Lecture 2: Architecture and Principles

Sylvia Ratnasamy

(Spring, 2021)

CS268 via Zoom

- **Not ideal but let's try our best to make it work – will need your help for this!**

- **Here's the current plan but we can iterate as we go:**
  - Please turn on your video!
  - I will not "mute all"; do mute yourself if you're in a noisy environment
  - If you have a question/comment:
    - Raise your (zoom) hand or post your question in the chat window
    - If the presenter doesn't spot it, feel free to just interrupt
    - Presenters: periodically stop and poll for questions
  - We'll use breakout rooms for discussions (more on this later)

Today’s Reading

Clark’88:

The Design Philosophy of the DARPA Internet Protocols

Context: David D. Clark (MIT)

- Chief Protocol Architect for the Internet from 1981-89
- Continues to be a network visionary today
- At the time of writing ...
  - Almost no commercial Internet
  - 1 year after Cisco's 1st product, IETF started
  - Number of hosts reaches 10,000
  - NSFNET backbone 1 year old; 1.5Mb/s

You said:

- *“Yes. Instead of taking an architecture for granted, if we understand *why* something is the way it is, we can dare to change it” -- Eric Liu*

- *“...incredibly important. By clarifying the reasoning, we can determine the first-principles ... and choose to continue to design to them or question them all together” – Sarah*

- *allows us to understand that the values of a system become the system. – Arjun*

- *“Potentially controversial answer: I wouldn't say this is an important paper.” – Aditya R.*

- *“I would rate it at medium importance.” – Michael*

- *I don't think it's particularly important – Zizheng*

You said:

- *Surprised at the reason performance was so challenging to tackle – Nidhi*
- *“Considering the internet was inspired/motivated by military use I am surprised there is not much discussion on ... membership management” – Shishir*
- *I found it surprising that some of the topics mentioned in this paper are still relevant in research today. – Ziyao*
- *It's surprising that the internet exists at all! – Aditya S.*

Tips for success?

- Be clear about your goals
- Grounded
  - *"… to give users on the packet radio network access to the large service machines on the ARPANET."*
- Learn by building, iterating ("engineering attitude")
  - e.g., separation of TCP/IP
- Attention to detail (to an almost mystifying level)
  - e.g., discussion of `End of Letter' flag

DDC'88

Goal 0: An "effective" technique for multiplexed utilization of existing interconnected networks.

Goal 1: Communication must continue despite loss of networks or gateways

Goal 2: Must support multiple types of communication service

Goal 3: Must accommodate a variety of networks [underneath]

Goal 4: Must permit distributed management of its resources

Goal 5: Must be cost effective

Goal 6: Must permit host attachment with a low level of effort

Goal 7: The resources used in the Internet architecture must be accountable

DDC'88

Goal 0: An "effective" technique for multiplexed utilization of existing interconnected networks.

Goal 1: Communication must continue despite loss of networks or gateways

Goal 2: Must support multiple types of communication service

Goal 3: Must accommodate a variety of networks [underneath]

Goal 4: Must permit distributed management of its resources

Goal 5: Must be cost effective

Goal 6: Must permit host attachment with a low level of effort

Goal 7: The resources used in the Internet architecture must be accountable

Prioritization for a commercial network?

DDC'88

Goal 0: An "effective" technique for multiplexed utilization of existing interconnected networks.

Goal 1: Communication must continue despite loss of networks or gateways

Goal 2: Must support multiple types of communication service

Goal 3: Must accommodate a variety of networks [underneath]

Goal 4: Must permit distributed management of its resources

Goal 5: Must be cost effective

Goal 6: Must permit host attachment with a low level of effort

Goal 7: The resources used in the Internet architecture must be accountable

What's missing?

Goal 0: An effective technique for multiplexed utilization of existing interconnected networks

- Motivated by a very tangible application:

"… to give users on the packet radio network access to the large service machines on the ARPANET."

