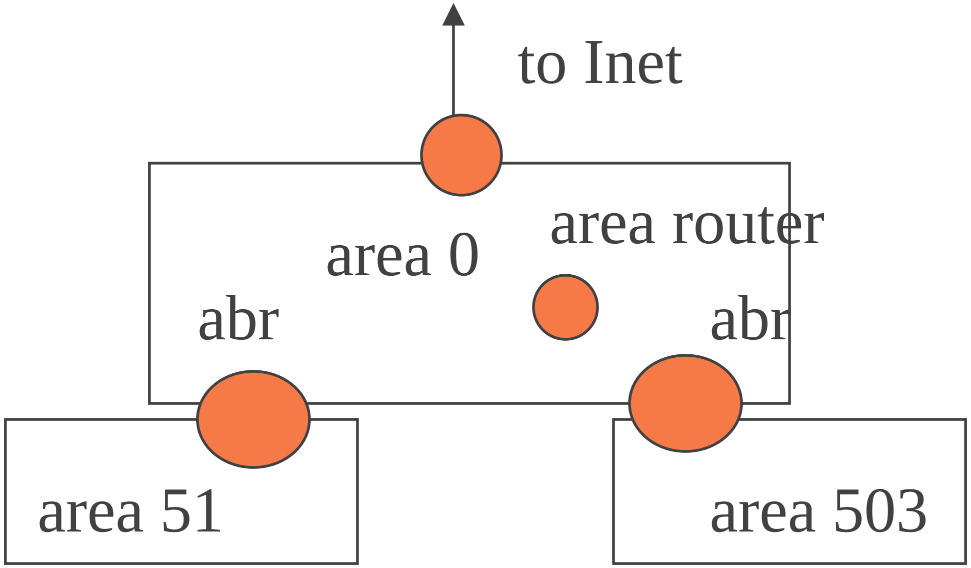
algorithmic complexity

- shortest path is links * nodes * log node count
- we keep candidate list sorted, therefore toss log node
- if we have DR, we have one node elected for N nodes on link, and can therefore further optimize # of LSAs sent
- this gives us more or less: N log N, where N is # of nodes
- on paper, Bellman-Ford is N2, SPF may be better depending on net topology

areas

- OSPF can have optional hierarchy, areas
  - 2 levels only
- must have backbone area, area 0
  - level 2 in ISO speak
- interface must belong to area, router can be ABR or Area Border Router
  - 2 i/fs in different areas
  - if all i/fs in same area, then ordinary area router

areas



hint: view areas as hub and spoke design

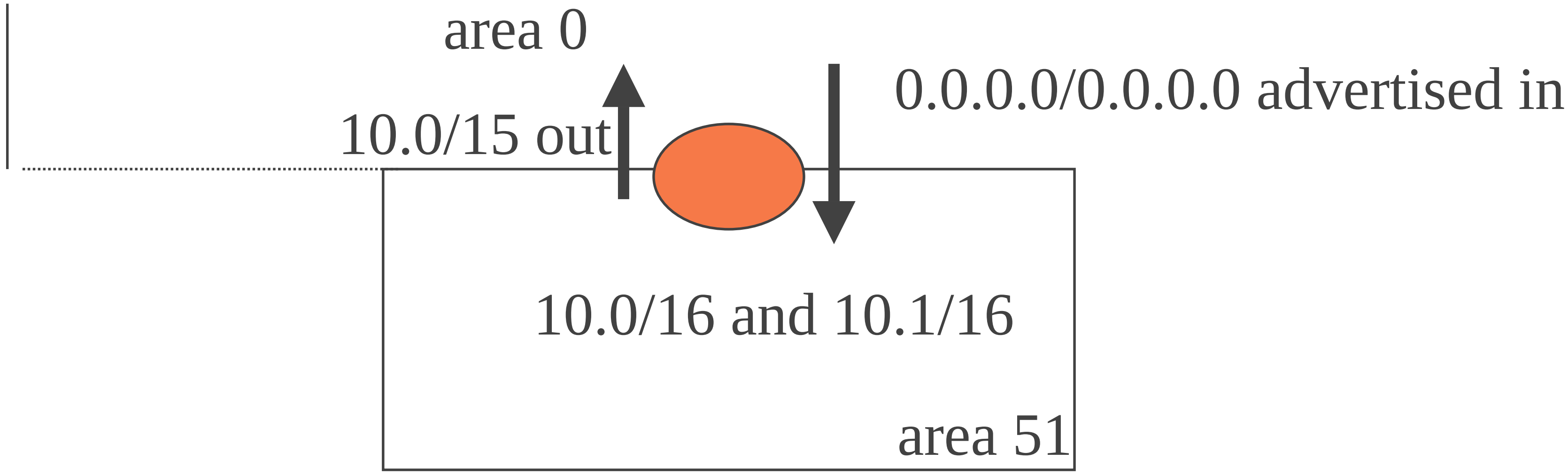
why bother?

