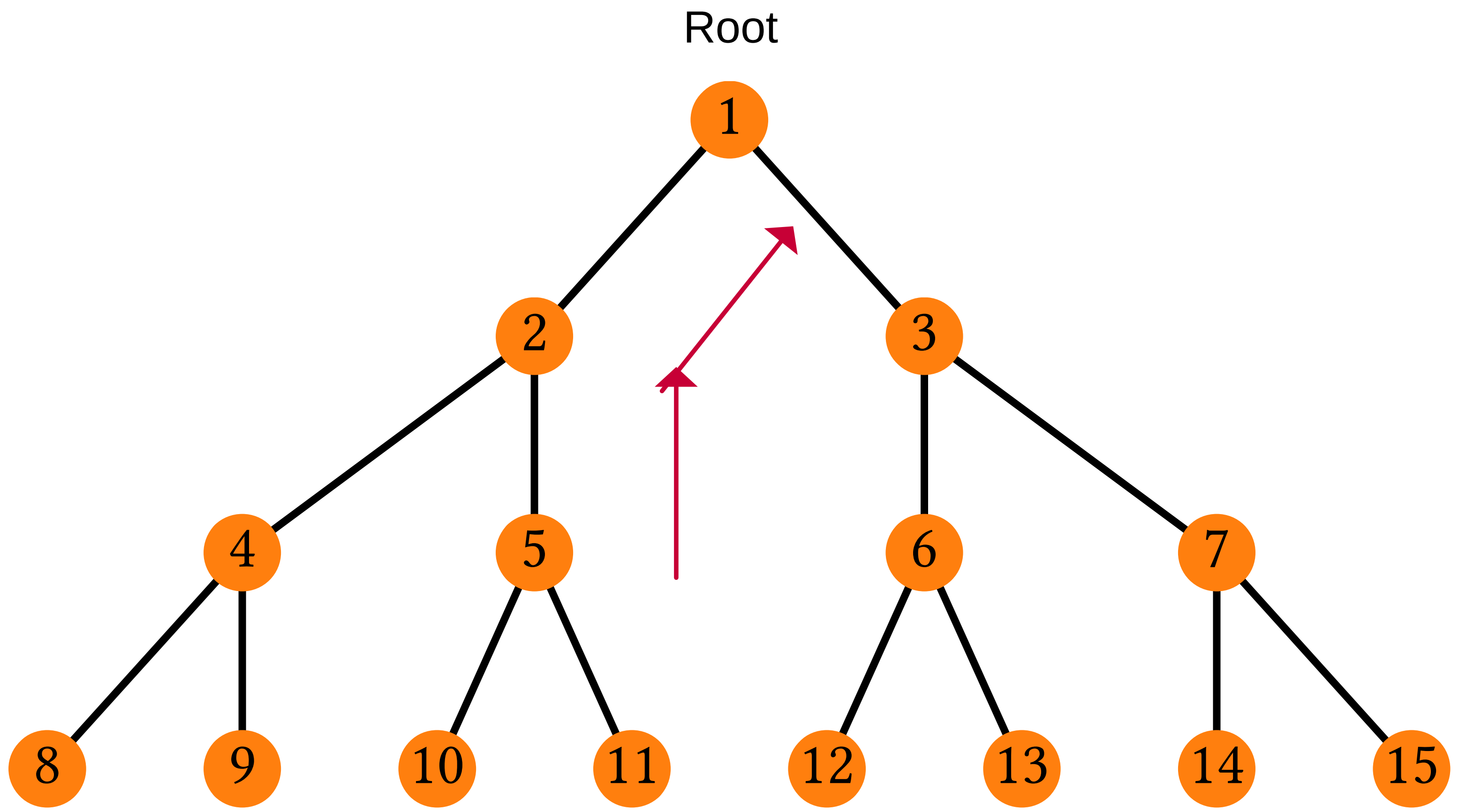
Source Routing for Downward Data Traffic

