Network Layer:

Global Internet Routing (Interdomain, BGP)

Qiao Xiang

https://qiaoxiang.me/courses/cnns-xmuf21/index.shtml

12/07/2021

This deck of slides are heavily based on CPSC 433/533 at Yale University, by courtesy of Dr. Y. Richard Yang.

Outline

- Admin and recap
- Network control plane
  - Routing
    - Link weights assignment
    - Routing computation
      - Basic routing computation protocols
        - Distance vector protocols (distributed computing)
        - Link state protocols (distributed state synchronization)
      - Global Internet routing

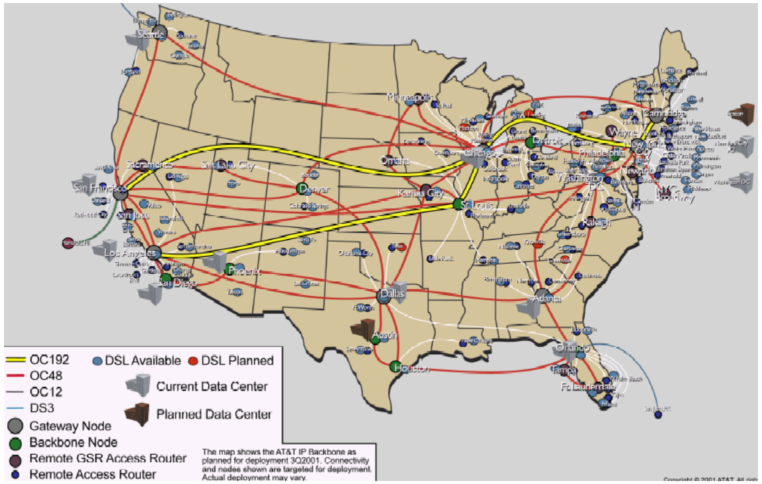
Admin

- If you haven't done so, please let me and TA know your project topic as soon as possible
- Final exam
  - Similar format as midterm
  - You still get to bring a cheatsheet
  - Cover lectures 16-27
  - 17 + 1 (bonus) question

Recap: Routing Context

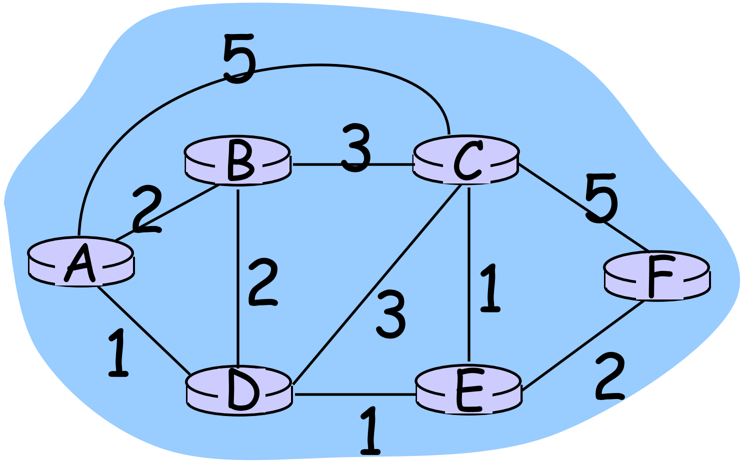
Routing

Goal: determine "good" paths (sequences of routers) thru networks from source to dest.



Often depends on a graph abstraction:

- graph nodes are routers
- graph edges are physical links
  - links have properties: delay, capacity, $ cost, policy

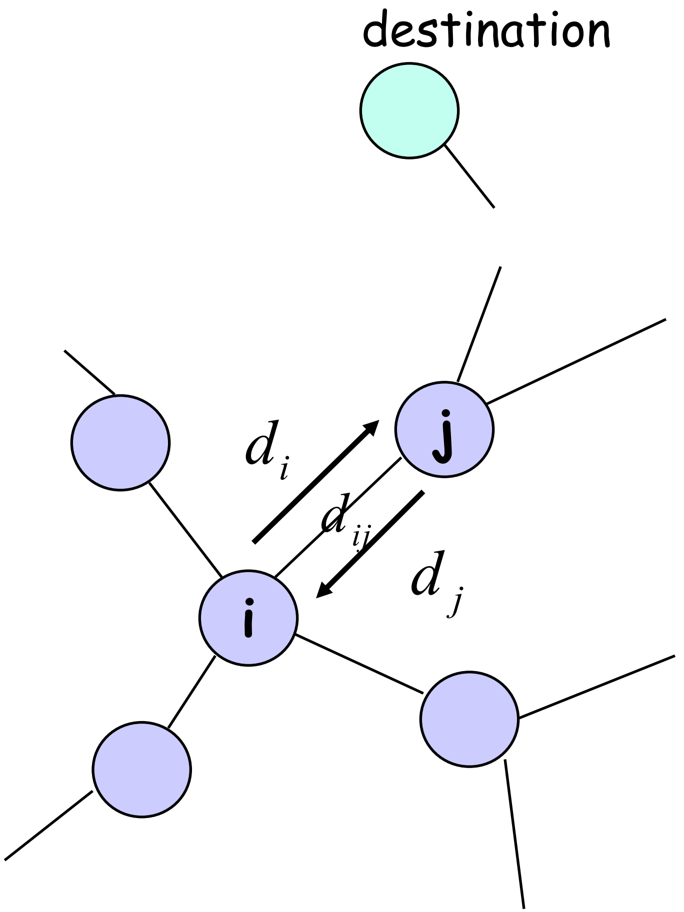


Recap: Routing Computation using Distance Vector/Bellman-Ford Routing

- Distributed computation: At node i, computes

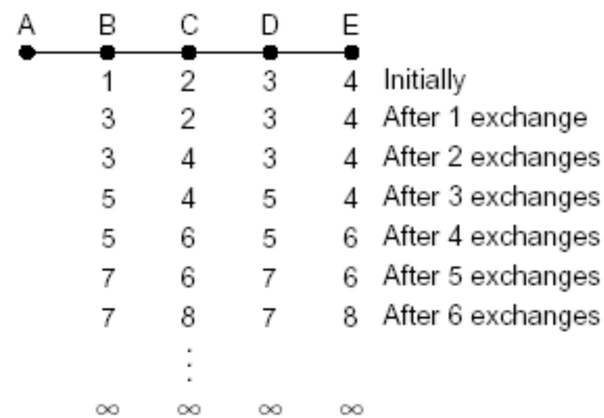
$$\boxed{d_i = \min_{j \in N(i)} (d_{ij} + d_j)}$$

- One way to understand BFA is to consider it as a dynamic programming alg, propagating from dest to other nodes



Recap: Fixing DV/BFA

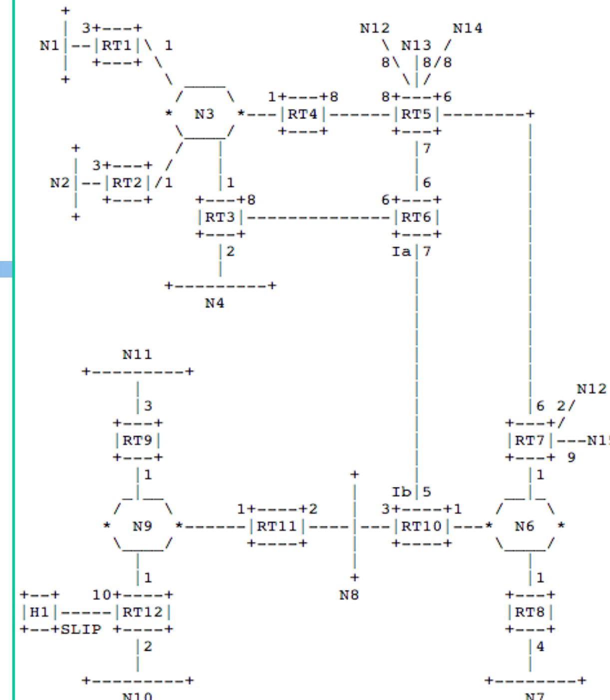
- Property of BFA
  - Bad news may propagate slowly due to loops