- **scalability** if many routers, many LSAs
  - areas can **limit LSA flooding**
  - ordinary LSAS stay within area (router and net LSAs)
- the latter point may be useful for **reliability/redundancy**
  - contain other administrations mistakes ... LSAS you don’t want or need - they do cause SPF to happen in your routers
- ABRs can aggregate routes in/out of area
  - summarize routing table as opposed to individual nets

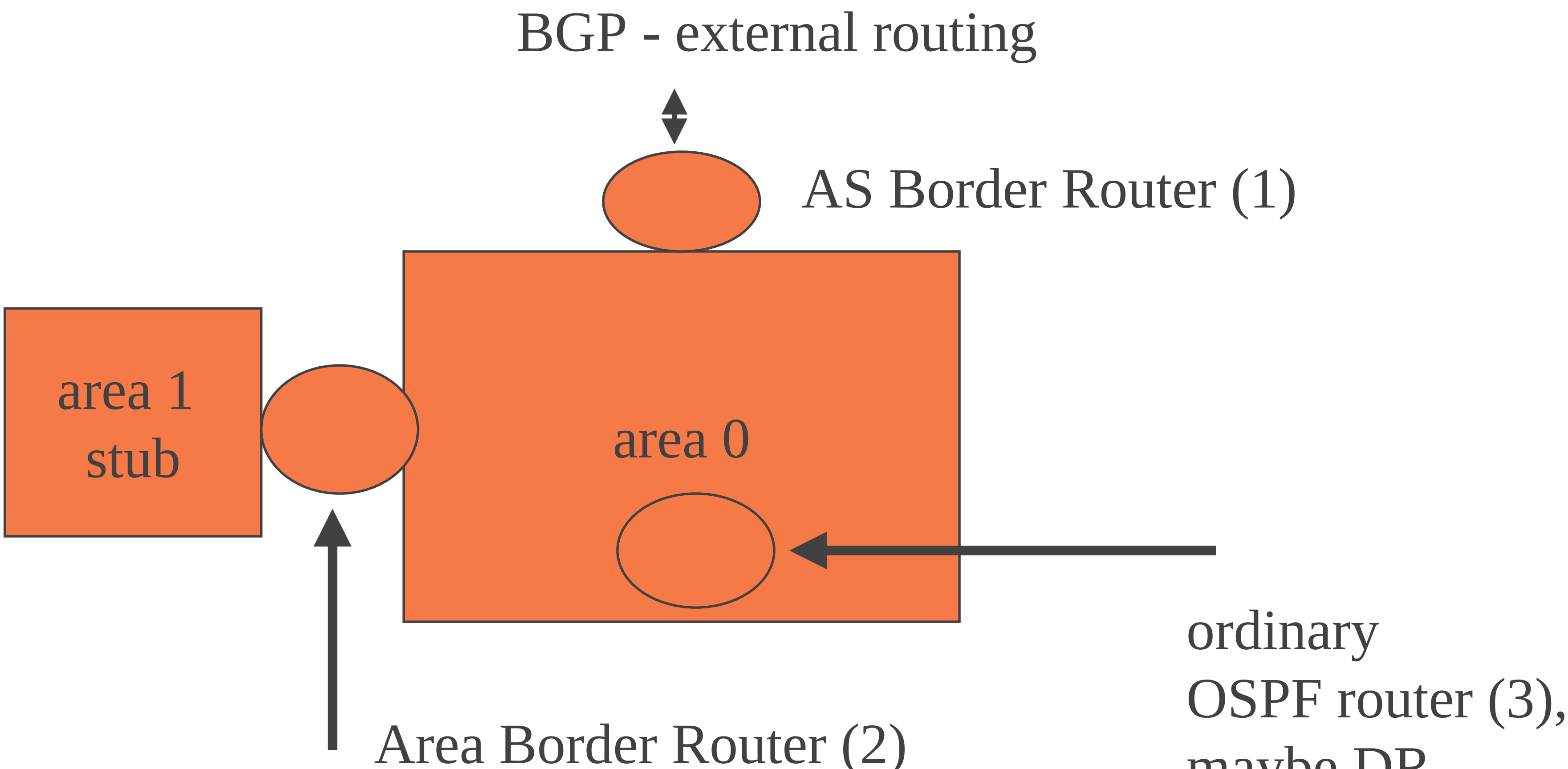
assume we have 10.0.0.0/8

- area 51 might have nets 10.0 and 10.1/16
- therefore the ABR could advertise
  - 10.0/15 into area 0
  - as opposed to many smaller subnets
- it might advertise the default route into area 51

area aggregation diagram



OSPF router types



router types then

- ABSR - OSPF router that may inject external routes
- ABR - area border router
- DR and BDR - designated routers
  - their LSAs are inter-area, not intra-area
- ordinary OSPF router (not DR)

virtual links

- as a 1st assumption OSPF sub-areas must physically connect to area 0
- however a “virtual link” can be used to tie a sub-area that is not contiguous to area 0
  - area0 --- area51 -- area666