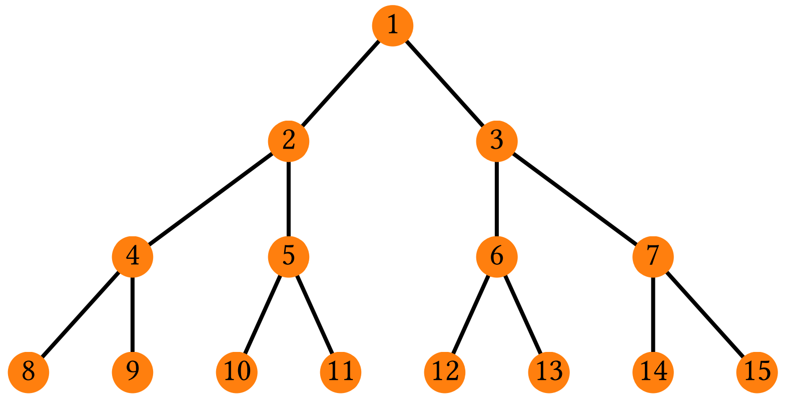
WSN Lab final project, 2017-2018

Upward traffic

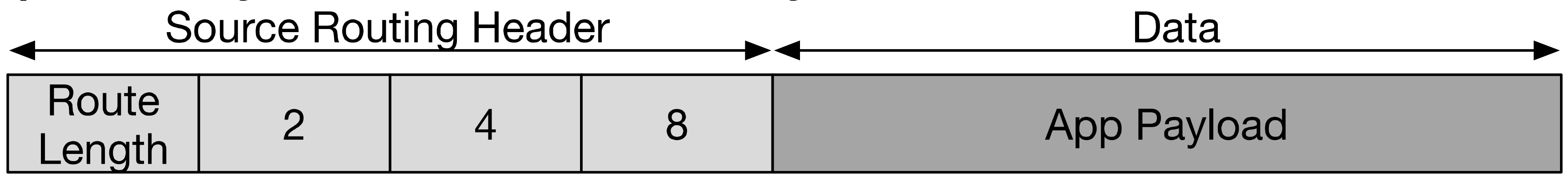


Many-to-one, up the tree

Source Routing – overview



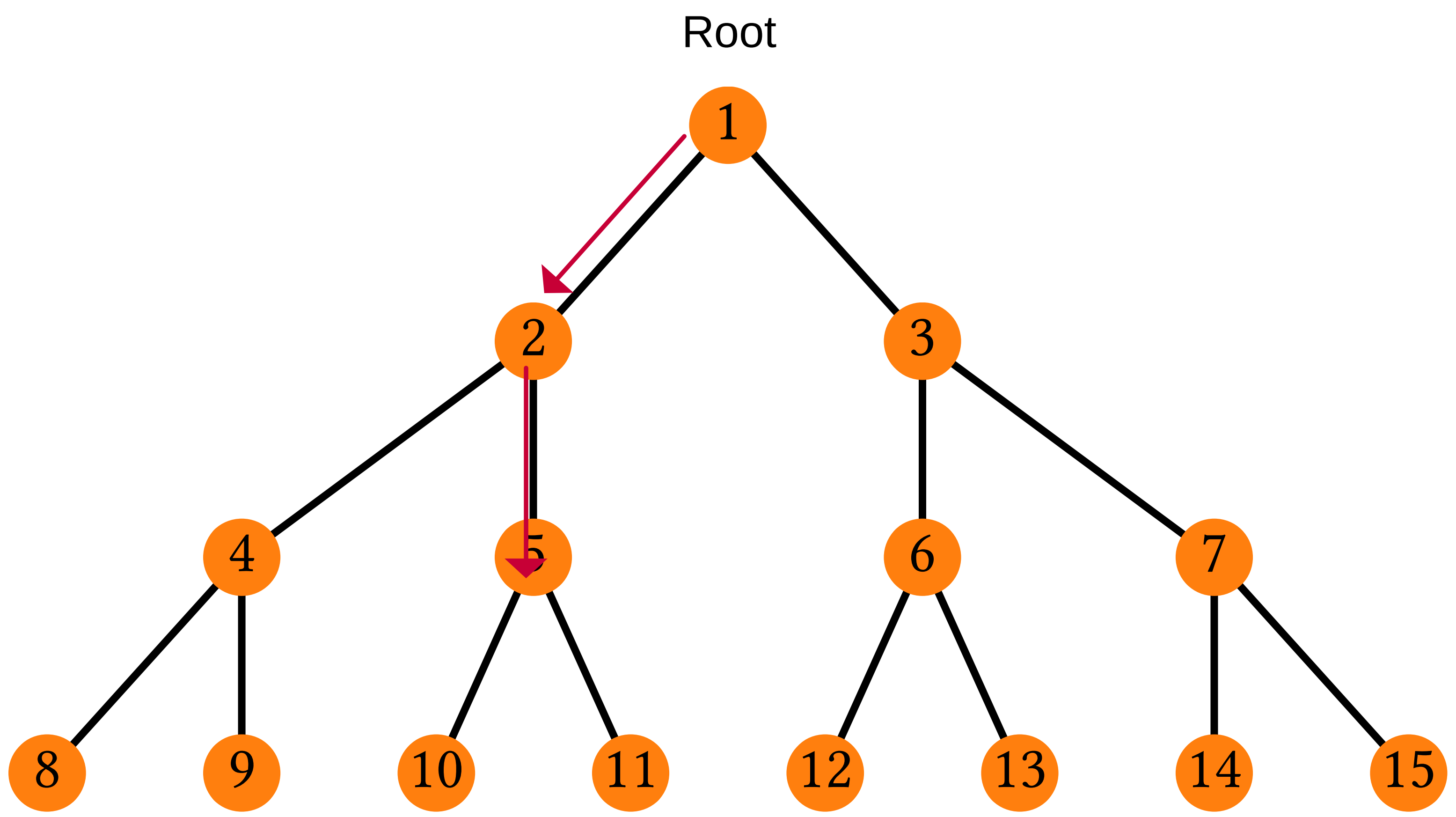
- Source specifies the entire packet route: complete path from source to destination
- Computed by the root node only



- Intermediate nodes just forward to specific next hop:

#2 would look at path in header and forward to **#4**, and so on until the packet reaches the destination node **D = 8**