- Techniques
  - Reverse poison
    - Avoid two-node loops
  - DSDV
    - Using destination seq to partition into epochs
    - A good example of analysis using global invariants
  - Diffusive Update Alg (DUAL)
    - Utilize backup routes

Recap: Link State Routing

- Basic idea: instead of distributed computing of routes, only distributed state distribution (synchronization)
- Link state distribution can still have much complexity, e.g.,
  - out of order delivery
  - partition and reconnect
  - scalability

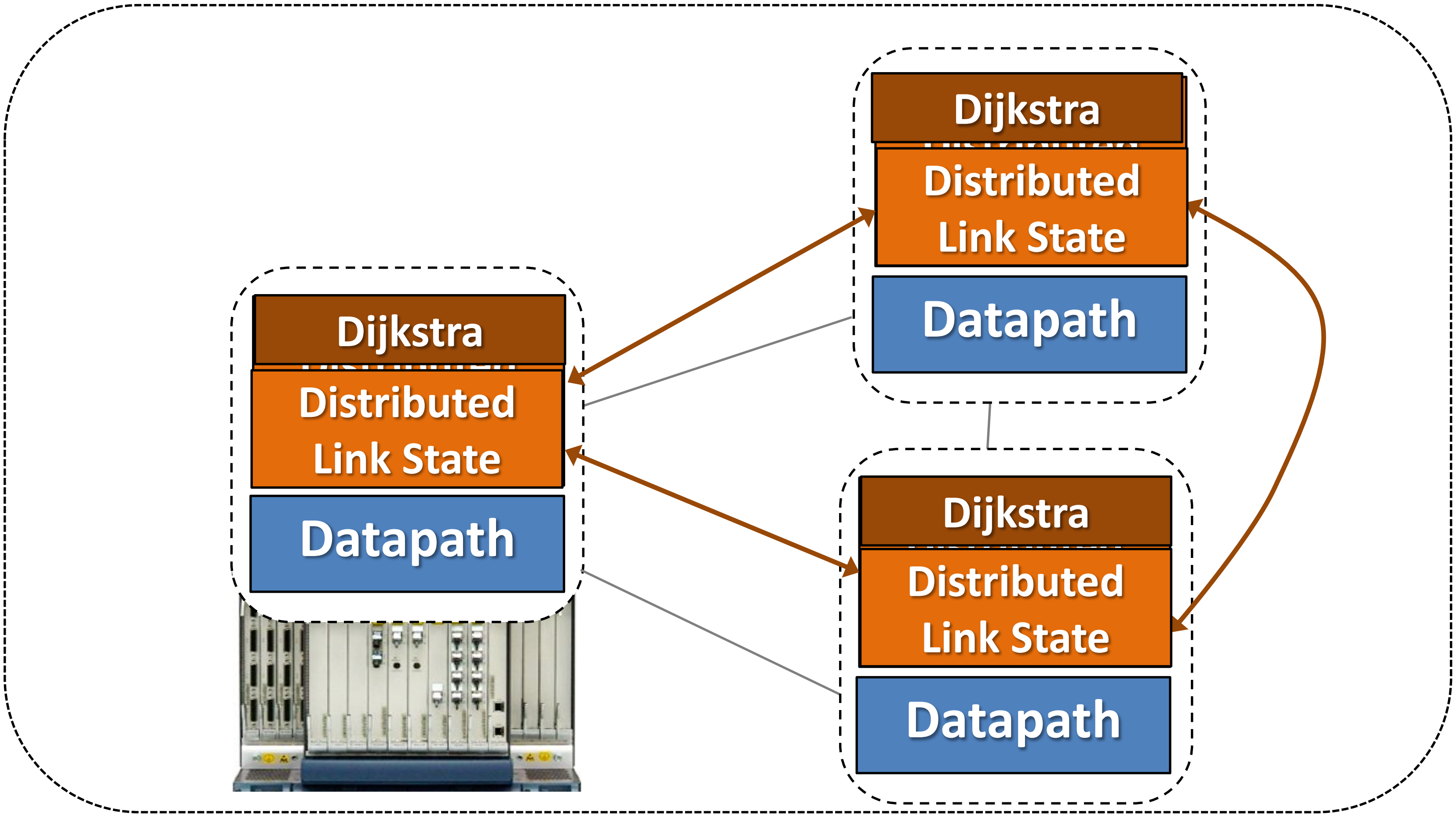


FROM

		RT 1	RT 2	RT 3	RT 4	RT 5	RT 6	RT 7	RT 8	RT 9	RT 10	RT 11	RT 12	N3	N6	N8	N9
	RT1													0			
	RT2													0			
	RT3						6							0			
	RT4					8								0			
	RT5				8		6	6									
	RT6			8		7					5						
	RT7					6									0		
*	RT8														0		
*	RT9																0
T	RT10						7								0	0	
O	RT11															0	0
*	RT12																0
*	N1	3															
	N2		3														
	N3	1	1	1	1												
	N4			2													
	N6							1	1		1						
	N7								4								
	N8										3	2					
	N9									1		1	1				
	N10												2				
	N11									3							
	N12					8		2									
	N13					8											
	N14					8											
	N15							9									
	H1												10				