    ←——————→
    virtual link

virtual link

- summary LSAs are exchanged
  - two endpoints must be ABRs
- tell router 1: to router 2, across shared non-backbone area N, can't transit a stub area
- however routing of data pkts will (should?) bypass having to go to the backbone when that makes sense e.g., areaVL1 to VL2

areaVL1 --- not-backbone-area --- areaVL2

↕

backbone

virtual links

- are manually configured
  - treated as unnumbered pt. to pt. i/f
  - cost is sum of internal transit links
- adjacency relationship established
  - called **virtual adjacency**
- AS-external-LSA not sent over VL as this info arrives via the transit area
- may be used to repair a network partition
- think of them as like an IPIP tunnel
  - but not actually implemented that way

types of LSAS (wake up)

- 1. **router-LSA**, per router, describes active neighbors and own i/fs
  - note: if pt-pt, we do not send network-LSA
- 2. **network-LSA**, describe net segment on broadcast net (for the most part)
  - sent by DR, list of routers on that net
  - 1 & 2 are fundamental flooding LSAs
- 3. **network-summary LSA**
  - ABRs eg., advertise to/from areas
  - default route generated for stub area

more LSAs

- 4. **ASBR-summary LSA**. ASBRs advertise internally how to get to them. note the point here is that this LSA uses the internal OSPF metric.
  - only flooded intra-area, format same as #3
  - note, 3,4,5 are all about hierarchical routing
- 5. **AS-external LSA**. describe external routes to internal areas (e.g., BGP external route into OSPF)
  - not internal metric, but outside dest X this way
  - flooded through ALL areas, intra-area, except
  - **stub areas do not take these**