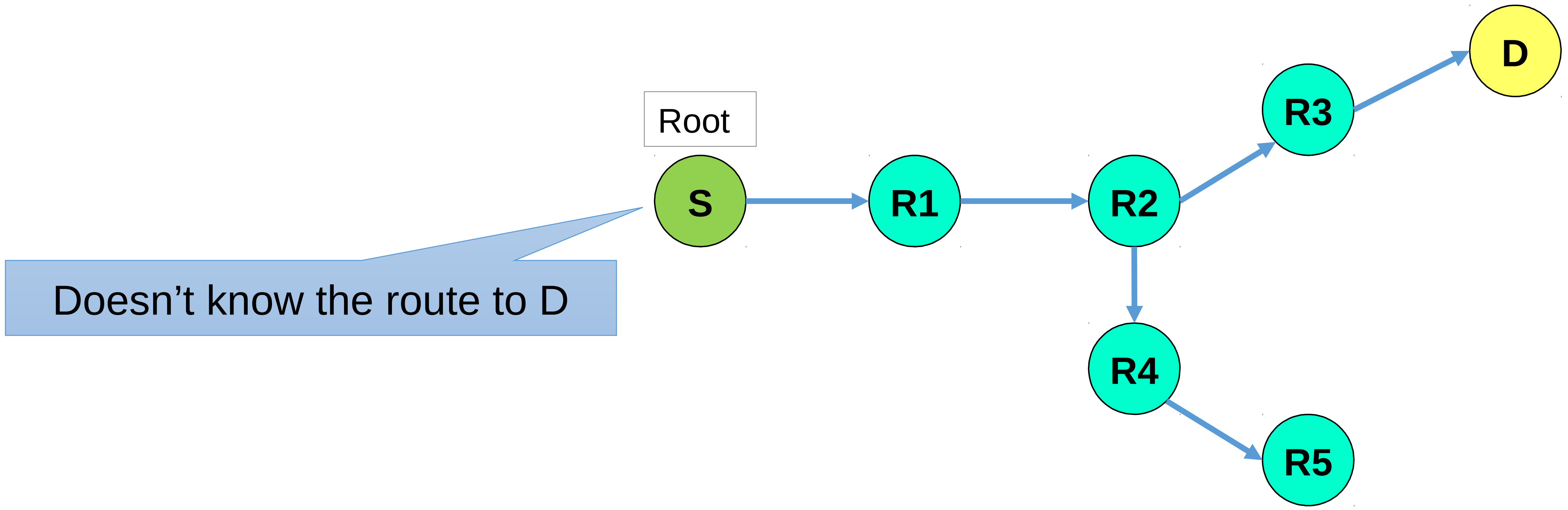
Downwards Source Routing – overview



Enable the sink to send data down the tree

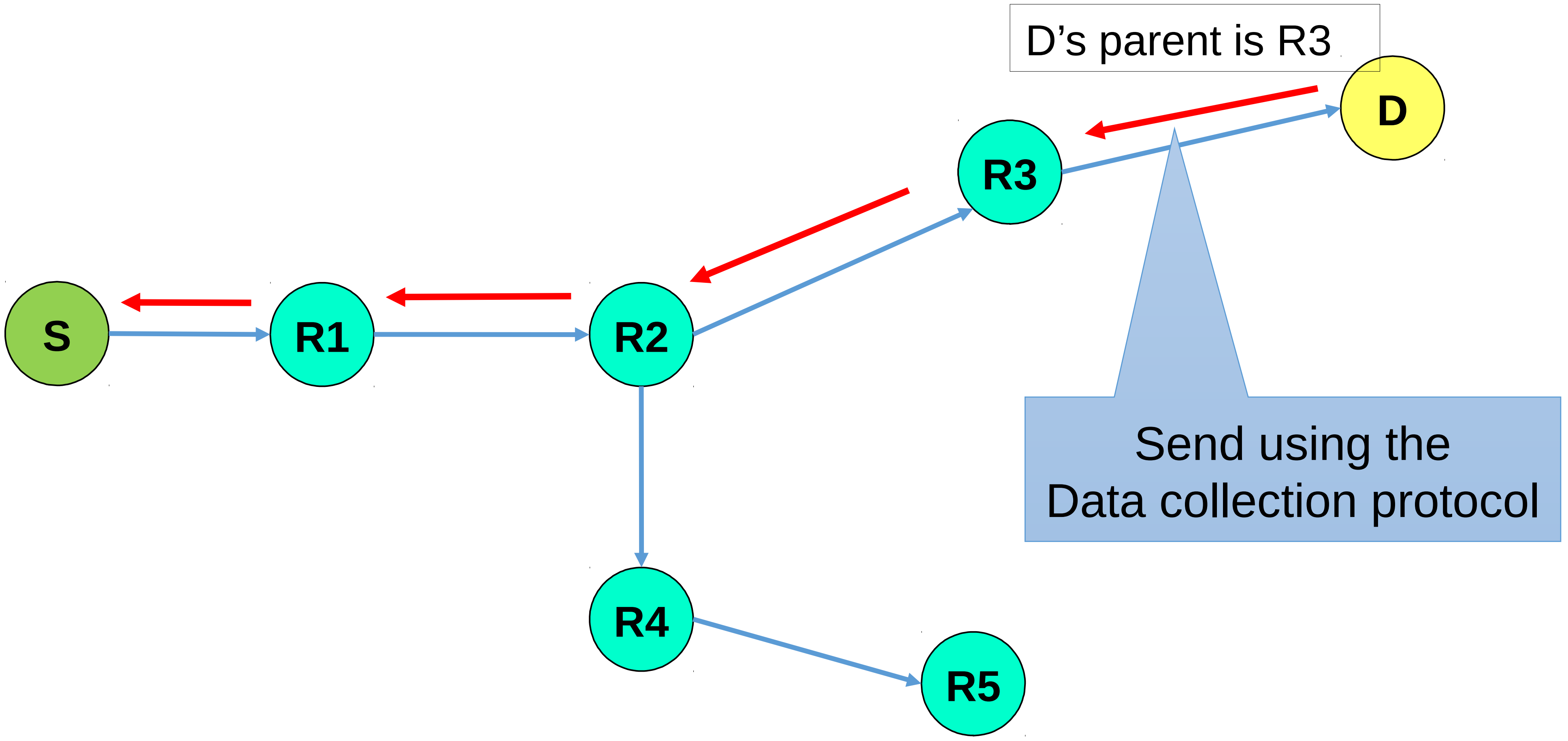
To begin with:

- Objective: Node S wants to send data to node D



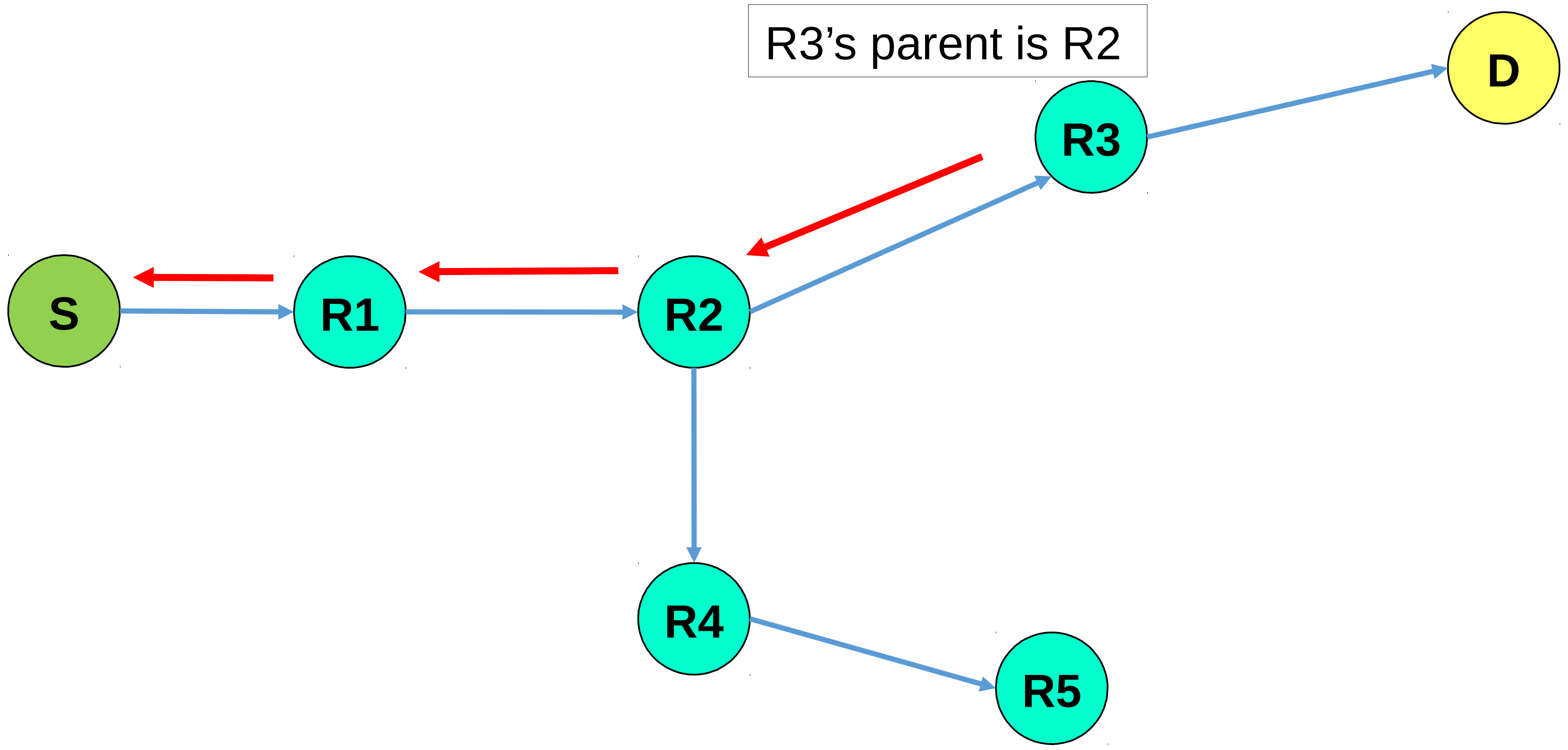
Phase 1: Collecting the routing information