- Guided by a pragmatic sensibility:

"… was felt necessary to incorporate the then existing network architectures if the Internet was to be useful in a practical manner"

- (Understated) novelty:

"… ambition to come to grips with the problem of integrating separately administered entities into a common utility"

Goal 0: An effective technique for multiplexed utilization of existing interconnected networks

- Multiplexing (sharing)
  - Shared use of a single communication infrastructure
- Existing networks (interconnection)
  - Tries to define an “easy” set of requirements for the underlying networks to support as many as possible

Goal 0: An effective technique for multiplexed utilization of existing interconnected networks

- How: different networks connected by packet switched, store-and-forward routers/gateways

---

- Was packet switching a good choice?
- What if performance were goal#1?

- *"Alternative of datagram?" – Zhanyuan*
- *"Has circuit switching been implemented at any scale?" – Aditya S.*

.

- *"I wonder what an architecture with performance as a high priority would look like" – Sarah*

Goal 1: Internet communication must continue despite loss of networks or gateways.

“Entities should be able to continue communicating without having to reestablish the high level state of their conversation”

“The architecture [should] mask any transient failure.”

Leads to:

1. “Fate-sharing”

“Fate Sharing”

- *A distributed system is one where a machine I’ve never heard of can cause my program to fail. — Leslie Lamport*

- Basic idea:
  - Communication shouldn’t be disrupted by the failure of a particular router/node in the network; two endpoints should be able to communicate if <u>some</u> path exists between them
  - Leads to the decision to keep communication state (e.g., which pkts have been transmitted) at the endpoints, not in routers
  - Now, if state is lost it’s because the endpoint failed so it doesn’t matter! I.e., an endpoint and its state “share the same fate”

Goal 1: Internet communication must continue despite loss of networks or gateways.

"Entities should be able to continue communicating without having to reestablish the high level state of their conversation"

"The architecture [should] mask any transient failure."

Leads to:

1. "Fate-sharing" → only lose commn. state if endhost is lost
2. Stateless packets switches → datagrams

Q. How successful?

Q. Is there a downside to this approach?

Q. Is this ideal for replicated / content-centric services?

Goal 1: Internet communication must continue despite loss of networks or gateways.

“Entities should be able to continue communicating without havi… …versation”

“Th… …e.”

service that the synchronization between the sender and the receiver may have been lost. It was an assumption in this architecture that synchronization would never be lost unless there was no physical path over which any sort of communication could be achieved. In other words, at the top of transport, there is only one failure, and it is total partition. The architecture was to mask

Leads…

1. “Fat… …is lost
2. Stateless packets switches → datagrams

Q. How successful?

Q. Is there a downside to this approach?

Q. Is this ideal for replicated / content-centric services?

Q. What level of survivability do we have today?

Goal 1: Internet communication must continue despite loss of networks or gateways.

"Entities should be able to continue communicating without havi… …versation"
"Th… …e."

Leads…

1. "Fat… …is lost
2. Statele…

service that the synchronization between the sender and

…st, it is still second to the top level goal of interconnection of existing networks. A more survivable technology might have resulted from a single multi-media network design. For example, the Internet makes very weak assumptions about the ability of a network to report that it has failed. Internet is thus forced to detect network failures using Internet level mechanisms, with the potential for a slower and less specific error detection.

Q. How successful?

Q. Is there a downside to this approach?

Q. Is this ideal for replicated / content-centric services?

Q. What level of survivability do we have today?

Q. Still a good assumption today?

Goal 2: Support multiple types of service

"Different services distinguished by differing requirements for such things as speed, latency, and reliability"

Leads to: separation of TCP from IP

Q: In retrospect, how important was this?

Goal 2: Support multiple types of service

"The hope was that multiple types of service could be constructed out of the basic datagram building block…For example, …"

"It proved more difficult than first hoped to provide multiple types of service without explicit support from the underlying network"

Q: Why is that? What's happened since?

Goal 3: Support varieties of networks

"[Very important that the Internet] be able to incorporate and utilize a wide variety of network technologies"

Leads to: a minimum set of assumptions about the function the network will provide

1. "network can transport a packet"
2. "of reasonable size"
3. "delivered with reasonable reliability"

Q: How successful?