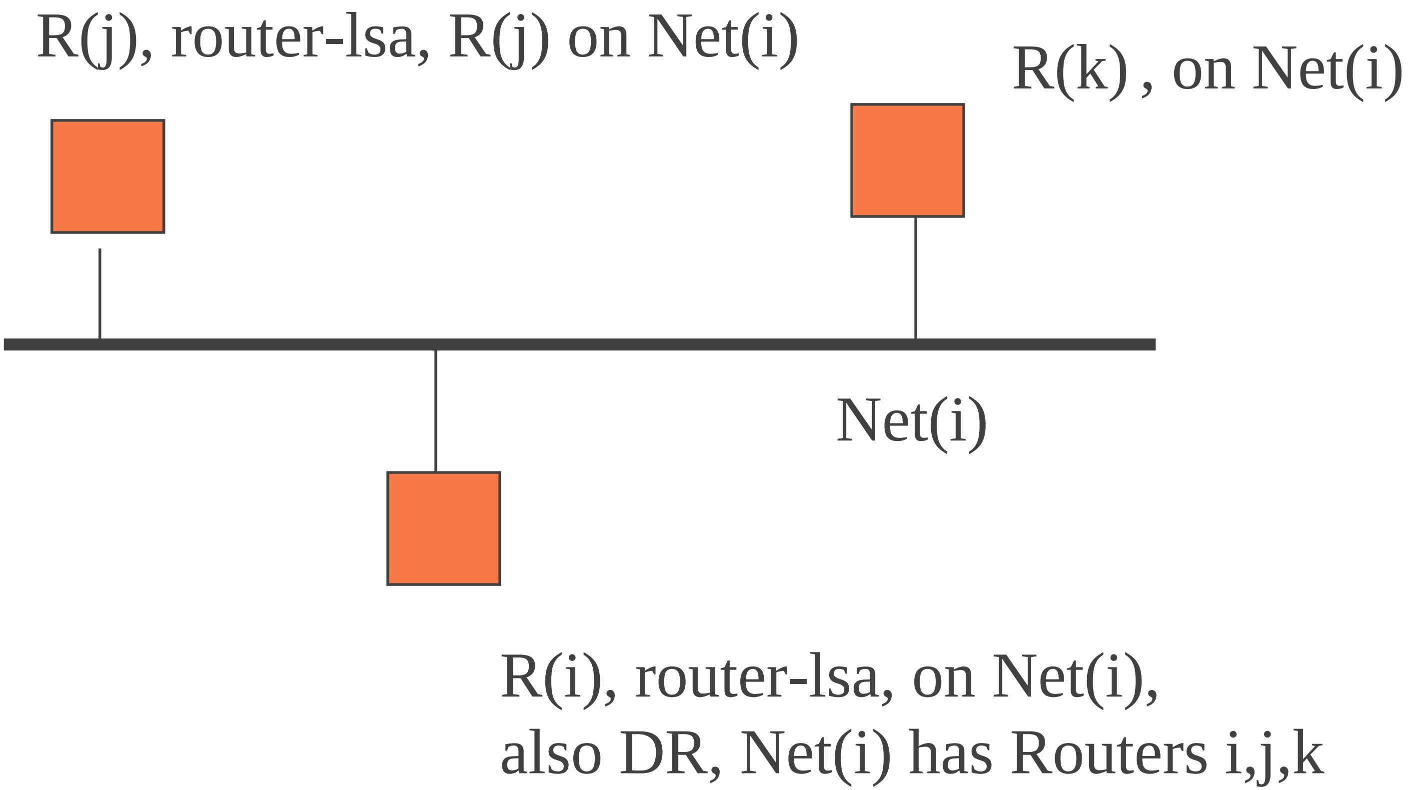
more LSAs

- 6. group-membership LSA, used in MOSPF to flood existance of multicast group
- 7. NSSA area import (later)
- 8. may be more ..., if we have some piece of info that needs flooding (reliable!!!)

why router/network LSA?

- if no DR, no net LSA, router-lsa would include links to all routers on network
- remember: net N might have many routers
- each router i would have a link to router j
  - j to i, etc.
- optimization: network LSA lists routers
- routers list networks ... therefore N * 2, not N * N
- DR originates network LSA, all routers originate router LSA

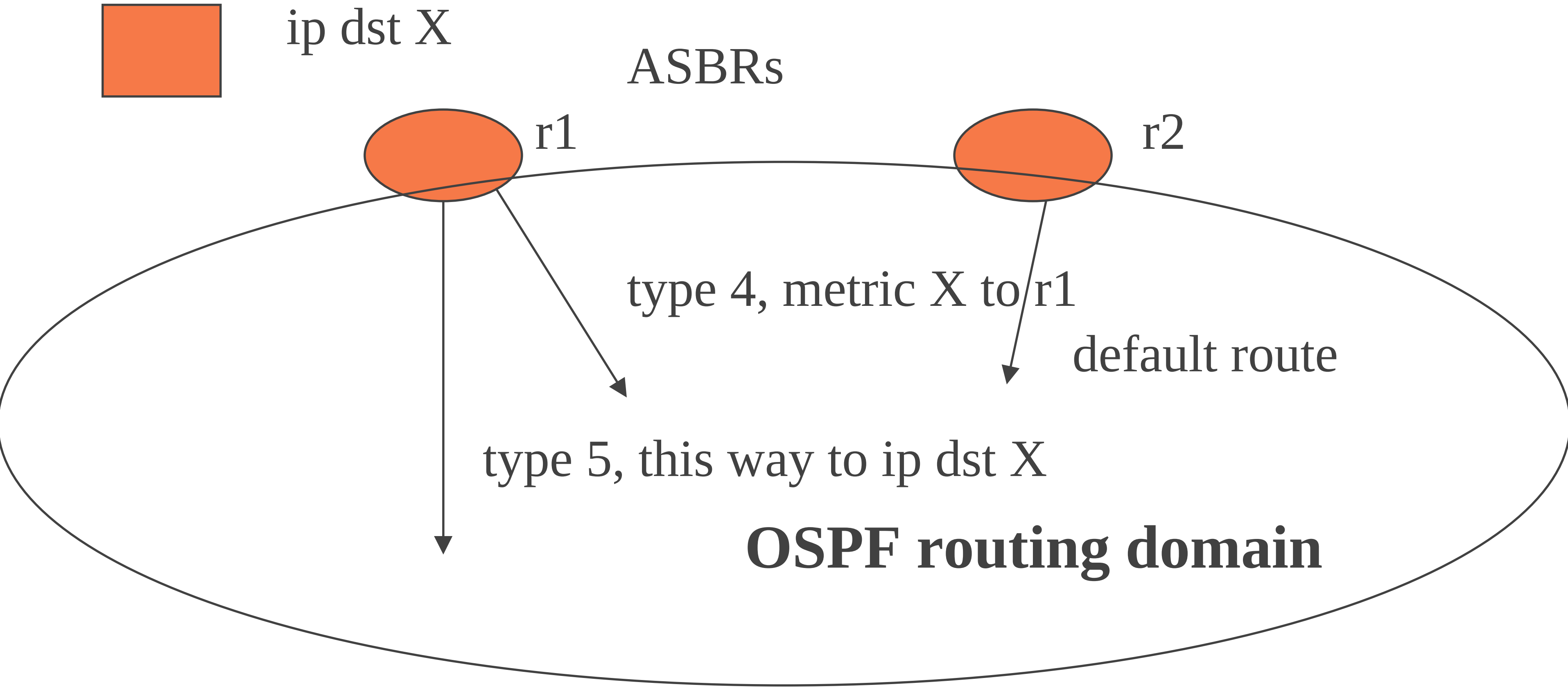
broadcast net, therefore



summary LSAs

- 3,4,5 all deal with areas
- 3 for area aggregation
- 4,5 for routing info needed for routing domain external routes
  - 4 says how to get to ASBR
  - 5 says here is a route beyond the ASBR/s
  - keep in mind possible > 1 ASBR