Child	Pare nt
D	R3



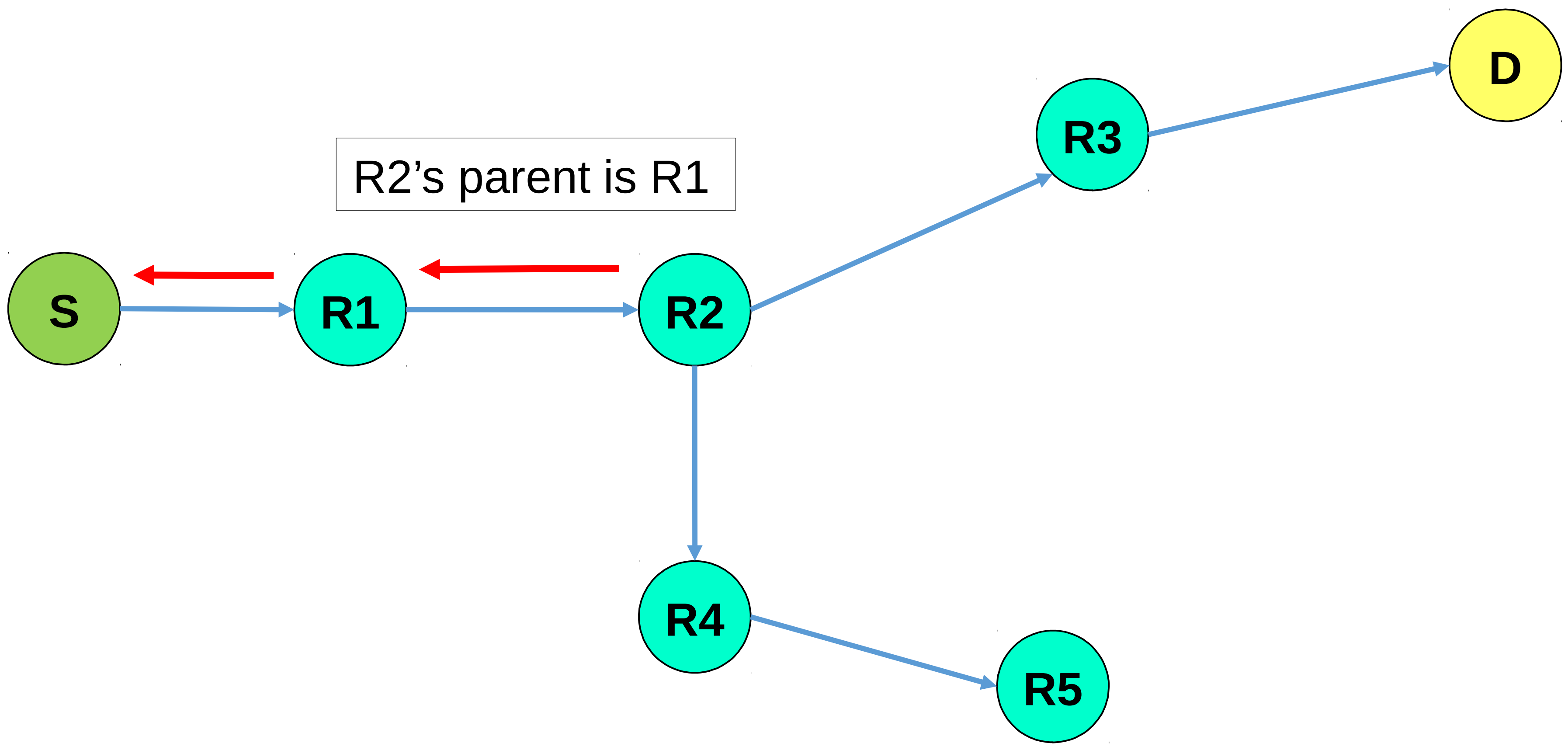
Phase 1: Collecting the routing information