Goal 3: Support varieties of networks

"Number of services explicitly not assumed from the network. These include reliable or sequenced delivery, network level broadcast or multicast, priority ranking of transmitted packets, support for multiple types of service, and internal knowledge of failures, speeds or delays"

Predicts many research efforts to come…

"…undesirable [that the network support these] because services would have to be reengineered for every single network..

Other goals

Goal 4: The Internet architecture must permit distributed management of its resources

Q. Does it accomplish this?

Goal 5: The Internet architecture must be cost effective.

Q. Is it cost effective?

Goal 6: Low cost of attachment

Goal 7: The resources… must be accountable

Q. What does this mean?

Q. What would such a network look like?

Other discussion questions

"The most important change in the Internet…will probably be the development of a new generation of tools for management of resources..."

Q. Has this happened?

Q. What exactly do we mean by management?

Other discussion questions

“The relationship between architecture and performance is an extremely challenging one...”

- *the goal of the architecture was […] to permit variability*
- *“no useful formal tools for describing performance”*

- Q: Have we made any progress on this?
- Q: Is performance more/less important today?

Other discussion questions

"There may be a better building block than the datagram …"

- *identify a sequence of packets -- "flow"*
- *Proposes that routers maintain flow state as optimization ("soft state")*

- Q: Do we see this today?

Author’s conclusion

- “Datagram” good for most important goals, but poor for the rest of the goals.
- Processing packets in isolation, resource management, accountability all hard.
- Anticipates flows and “soft-state” for the future.

Drawbacks?

- Recognition / discussion of control vs. data plane
  - Clearer stance on network state
- Elevate/exploit the role of IP border gateways
- Contemplate success

Design time.

- Let’s split up into breakout rooms
- Discuss: how would you extend/redo the Internet architecture to provide accountability?
  - Remember: define what you mean by accountability
  - Be clear: accounting of what? To whom? Etc.
  - Tip: start with *some* solution that meets your goal then worry about optimizing for scalability, etc.
  - Remember: you’re tackling an unsolved problem!

Quick introduction to next week's readings

- Three topics:
  - Active networks
  - Quality of Service (QoS)
  - Multicast
- Each a "hot topic" in its time
- Not your usual paper reading fare
  - Can feel a little dated: ASCII art, no graphs, etc
  - Some assume too much detail, others offer too little

A few tips:

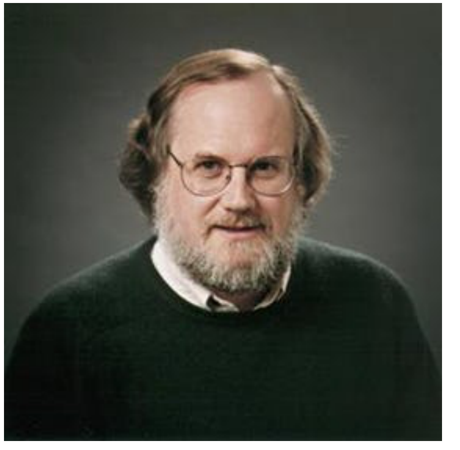
- The Active Networking paper
  - Focus on the vision, high-level idea of how, and why
  - Skim details of how (safe execution, intermediate code, etc)
  - But do consider whether modern programming environments make their goal easier/harder
- The QoS paper
  - Involves a lot of building blocks many of which were presented in detail elsewhere
  - Try and get a sense of the overall solution
  - E.g., just skim S4.3 and S5

A few tips: Multicast

- The papers you'll read assume knowledge of the original proposal for IP multicast: [Deering'88]
  - Provided as optional reading
- I'll do a quick overview. The key is to understand
  - What was the IP multicast service model
  - Scaling issues with Deering's multicast routing proposal

Steve Deering

- *Early developer on X.25 (precursor to IP) and first email system*
- *Stanford PhD (proposed IP Multicast)*
- *Cisco Fellow*
- *Lead designer of IPv6*