Figure 3: The resulting directed graph

Roadmap: Routing Computation Architecture Spectrum

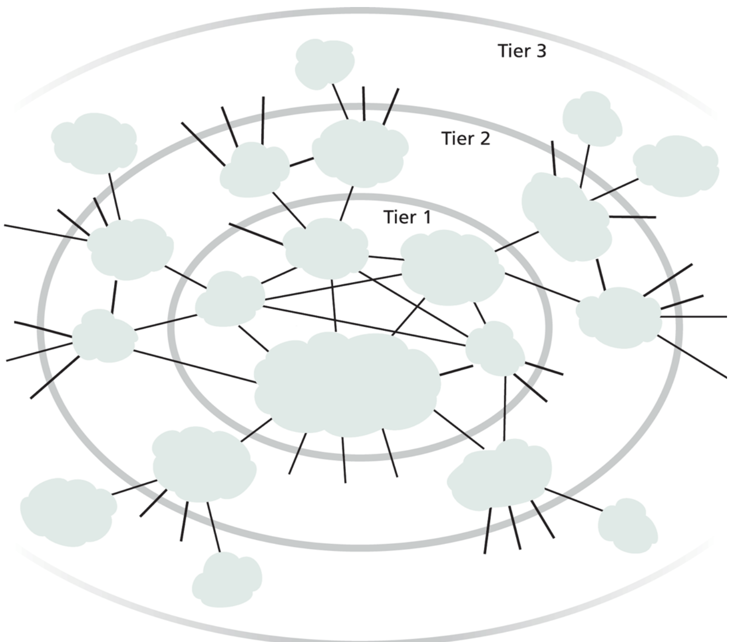


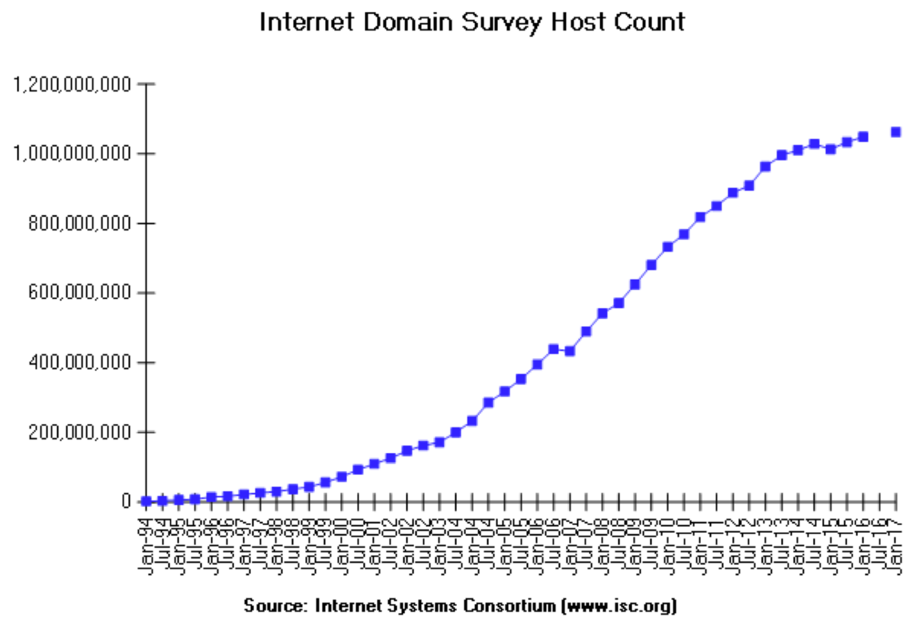
Outline

- Admin and recap
- Network control plane
  - Routing
    - Link weights assignment
    - Routing computation
      - Basic routing computation protocols
        - Distance vector protocols (distributed computing)
        - Link state protocols (distributed state synchronization)
      - Global Internet routing

Exercise

- Does it work to use DV or LS as we discussed for global Internet routing?





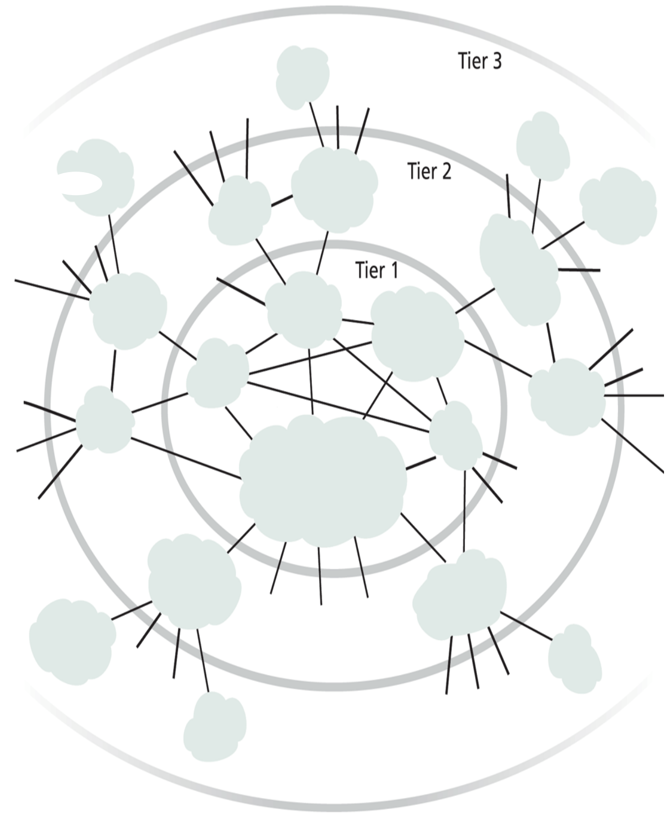
Requirements and Solution of Current Global Internet Routing

- Scalability: handle network size (#devices) much higher than typical DV or LS can handle
  - Solution: Introduce new abstraction to reduce network (graph) size

- Autonomy: allow each network to have individual preference of routing (full control of its internal routing; control/preference of routing spanning multiple networks)
  - Solution: hierarchical routing and policy routing

New Abstraction: Autonomous Systems (AS)

- Abstract each network as an autonomous system (AS), identified by an AS number (ASN)
- Conceptually the global routing graph consists of only autonomous systems as nodes

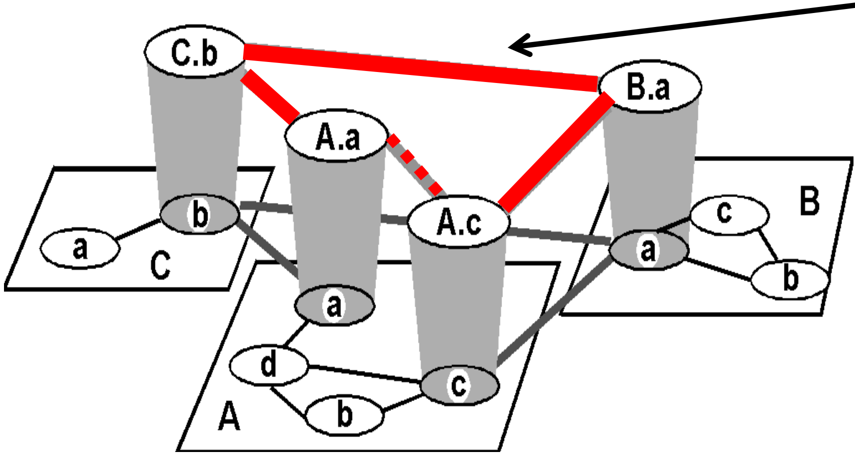


http://www.bgplookingglass.com/list-of-autonomous-system-numbers

Routing with Autonomous Systems

- Internet routing is divided into intra-AS routing and inter-AS routing
  - Intra-AS routing (also called intradomain routing)
    - A protocol running insides an AS is called an Interior Gateway Protocol (IGP), each AS can choose its own protocol, such as RIP, E/IGRP, OSPF, IS-IS
  - Inter-AS routing (also called interdomain routing)
    - A protocol runs among autonomous systems is also called an Exterior Gateway Protocol (EGP)
    - The de facto EGP protocol is BGP