multi-homed routing domain



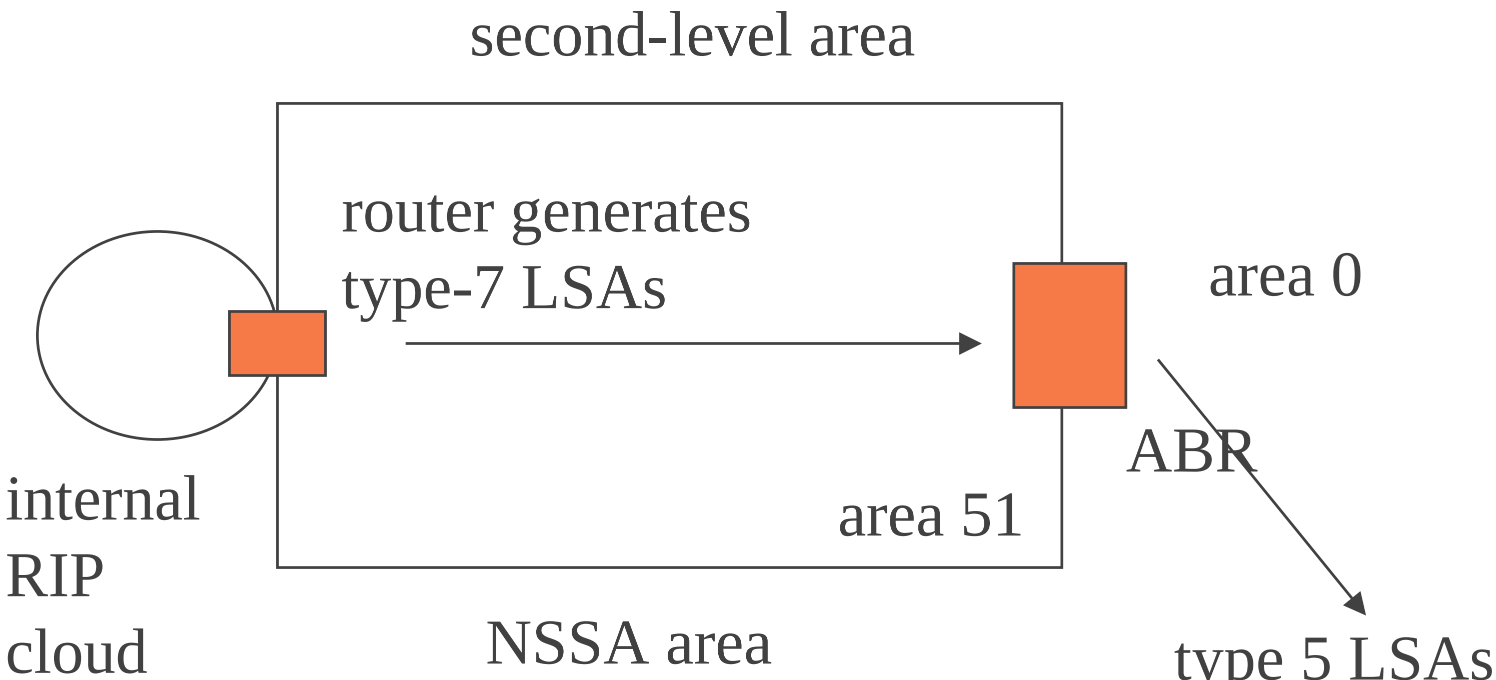
types of areas

- ordinary joe bob area (this is about stub areas really, so this is NOT a stub area)
- stub area
  - no transit traffic, no virtual links
  - does not accept external LSA
  - only one way out
  - consumes least resources
- not so stubby area (NSSA)

NSSA - not so stubby

- assume stubby, but one change
- type 7 NSSA lsa can be used to export NSSA internal routes
- type 7 has area scope
- translated at ABR to type 5

why NSSA diagram?



OSPF protocol

- OSPF uses IP direct, not on top of UDP, IP proto = 89

ethernet	ip p=89	OSPF pkt hdr, etc.