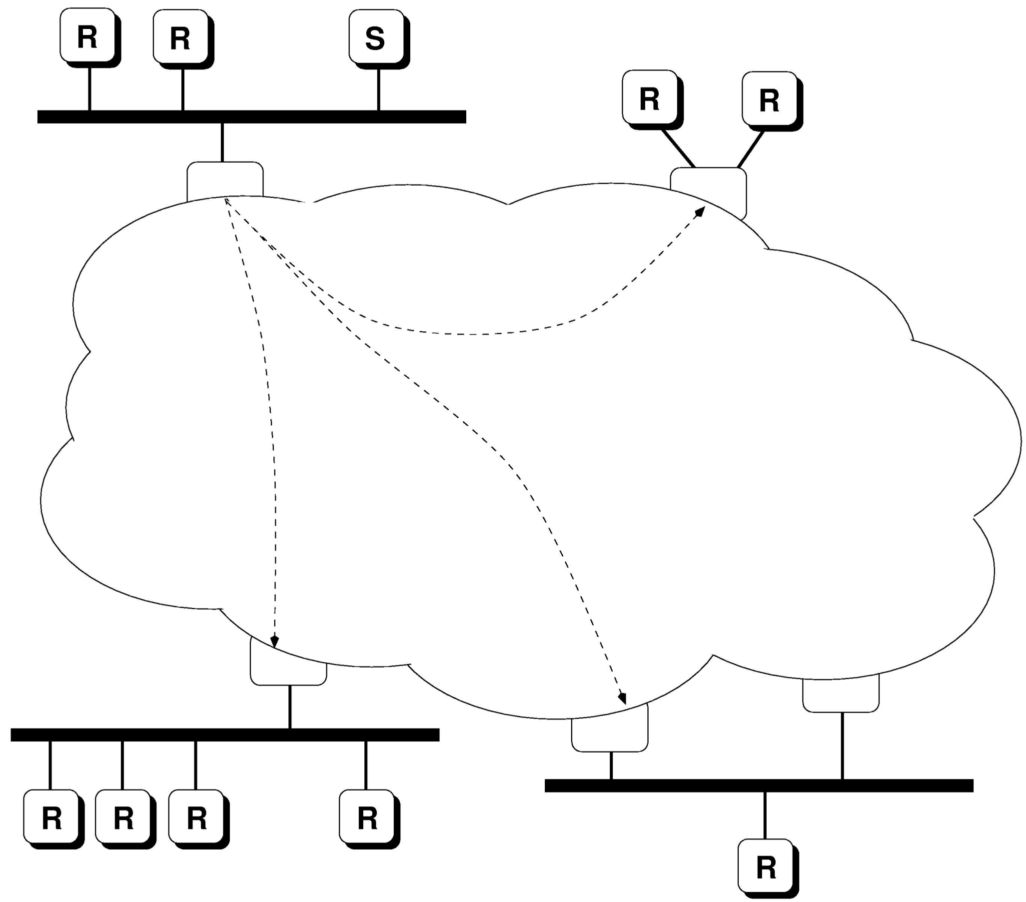
The IP Multicast Service Model

- Problem: how do you efficiently deliver a packet to multiple receivers

- Proposal: a new group delivery service model
  - *group_address* : an IP address designated for multicast
  - JOIN/LEAVE(*group_address*), SENDTO(*group_address*)
  - Open: anyone can join, anyone can send
  - Indirection: logical group address hides details of membership from hosts
  - Best-effort delivery semantics of IP unicast
  - Dynamic group membership: hosts join/leave at will

Deering's Network Model

- Interconnected LANs
- LANs support link-level multicast
- Approach: extend existing LAN and unicast routing solutions