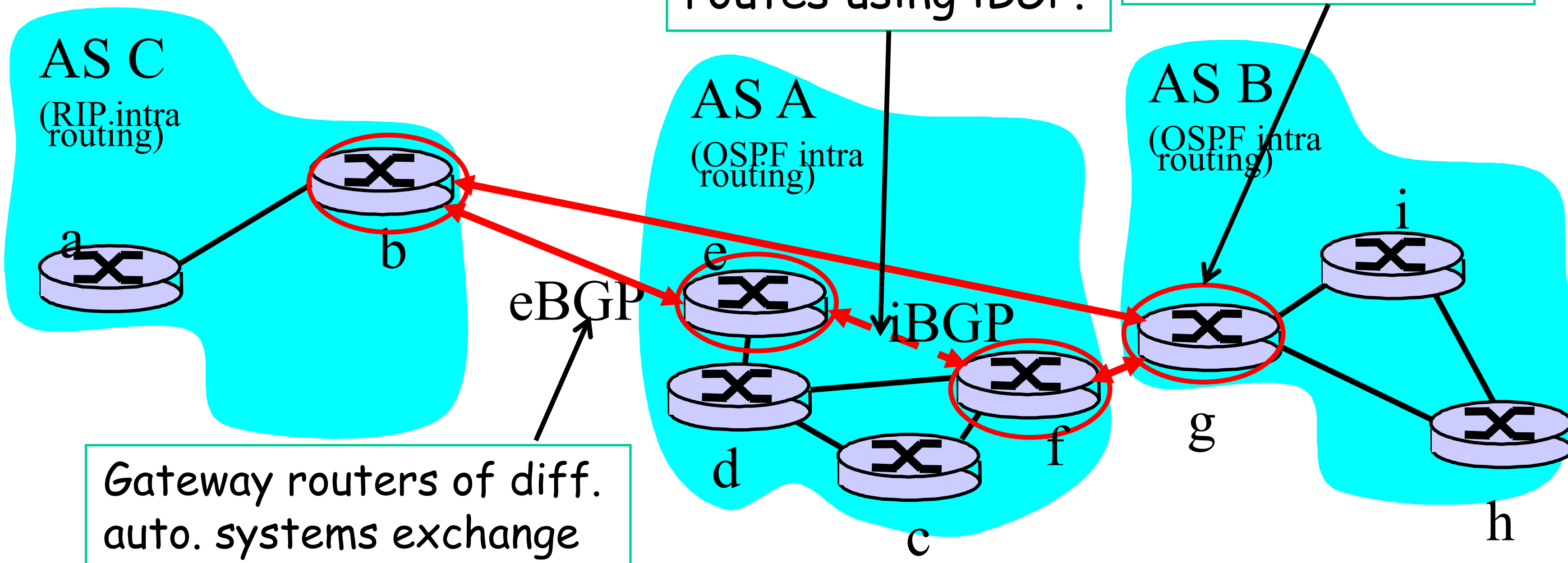
Routing with Autonomous Systems



Inter-AS routers form an overlay

Gateway routers of same AS share learned external routes using iBGP.

Gateway routers participate in intradomain to learn internal routes.



Gateway routers of diff. auto. systems exchange routes using eBGP

Summary: Internet Routing Architecture

- Autonomous systems have flexibility to choose their own intradomain routing protocols
  - allows autonomy

- Only a small # of routers (gateways) from each AS in the interdomain level
  - improves scalability

- Interdomain routing using AS topology instead of detailed topology
  - improves scalability/privacy

Outline

- Admin and recap
- Network control plane
  - Routing
    - Link weights assignment
    - Routing computation
      - Basic routing computation protocols
      - Global Internet routing
        - Basic architecture
        - *BGP (Border Gateway Protocol): The de facto Inter-domain routing standard*

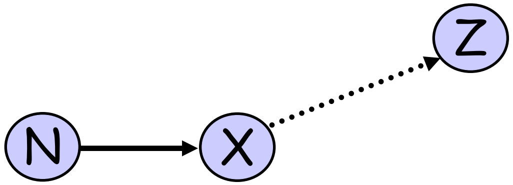
BGP Basic Operations

- BGP is **a Path Vector** protocol
  - similar to Distance Vector protocol
  - a border gateway sends to a neighbor *entire path* (i.e., a sequence of ASNs) to a destination, e.g.,
    - gateway X sends to neighbor N its path to dest. Z:

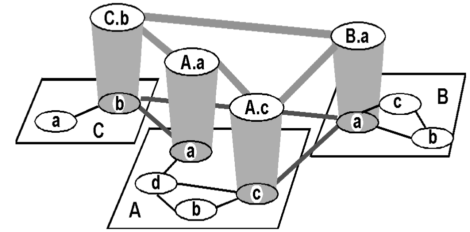
$$path\ (X,Z) = X,Y1,Y2,Y3,\ldots,Z$$

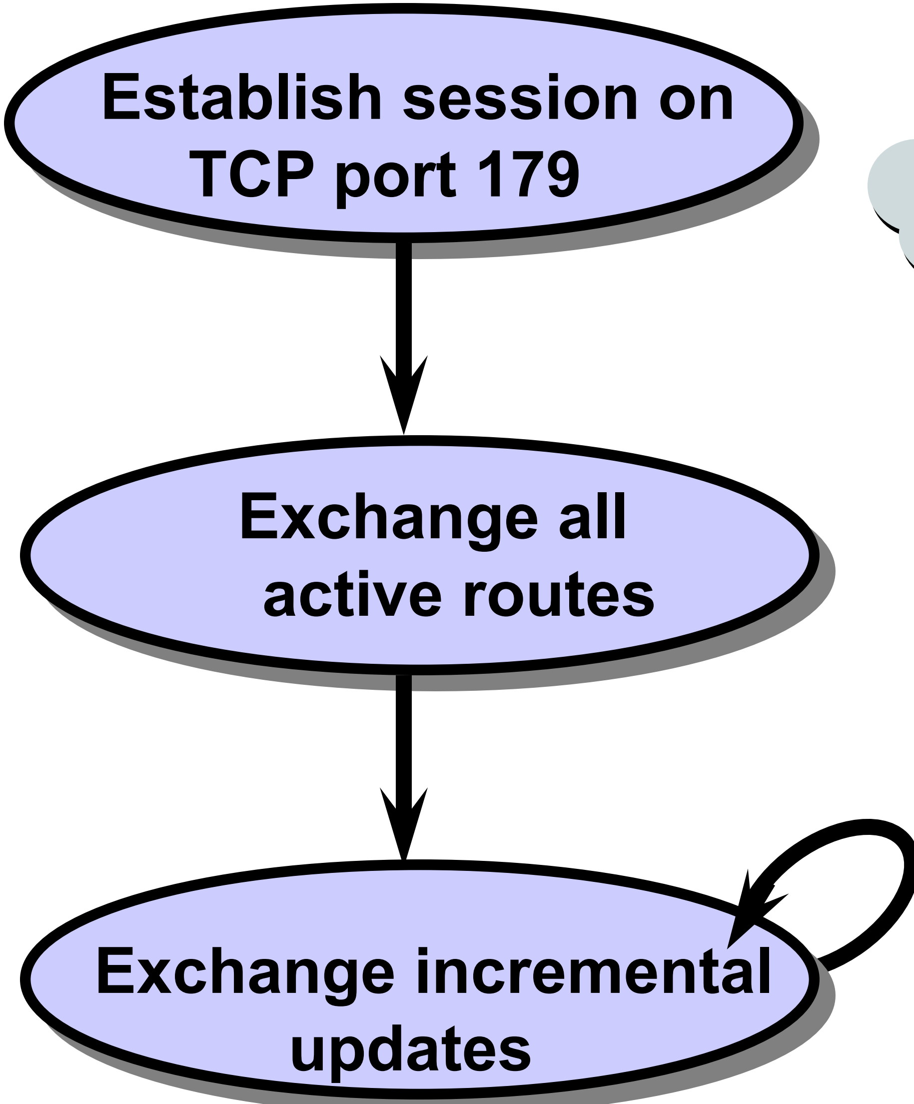
  - if N selects path(X, Z) advertised by X, then:

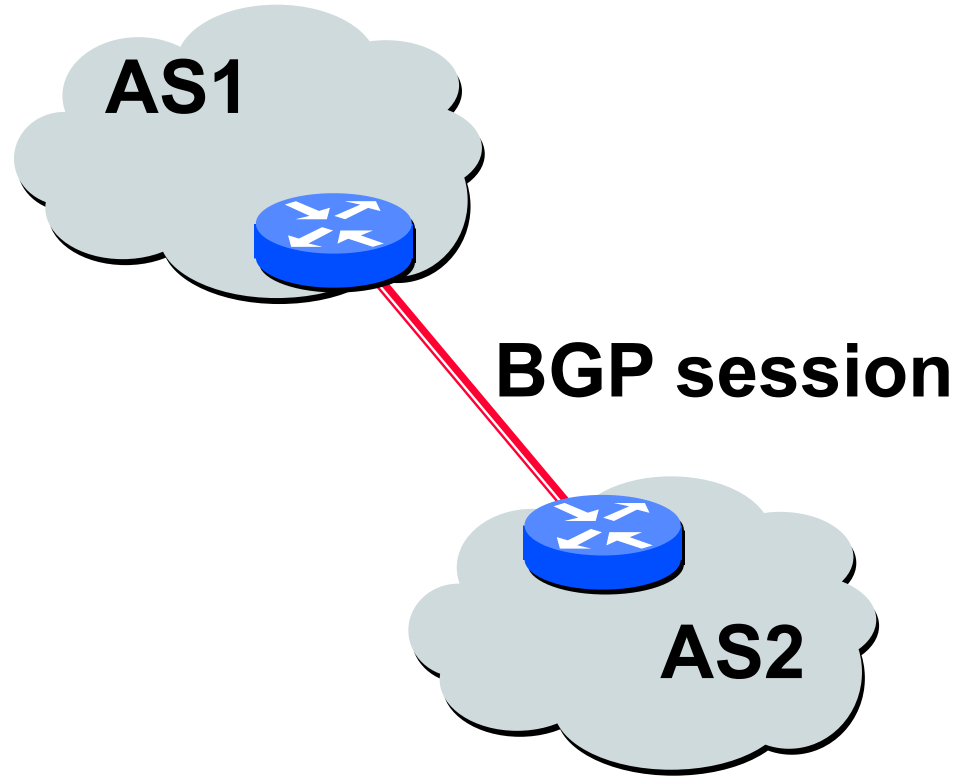
$$path\ (N,Z) = N,\ path\ (X,Z)$$



BGP Basic Operations







while (connection is ALIVE)
  exchange UPDATE message
  select best available route
  if route changes, export to neigh.

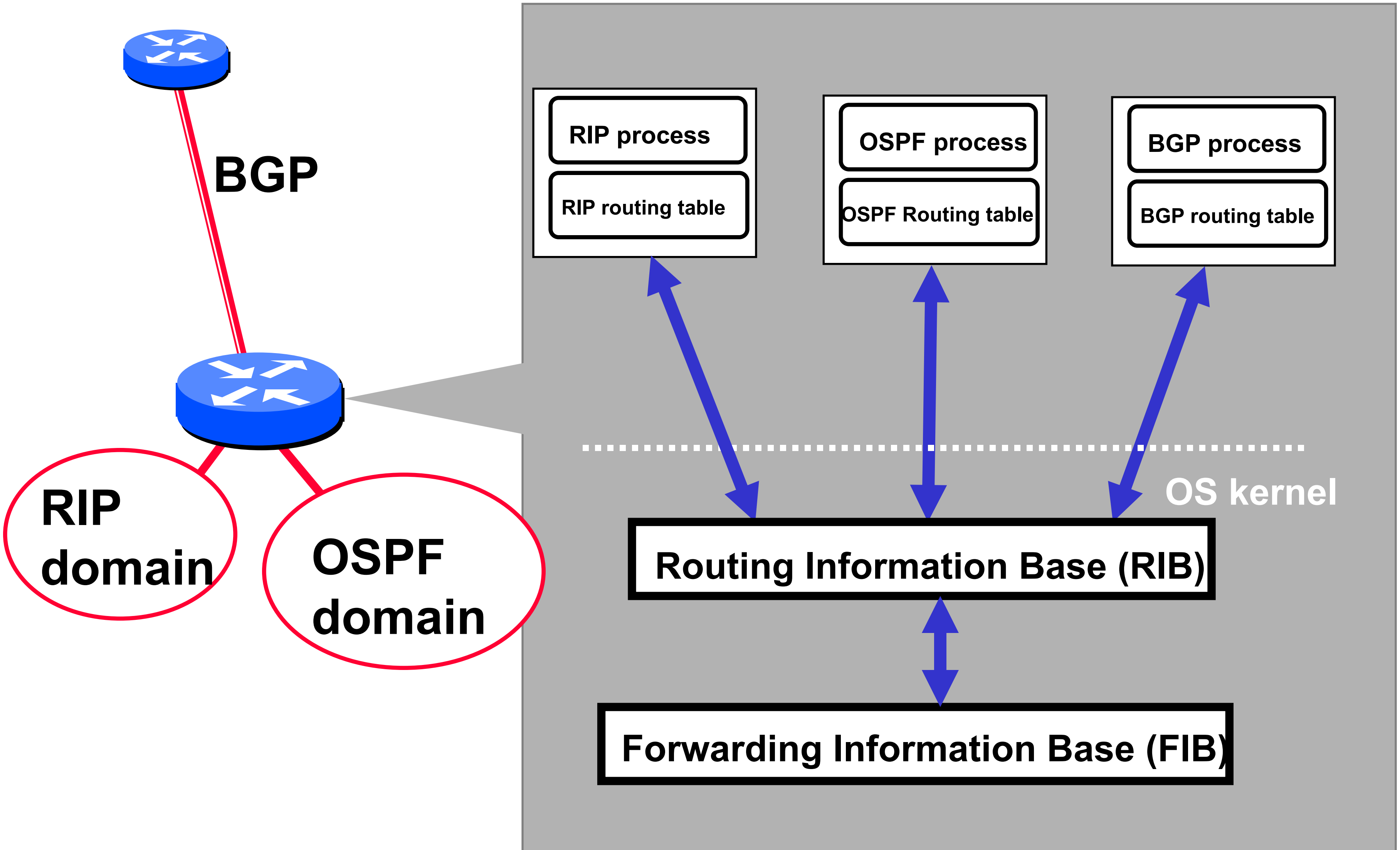
BGP Messages

- Four types of messages
  - OPEN: opens TCP connection to peer and authenticates sender
  - UPDATE: advertises new path (or withdraws old)
  - KEEPALIVE keeps connection alive in absence of UPDATES; also ACKs OPEN request
  - NOTIFICATION: reports errors in previous msg; also used to close connection