OSPF packet types

- all have common 24 byte pkt header
- 5 distinct pkt types
  - 1 hello, 2 database description, 3 link state request, 4 link state update, 5 link state ACK
- all but hello may be viewed as LSA lists
  - link state update is flooded
  - database description used in bringing up adjacencies
- LSA itself has its own structure

common OSPF protocol header (24 bytes)

version	type	pkt length
router ID		
area ID		
IP checksum		auth type
64 bits of authentication		

pkt header fields

- router ID - typically an IP address
- area ID - area this packet belongs to
- checksum - IP checksum for all bytes in packet, does not include authentication, may be absent for some authentication types if redundant

hello packet (type = 1)

common pkt hdr = 24 bytes ...		
network mask		
HelloInterval	Options	Rtr Pri
RouterDeadInterval		
DesignatedRouter		
BDR		
1 of N Neighbor IDs ... (variable length)		

a few hello details

- OSPF multicast addresses:
  - 224.0.0.5 - all SPF routers (I speak OSPF)
  - 224.0.0.6 - all DR routers
  - note 224.0.0.5 is enet 01:00:0e:00:00:05
- bcast hello time - 10 seconds
- bcast dead time - 40 seconds
- IP addr (routerID) and priority used in DR election
- note if local OS can tell you link is down, use that else 2-way exchange can tell us

more details

- ip ttl = 1
- dest ip = 224.0.0.5
- DR/BDR values, 0 means none yet
- Neighbor IDs are IP addresses

DDescription packet (type = 2)

common pkt hdr = 24 bytes ...			
0	0	options	flag bits
DD sequence number			
Link State Type			
Link State ID			
Advertising Router			
Link State Sequence Number			
checksum		age	

1 of NLSA.hdr

request packet (type = 3)

common pkt hdr = 24 bytes ...
LS type
Link State ID
Advertising Router
more LSAS, specified by 3-tuple (type, ID, advertising router) ...

note: we do not specify instance, we assume we want most fresh LSA

update packet (type = 4)

common pkt hdr = 24 bytes ...
of LSAS
LSA #1 (with LSA hdr/body)
LSA #2
more complete LSAS ...

note: this is standard flooded LSA, LSAs are complete

Link State ACK, type = 5

- may be sent to all-spf-routers or all-DR-routers or unicast for that matter
- format similar to DD packet
- type 5, with OSPF hdr first
- followed by 1..N LSAs headers, which must include ACK'ed instance
- may be slightly delayed in hope that ACKs will be more cumulative
- may use unicast to fast ACK DUP LSA

LSA formats, 1st global header

header followed by per LSA info
this is just an LSA, not a OSPF packet

LS age	Options	LS type
Link State ID		
Advertising Router		
LS sequence number		
LS checksum	length	

LSA header details

- key for LSA is (type, LS ID, advert router)
- types are 1-5 for basic LSAS (router/network, area summary, etc)
  - > 5 for extended LSAs
- advert router, who originated LSA, note may or may not be same as Link State ID
- sequence number - inc if LSA fresh
- LSA csum, fletcher (ISO), not IP
- length, includes LSA hdr, must fit in IP pkt
- age, 0 when 1st sent

LSA link state ID

- associated with type
- type 1, originating router ID
- type 2, IP of i/f of network DR
- type 3, destination net IP addr
- type 4, router ID of ASBR
- type 5, destination net IP address

router-LSA summary info

- router X
  - has separate links for interfaces
  - e.g., 3 links
  - each of which mentions a network
  - and metric on that network
  - all router interfaces must be mentioned

type 1 LSA, router-LSA

LSA 20-byte header ...		
bits including VBE		# of links
Link ID; e.g., pt/pt, then other guy		
Link Data		
net type	TOS=0	16-bit metric value
more possible link tuples here		

router-LSA notes

- intra-area only, LS ID is router ID
- bit flags, V means router is VT endpoint
  - B, ABR, and E ASBR
  - note this describes routers hierarchical role
- links, links router has in area
- types mean i/f type
  - pt./pt., transit network, stub network, virtual
- link id depends on type
- TOS if 0, then default, if non-zero then backward compatible, only one as > 1 TOS not done

link IDs

- type 1, neighbor router router ID
- type 2, IP address of DR
- type 3, IP network/subnet number
- type 4, neighbor router router ID

type 2 LSA, network

LSA header followed by N routers
note Link State ID is DR IP

LSA 20-byte header ...
network mask
attached router ID # 1
attached router #2
more attached routers ...

type 3,4 summary LSA

used by ABRs or ASBR, intra-area only
may advertise default route in stub

LSA 20-byte header ...	
network mask	
0	metric
tos	tos metric
more mask/metric tuples ...	

type 5, external summary LSA

used by ASBR, intra-area only (no entry to stub)
may advertise default route as “type 2 external”

LSA 20-byte header ...	
network mask	
E bit & TOS=0	metric (24 bits)
forwarding address, 0 = none	
external route tag	

notes on external-LSA

- metric E bit if set, specifies type 2, else type 1 external route
- type 2 external - means this metric is more important than any internal metric; e.g., BGP path cost > OSPF internal cost
- type 1 external, external metric of same kind as internal
  - e.g., assume OSPF uses hop count
  - we import RIP metrics

external notes, cont.