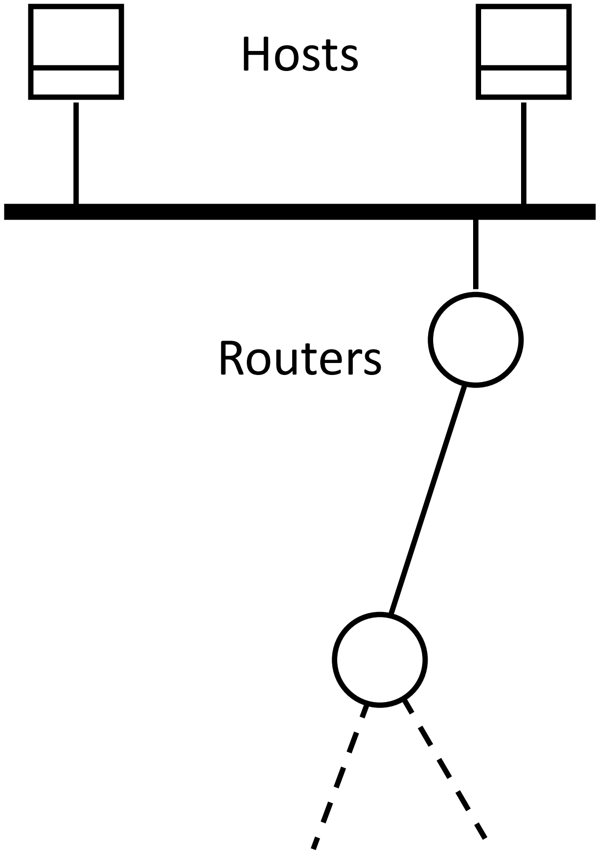


Architecture

Service model

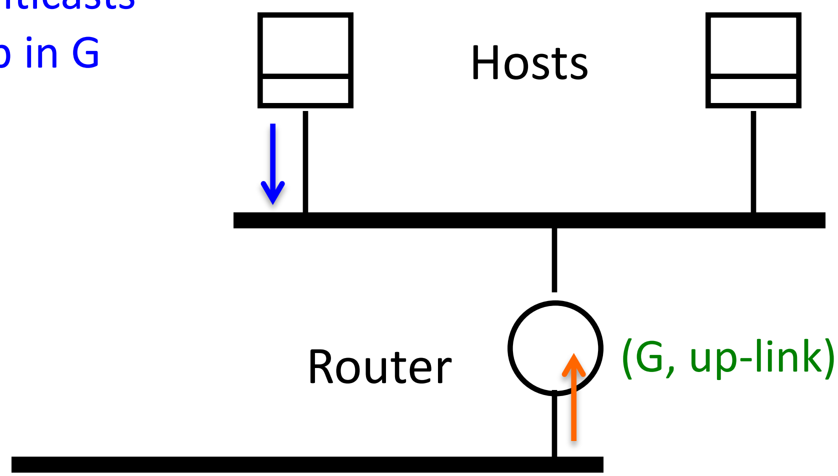
Host-to-router protocol
(IGMP)

Multicast routing protocols
(extends LS, **DV**)