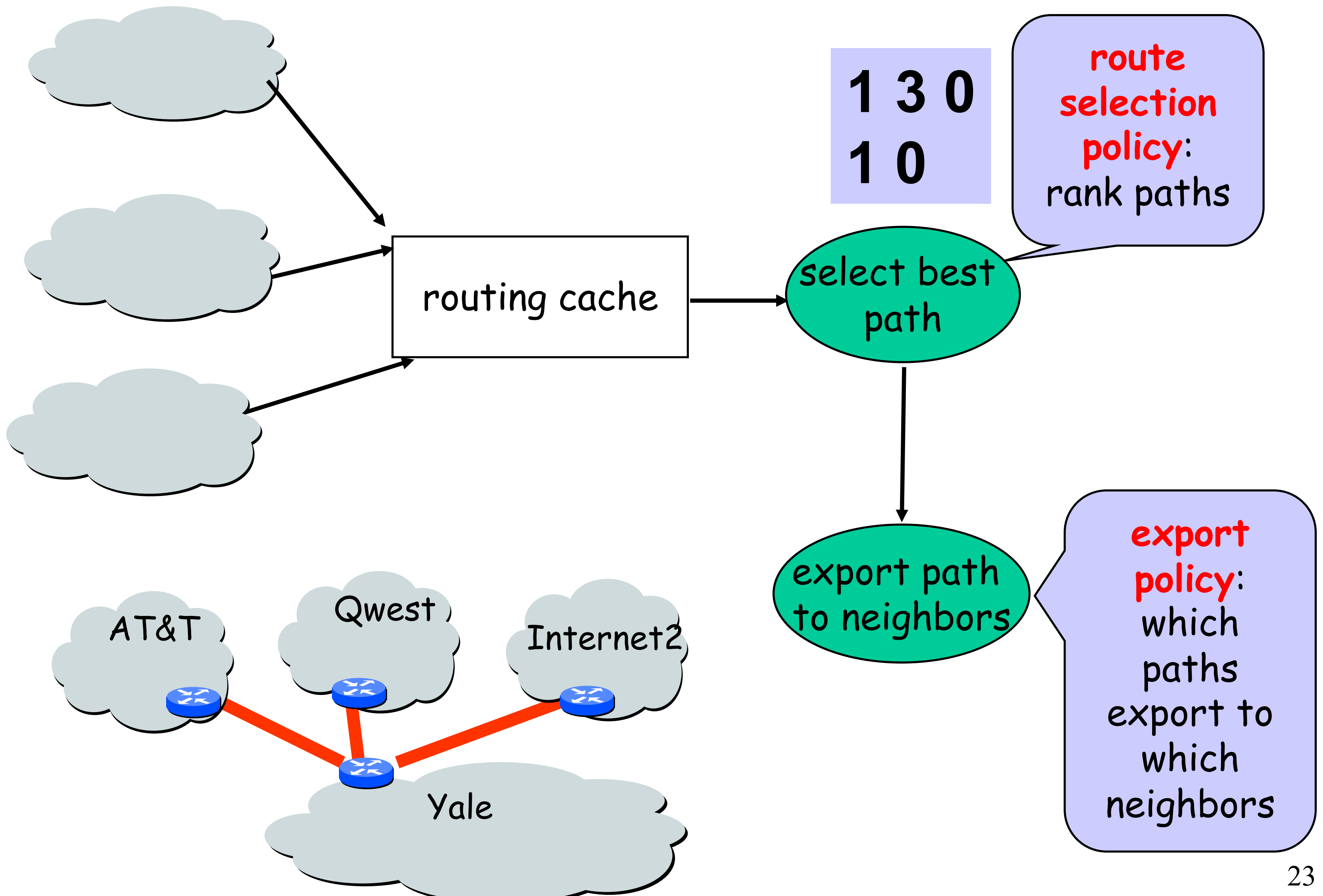
Outline

- Admin and recap
- Network control plane
  - Routing
    - Link weights assignment
    - Routing computation
      - Basic routing computation protocols
      - Global Internet routing
        - Basic architecture
        - *BGP (Border Gateway Protocol): The de facto Inter-domain routing standard*
          - Basic operations
          - *BGP as a policy routing framework (control interdomain routes)*

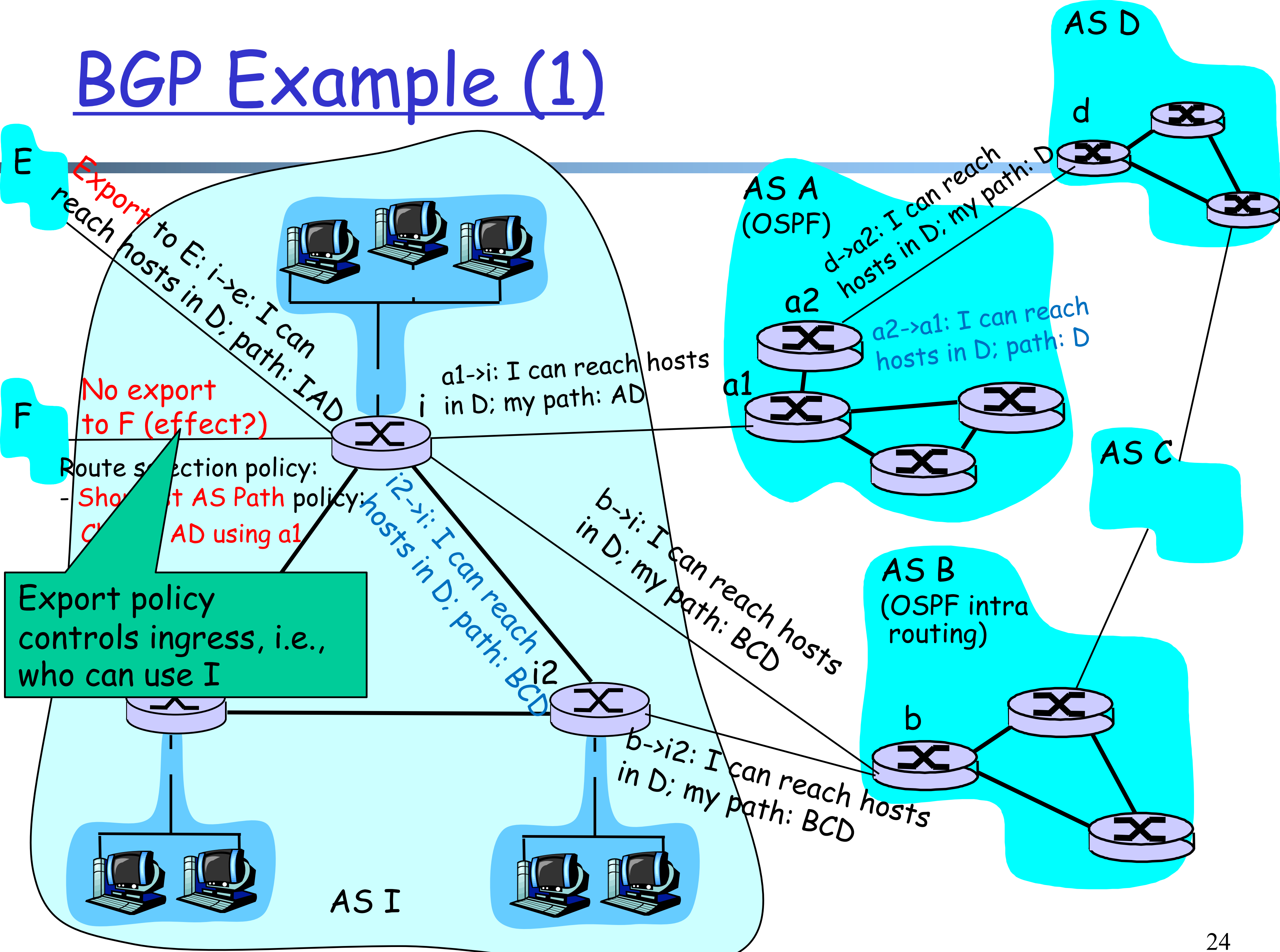
BGP Policy Routing Framework: Label Route Information Sources