Child	Parent
R3	R2
D	R3



Phase 1: Collecting the routing information

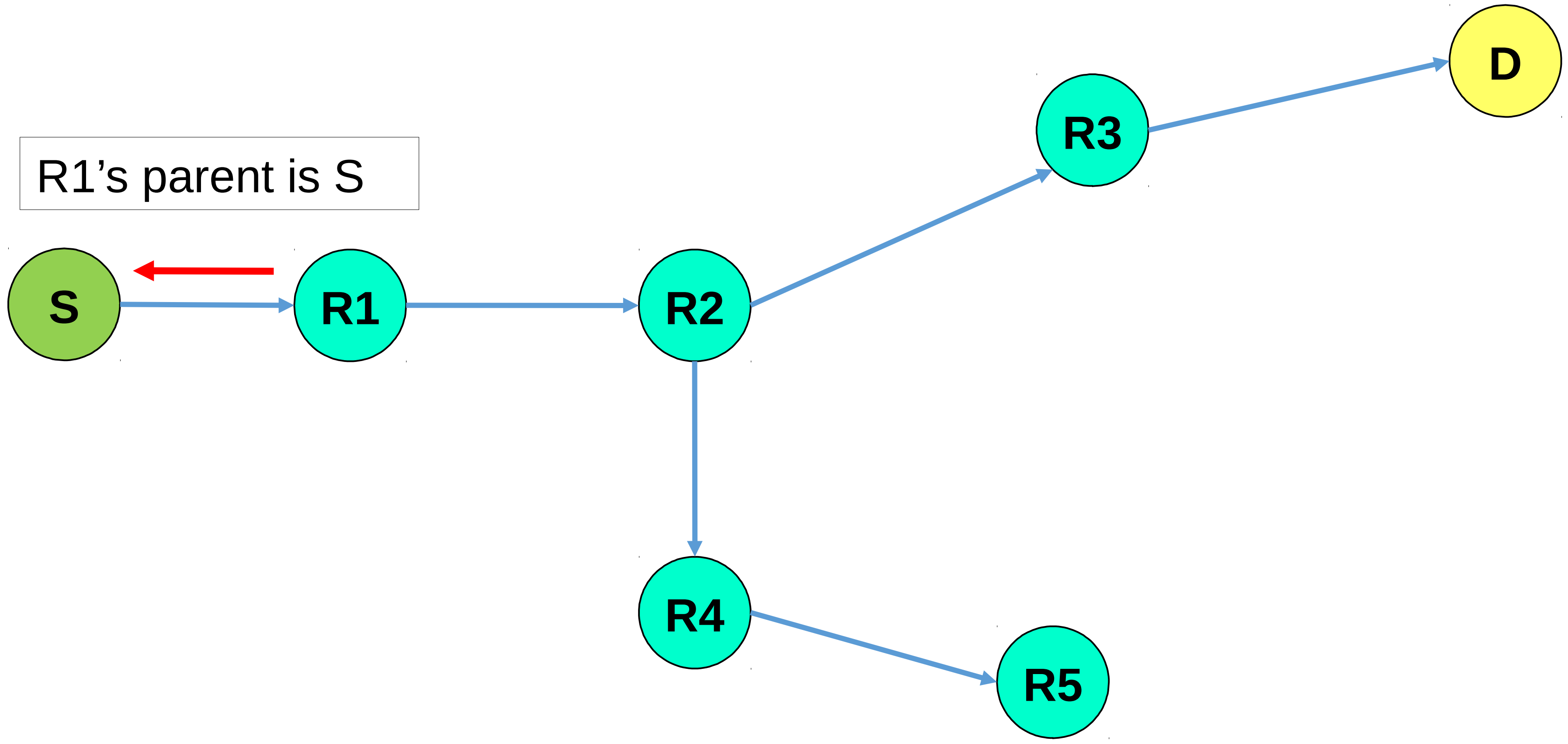
Child	Pare nt
R2	R1
R3	R2
D	R3



Phase 1: Collecting the routing information

Child	Pare nt
R1	S
R2	R1
R3	R2
D	R3

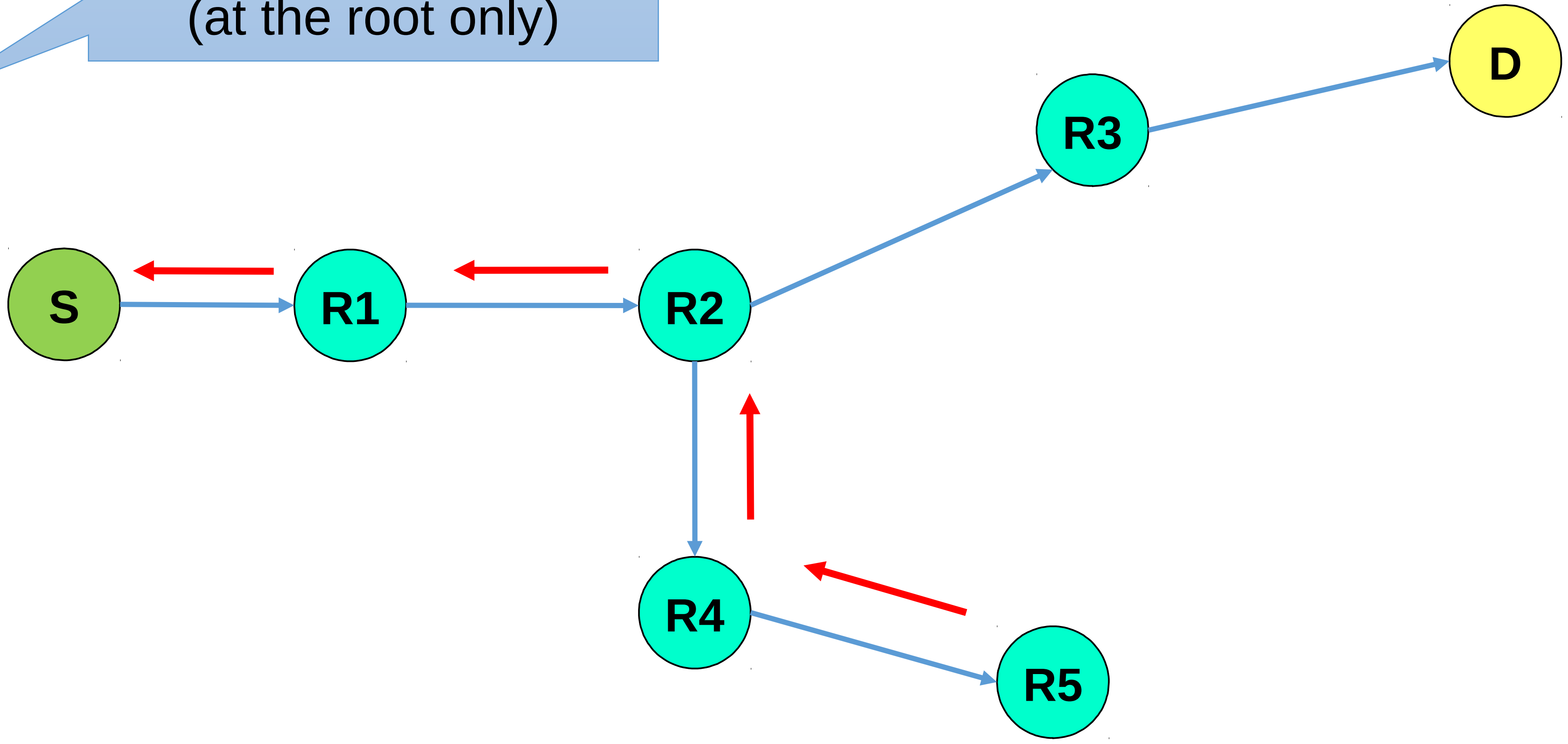
R1's parent is S



Phase 1: Collecting the routing information

Source routing table (at the root only)

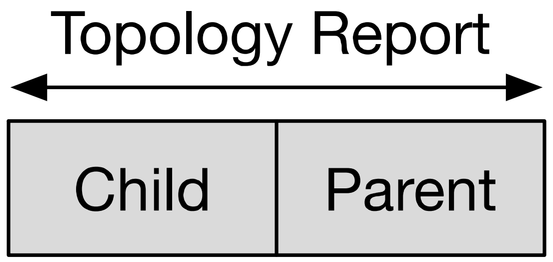
Child	Pare nt
R1	S
R2	R1
R3	R2
D	R3
R5	R4
R4	R2



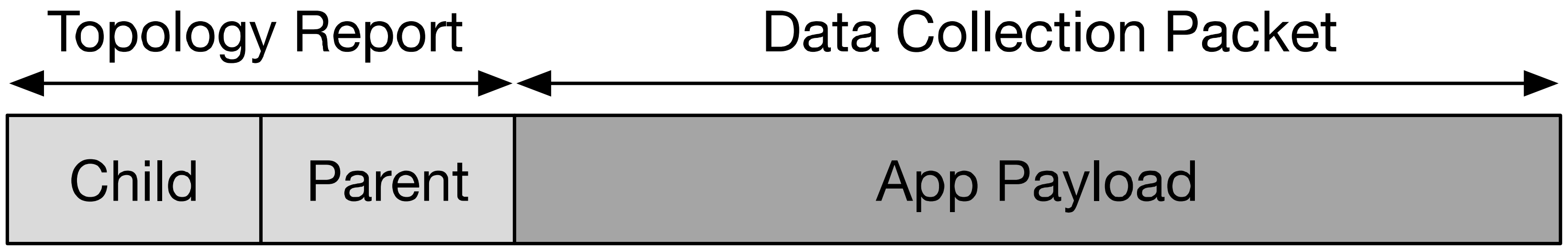
Topology Reports

Child	Pare nt
R1	S
R2	R1
R3	R2
D	R3
R5	R4
R4	R2

- **Dedicated Topology Reports:**
  - Dedicated control traffic to inform the sink of the parent of each node



- **Piggybacking:**
  attach topology reports to data collection packets
  - When the application sends a packet, e.g., using: **my_collect_send()**
  - Include a header with the parent of the sender



When Sending Packets

- When node **S** sends a data packet to **D**, it checks a source routing table. If route exists, then entire route is included in the packet header
  - Hence the name source routing