Host-to-Router

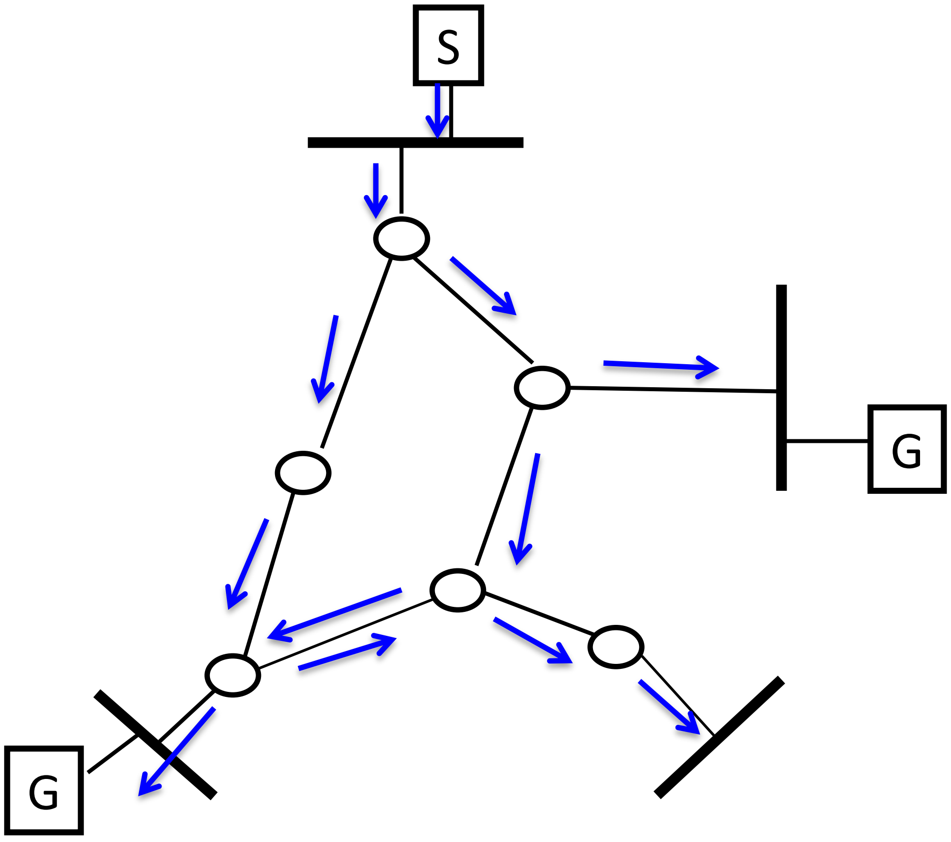
(1) Host periodically broadcasts/multicasts its membership in G



(2) Router installs forwarding state for G

(3) When a packet destined for G arrives, the router multicasts it on the LAN if it has an entry for G

Between Routers

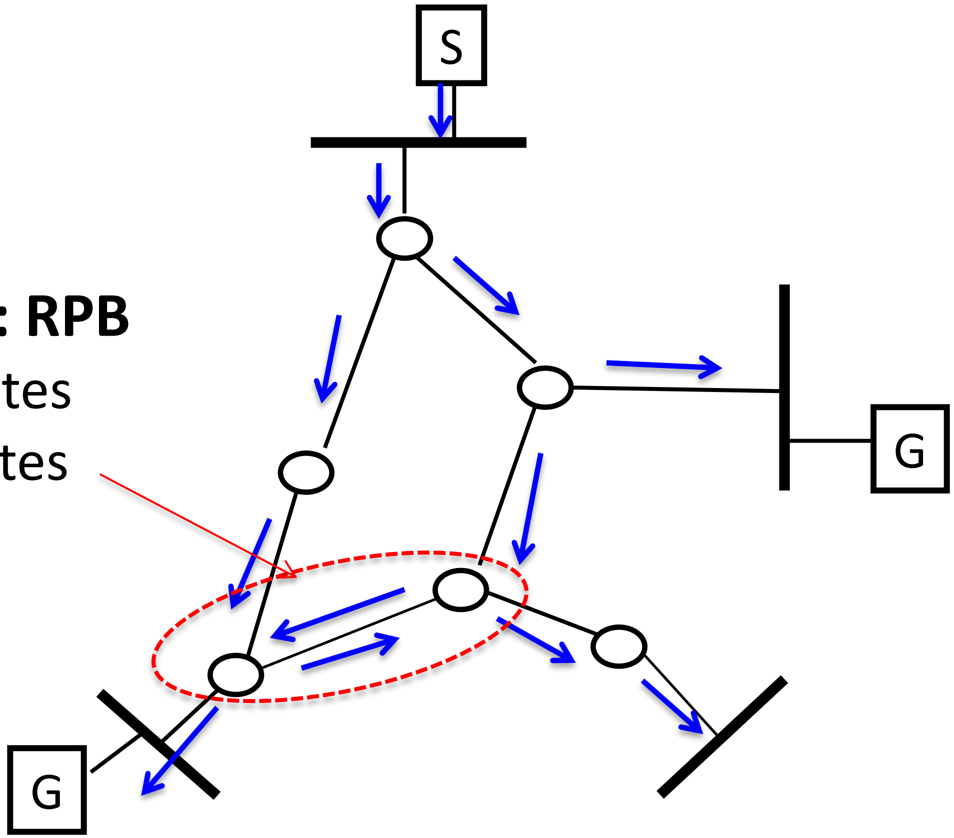


Step 1:
Reverse Path Flooding
If a packet from S arrives via the shortest path **to** S, then forward on all but the incoming link