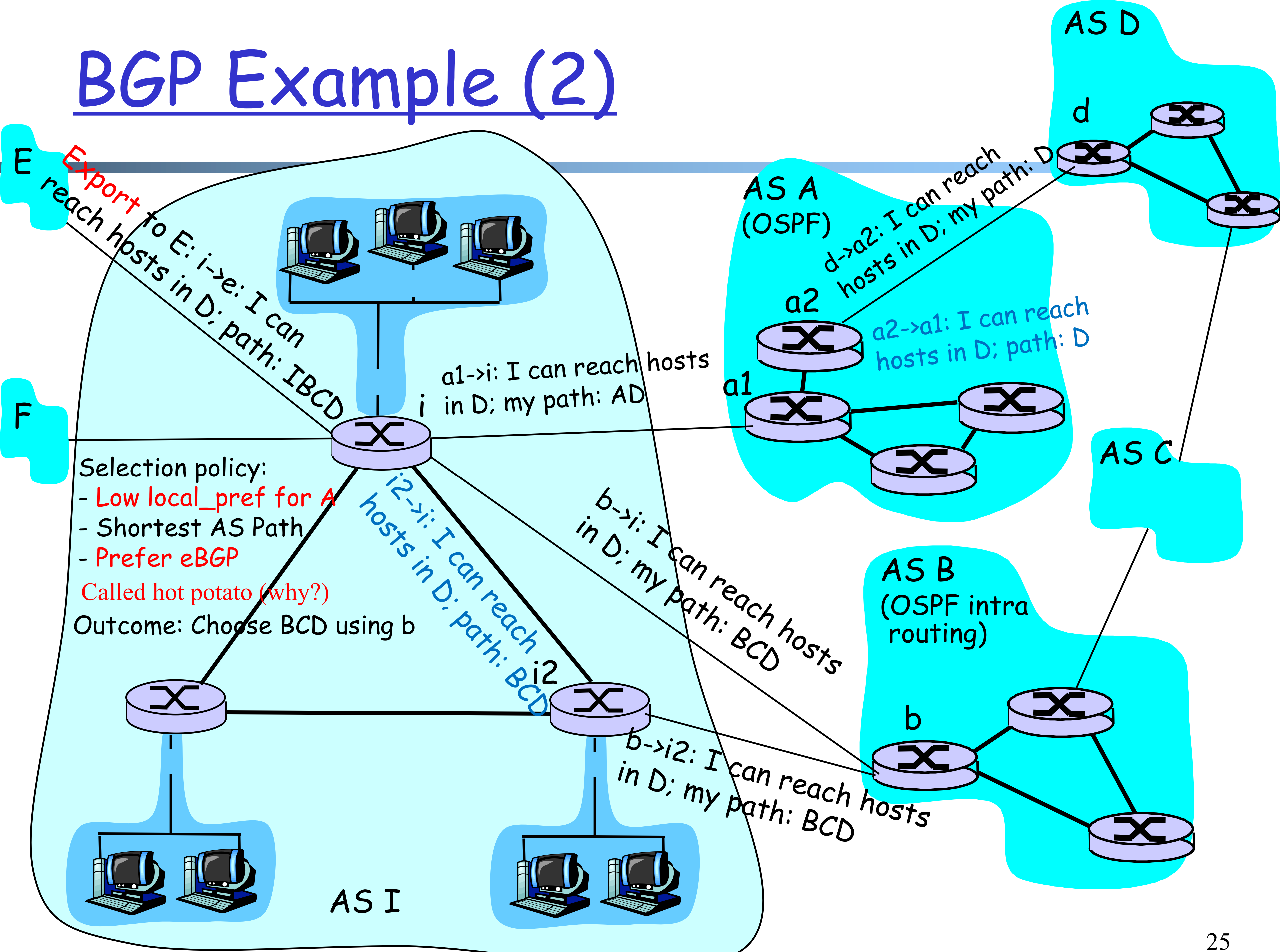
BGP Policy Routing Framework: Decision Components



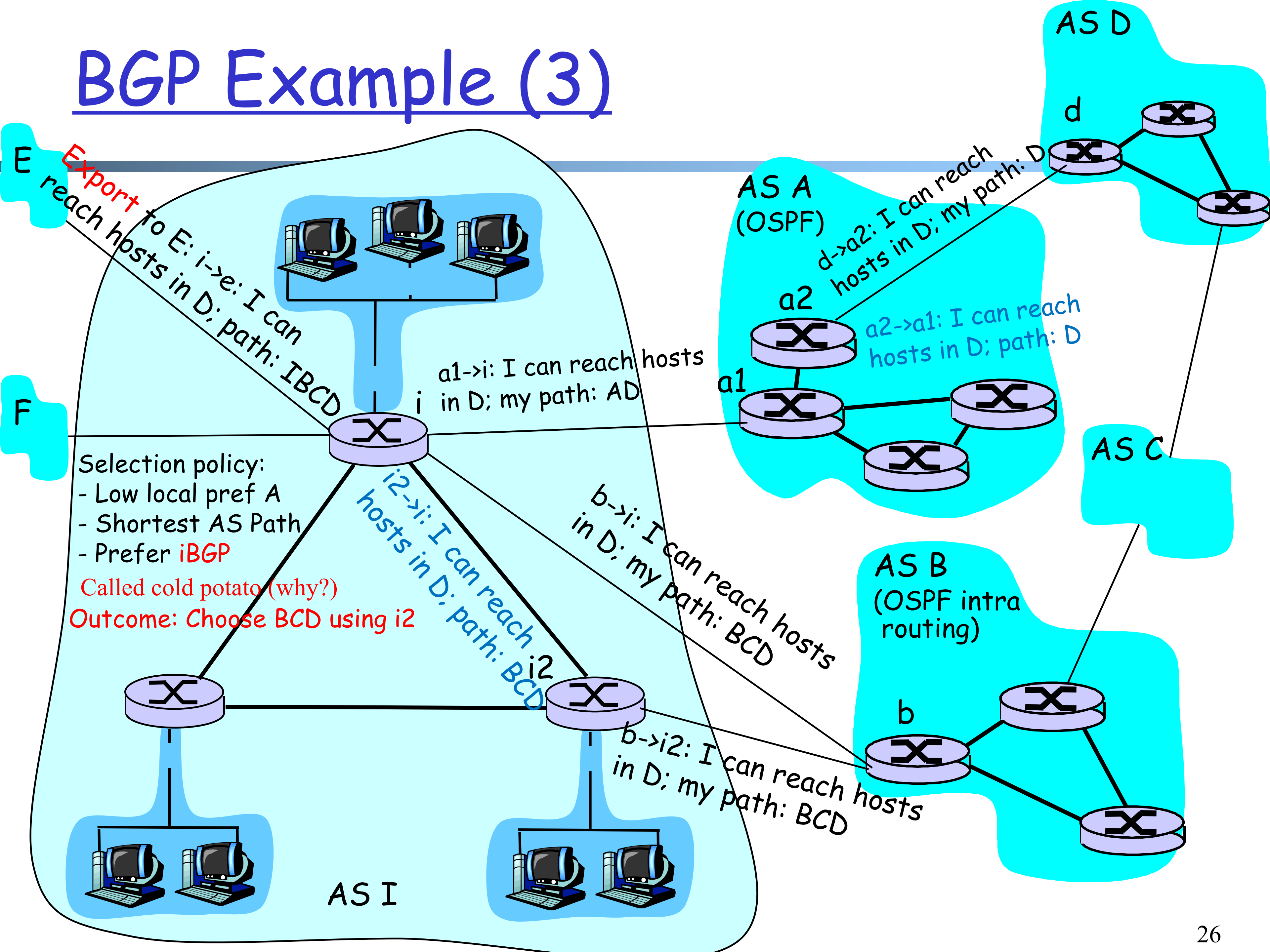
BGP Example (1)



BGP Example (2)



BGP Example (3)