- field **forwarding address**, set if we desire to route packets to somebody other than originator
  - this may help us avoid a hop going out OR fit in some other clever scheme (level of indirection)
- **external route tag** - not used by OSPF, might be used by something like BGP to communicate info across transit system

therefore OSPF has 4-level routing hierarchy, prefers

- 1. same area
- 2. across area
- 3. type 1 external better than
- 4. type 2 external

default route summary

- ASBR can generate type 5, external LSA into area 0
  - external type 2 metric
  - view as summary of external routes
- however this won’t help a stub area (or NSSA)
  - cannot take external LSA,
  - **needs type 3 ABR summary for default route**

OSPF security

- authentication, no confidentiality
- 3 defined forms of authentication
  - for all pkts, in pkt header there is auth. type
  - 64 bits of data for use by authentication scheme
  - types include:
  - 0 - NULL authentication
  - 1 - plaintext ASCII password
  - 2 - media digest (MD5) shared-secret authentication

authentication

- only the last form should be taken seriously
  - plaintext password can be useful to ignore “accidental” routers or packets from another admin. entity on shared network
  - sniffable obviously, active attack possible
- plaintext password
  - uses 64bit, 8-byte field
  - keep in mind checksum exists for OSPF pkt itself (not part of this functionality)

cryptographic authentication

- shared secret key (say 128 bits in hex for MD5) configured in routers
  - per network (as with password)
  - could of course be same key per domain
- message digest is appended at end of OSPF packet
  - but not formally part of packet
  - reader learns auth type from header, and using other info in header can suck in hash trailer

auth field with crypto authentication

64 bits

0	Key ID	auth data len
sequence number (not the hash)		

key id: ids algorithm used (e.g. MD5)
auth data len: how many bytes at end of packet
sequence number: unsigned 32-bit nondecreasing #
used to guard against active replay attacks

RFC 2154 - digital signature authentication for OSPF

- from TIS, 1997, Murphy, Badger, Wellington
- experimental protocol
- Perlman and IDPR both considered signing of LS information
- basic ideas:
  - 1. distribute signed router LSAs
  - 2. do other non-flooding with MD authentication
  - 3. be able to distribute public keys in an LSA
- 1 & 3 considered interesting here

rough: how it works

- each router in domain has private, public key pair and public key for Trusted Entity
- LSA is signed with usual mechanism (sign the MD) and append sig
- a priori per router public key (cert) must be shipped using new PKLSA (flooded) to all other routers (great idea)
- that key is verified with the public TE key
- TE must generate per router cert/sign it

OSPF summary

- pros:
  - fast convergence, LSA flooding is fast
  - low bandwidth, LSAs not flooded that often
  - flooding is POWERFUL routing design technique
  - more scalable than RIP!
  - metric like static throughput helps with heterogeneous links (gE, 100BASE, 10BASE ethernet)
- cons:
  - SPF calculation can be costly
  - very complex with lots of optimizations

study questions

- router to router addressability (how exactly do I talk to you?) is always a priori important, because "routing" may not exist before the establishment of IGP convergence. How does OSPF establish addressability?
  - in a broadcast domain?
  - in a point to multipoint domain?
  - with virtual links?

study questions

- outline any security attacks that might exist for each of the following OSPF authentication methods
  - 2.1 null
  - 2.2 ASCII plaintext
  - 2.3 message digest/shared secrets
  - 2.4 (extra credit...) OSPF with dig. sigs

study questions

- explain what a router-LSA might look like?
- why do we have router-LSAs and network-LSAs?
- explain the protocol exchange including hellos needed for bringing up adjacencies?
- what the heck is an adjacency anyway?

study questions

- compare and contrast the 5 basic LSA types
- explain the 5 basic OSPF types of messages
  – which have something to do with LSAs?
- compare and contrast the OSPF basic network types
  – what differences do broadcast networks bring with them?
  – what is a virtual link?