Between Routers

Step 1:
Reverse Path Flooding
If a packet from S arrives via the shortest path **to** S, then forward on all but the incoming link

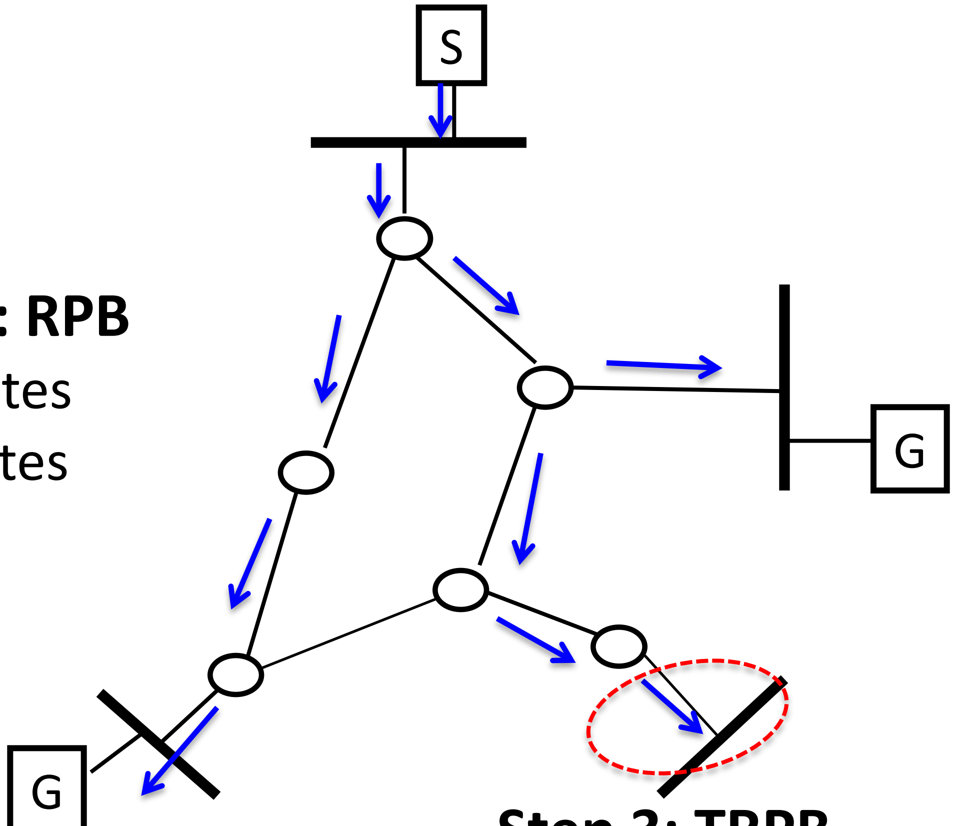
Step 2: RPB
Eliminates duplicates



Between Routers

Step 1:
Reverse Path Flooding
If a packet from S arrives via the shortest path **to** S, then forward on all but the incoming link