Observing BGP Paths

- Using one of the looking glass servers: http://www.bgp4.as/looking-glasses

  https://www.gin.ntt.net/looking-glass/

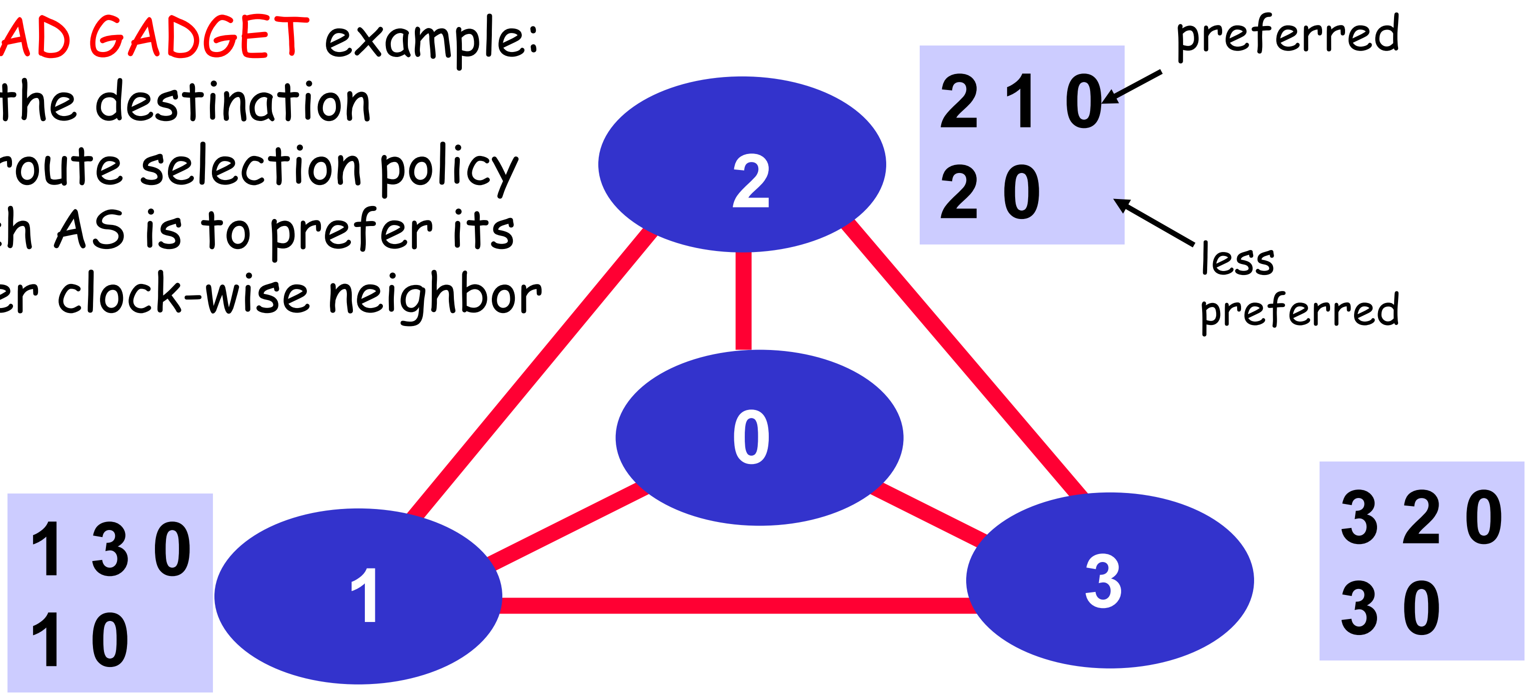
Outline

- Admin and recap
- Network control plane
  - Routing
    - Link weights assignment
    - Routing computation
      - Basic routing computation protocols
      - Global Internet routing
        - Basic architecture
        - *BGP (Border Gateway Protocol): The de facto Inter-domain routing standard*
          - Basic operations
          - BGP as a policy routing framework (control interdomain routes)
          - *Policy/interdomain routing analysis*

Motivation: Policy Routing Stability

- A policy routing system can be considered as a system to aggregate local preferences, but aggregation may not be always successful.

The BAD GADGET example:
- 0 is the destination
- the route selection policy of each AS is to prefer its counter clock-wise neighbor

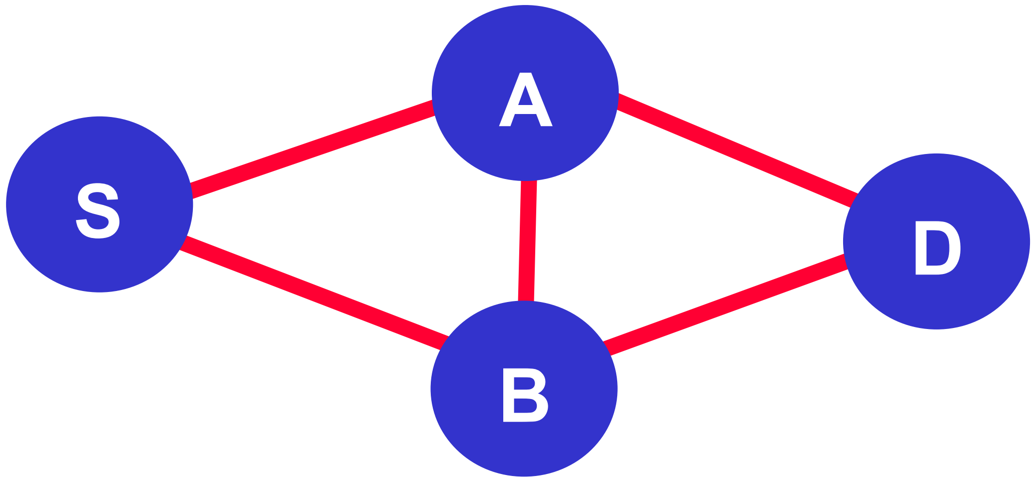


Policy (preferences) aggregation fails: routing instability !

General Framework of Preference Aggregation

- Also called Social Choice
  - Given individual preferences, define a framework to aggregate individual preferences:
    - A set of choices: a, b, c, …
    - A set of voters 1, 2, …
      - Each voter has a preference (ranking) of all choices, e.g.,
        - voter 1: a > b > c
        - voter 2: a > c > b
        - voter 3: a > c > b
    - A well-specified aggregation rule (protocol) computes an aggregation of ranking, e.g.,
      - Society (network): a > b > c

Example: Aggregation of Global Preference



- Choices (for S->D route): SAD, SBD, SABD, SBAD
- Voters S, A, B, D
- Each voter has a preference, e.g.,
  - S: SAD > SBD > SABD > SBAD
  - ...

Arrow's Aggregation Framework

- Axioms:
  - Transitivity
    - if a > b & b > c, then a > c
  - Unanimity:
    - If all participants prefer a over b (a > b) => a > b
  - Independence of irrelevant alternatives (IIA)
    - Social ranking of a and b depends only on the relative ranking of a and b among all participants
- Result:
  - Arrow's Theorem: Any constitution that respects transitivity, unanimity and IIA is a dictatorship.

Proofs of Arrow's Theorem

- There are quite a few proofs, and the six-page paper linked on the Schedule page gives three simple proofs.
- Below, I give the key insight of the proof using approach 1.

The Extremal Lemma

- Let choice b be chosen arbitrarily. Assume that every voter puts b at the very top or the very bottom of his ranking. Then society must as well (even if half voters put b at the top and half at the bottom)
- Proof: by contradiction.
  - Assume there exist a and c such that society has a >= b; b >= c.
    - By transitivity, a >= c
  - We can move c above a w/o changing ab or cb votes, leading to c > a
    - By unanimity, c > a