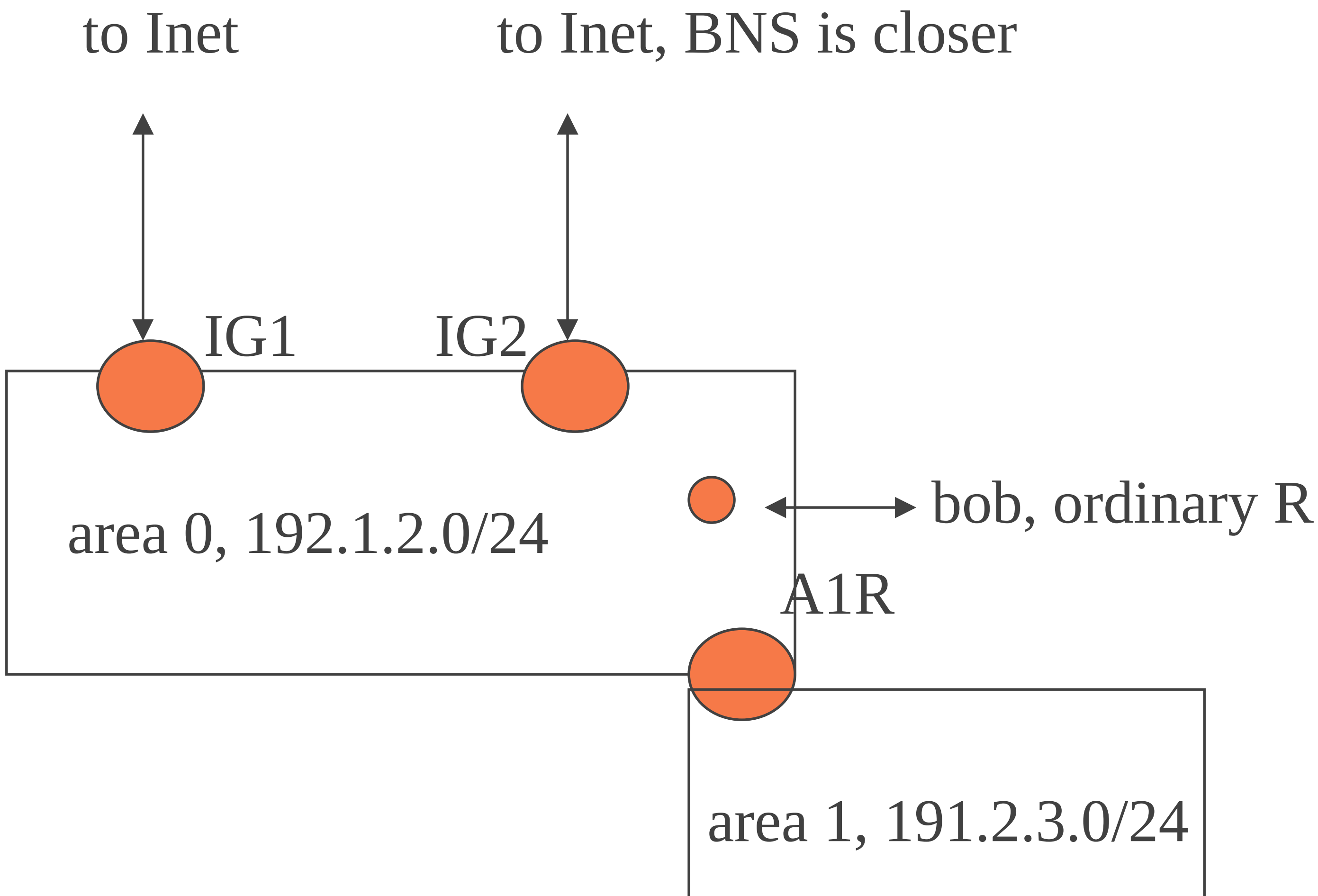
study questions (non-trivial)

- ok, you want to implement Mobile-IP as a local area/IGP kinda routing protocol
  – how could you take advantage of OSPF flooding? (btw, OSPF can handle host routes)
- is OSPF a good candidate for a mobile ad hoc routing protocol?
  – see if you can give one pro and one con

study question (see next 2 slides)

- assume we have a multi-homed stub network, and we are using OSPF
  BNS - big nearby school
  IG1, IG2, our Inet border routers, assume entire Inet routing table
  A1R - area 1 router, an ABR
- the AS has two class C subnets, that are not contiguous, 192.1.2.0/24 & 192.2.3.0/24. It has two OSPF areas, 0, and 1.

picture of network



study questions based on picture

- 1. what kind of LSAs do the 2 ASBRs inject into the OSPF domain?
- 2. name the routers that are ASBRs and ABRs.
- 3. what kind of LSAs does A1R send/recv?
- 4. what kind of LSAs do IG1 and IG2 recv from the area 0 routers?
- 5. add net 201.0.1.0/24 to area 1, what do you have to do to the ABR?
- 6. what kind of LSAs do Bob (not a DR), and Doris, (Bob’s DR) send/recv?