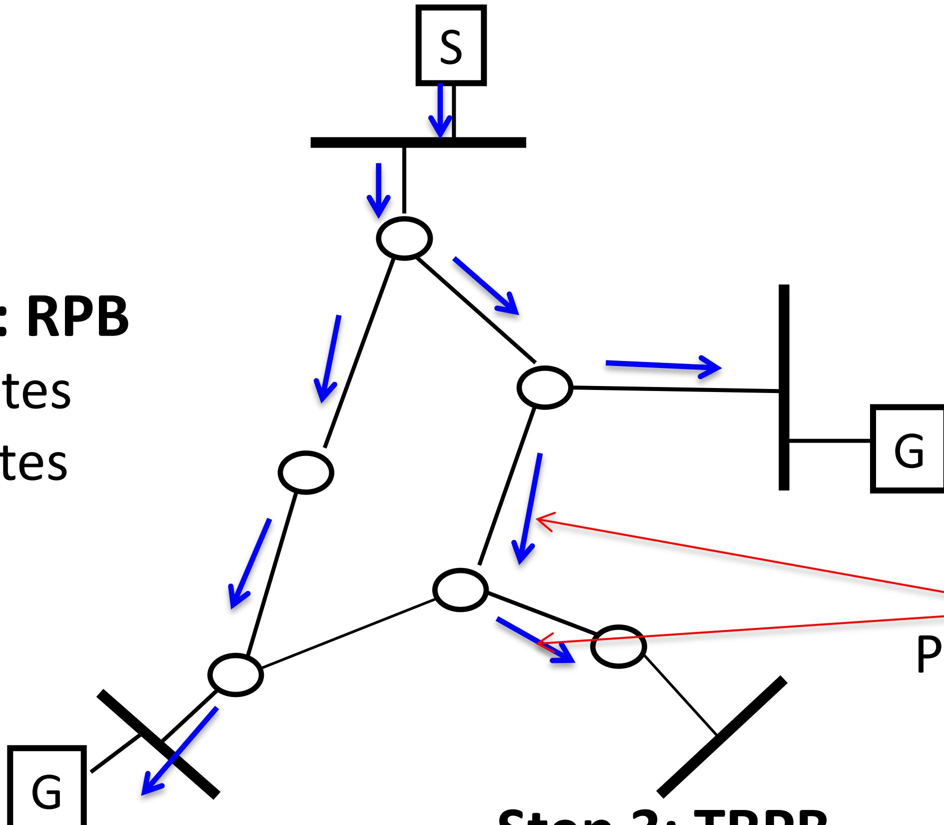
Step 2: RPB
Eliminates duplicates



Step 3: TRPB
Truncate leaves with no members

Between Routers

Step 1:
Reverse Path Flooding
If a packet from S arrives via the shortest path **to** S, then forward on all but the incoming link



Step 2: RPB
Eliminates duplicates

Step 3: TRPB
Truncate leaves with no members

Step 4: RPM
Prune unnecessary branches

What about new joins?

Deering88 Recap

- Simple extensions to existing routing protocols
- Scalability?
  - Think about it
  - Then read the CBT paper ...

Next Lecture

- QoS (IntServ / Diffserv)

- Active Networks

Reminder: submit your paper preferences

Submit reviews on time!

- The author could have dived more deeply into ....how internet service providers actually function as points of centralization in a general decentralized framework, which would conflict with one of the core design principles. – Arjun
- Clark mentions "soft state" as a potential next step. What work has been done on this and how effective has it been? – Richard
- I am very interested to see the different considerations of fate-sharing model and replicate model under the scope of modern network usage. – Yiheng
- the options of replication with “fate-sharing.” – Saurav
- whether the "flow" idea mentioned in the conclusion section has been explored. – Zizheng
  it would be interesting to talk about the concept of "flow" the paper proposed - does this violate the end to end principle? Is it scalable? Practical? – Aditya R.

difficulties arose both because the goal of the architecture was not to constrain performance, but to permit variability, and secondly (and perhaps more fundamentally), because there seemed to be no useful formal tools for describing performance.

On the other hand, some of the most si problems with the Internet today relate to lack of sufficient tools for distributed management, especially in the area of routing. In the large internet being currently operated, routing decisions need to be constrained by policies for resource usage. Today this can be done only in a very limited way, which requires manual setting of tables. This is error-prone and at the same time not sufficiently powerful. The most important change in the Internet architecture over the next few years will probably be the development of a new generation of tools for management of resources in the context of multiple administrations.

performance constraints must be added to the specification, and to specify them properly to achieve a realization which provides the required types of service. We do not have a good idea how to offer guidance in the architecture for the person performing this task.