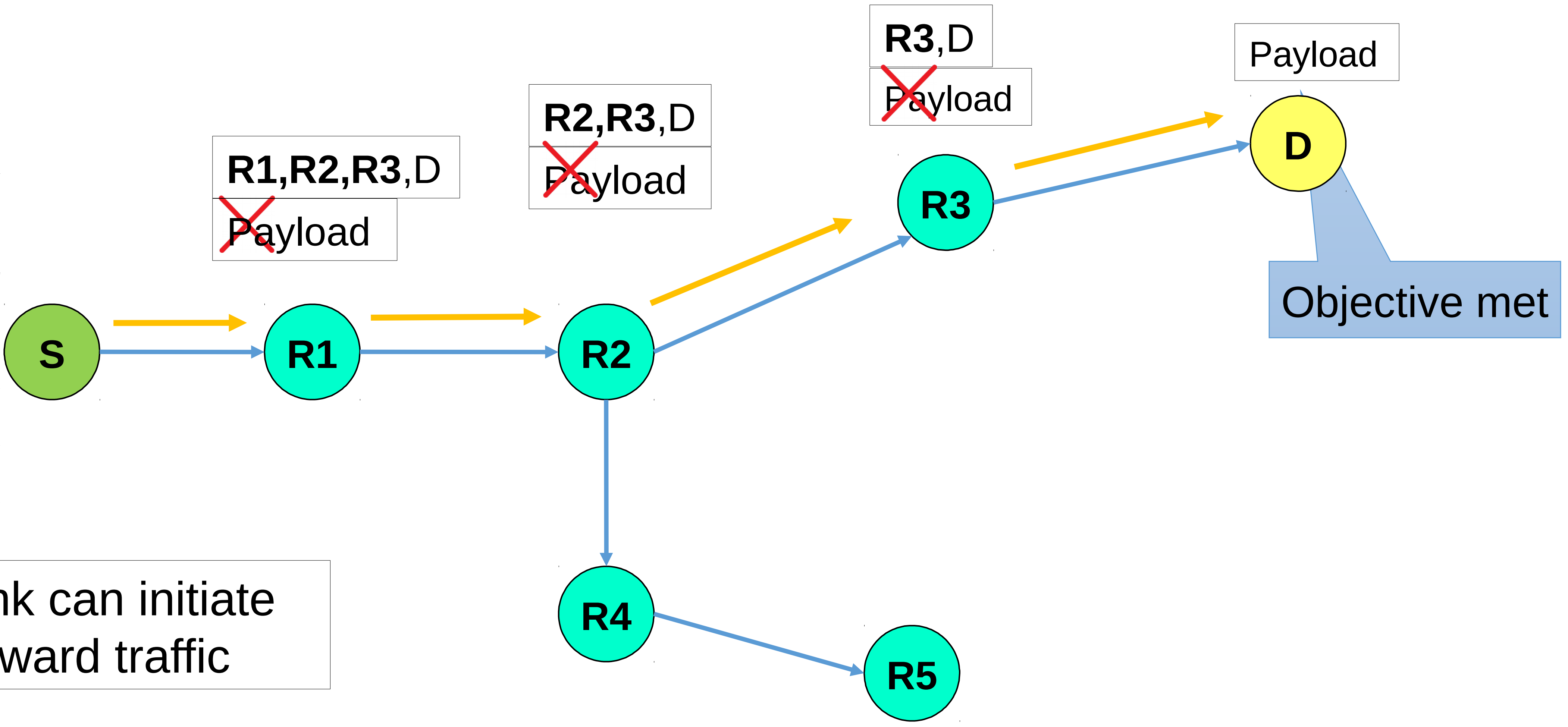
Child	Parent
R1	S
R2	R1
R3	R2
D	R3
R5	R4
R4	R2

Algorithm:

1. Assign N:=D
2. Search for node N in the table to find N's parent P
3. If N is not found or a loop is detected, drop the packet
4. If P ==root, transmit the packet to next-hop node N
5. Else add N to the source routing list of the packet, assign N:=P, go to step 2

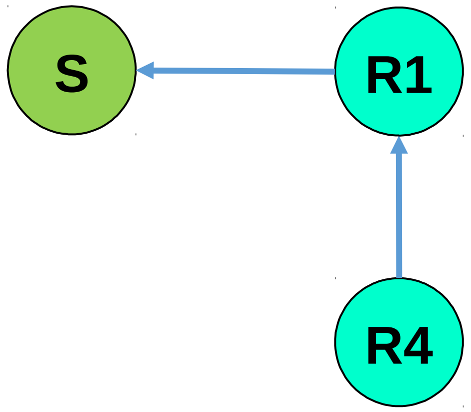
Source Routing - Phase 2: Data Delivery

Child	Parent
R1	S
R2	R1
R3	R2
D	R3



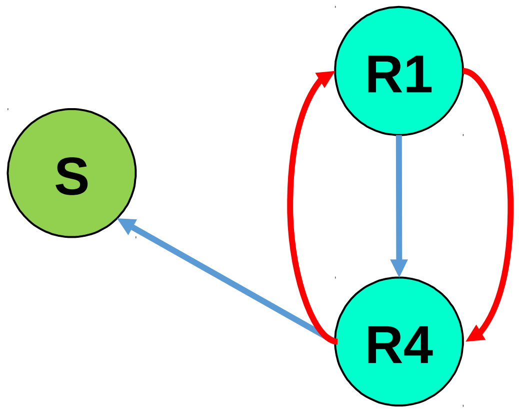
Only sink can initiate downward traffic

Source Routing – Loop detection



Child	Parent
R1	S
R4	R1

If the message "**R4's parent is S**" is lost, we have a loop:



Child	Parent
R1	R4
R4	S

Intended record

Child	Parent
R1	R4
R4	R1

Stale record

Program Structure

- Your program should enable:
  - **Many-to-one data collection** (you already have)
  - **One-to-many data delivery** from the sink to whichever network node

  - The **one-to-many interface** should provide two main functions:
    - **Send Function:**
      int **sr_send**(struct my_collect_conn *c, const linkaddr_t *dest);
    - **Recv Callback:**
      void **sr_recv**(struct my_collect_conn *c, uint8_t hops);

  - The top-level application should use these functions