The Control Plane: Link State Routing

In this lesson, we'll study link state routing.

-Another way to create a routing table with the most efficient path between two routers or 'nodes' is by using **link-state** routing.

Link state routing works in two phases: **reliable flooding** and **route calculation**. Let's look at phase I now.

Phase I: Reliable Flooding
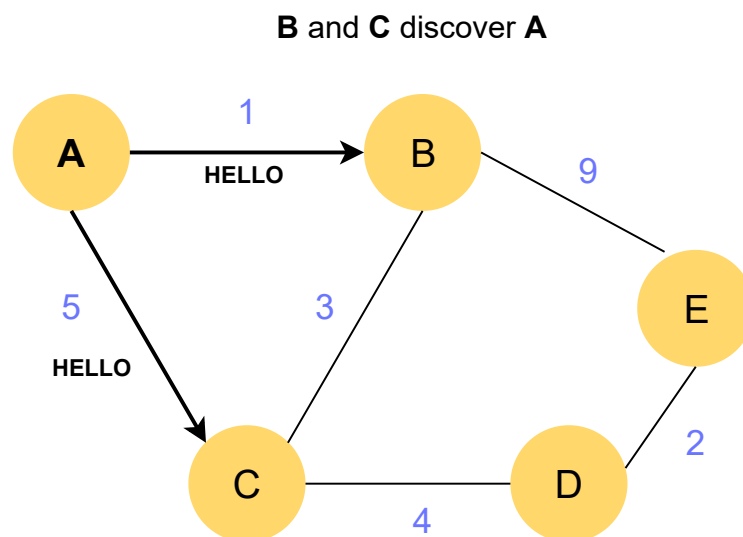
Neighbor Discovery

When a link-state router boots, it first needs to discover its neighbors. To do this:

- Each router sends a **HELLO** message every N seconds on all of its interfaces. This message contains the router's unique address.

- As its neighboring routers also send **HELLO** messages, the router discovers its neighbors.

- These **HELLO** messages are only sent to neighbors that are **directly**

connected to a router, and a router never forwards the **HELLO** messages that they receive.

- **HELLO** messages are also used to detect link and router failures. A link is considered to have failed if no **HELLO** message has been received from the neighboring router for a period of $k \times N$ seconds.
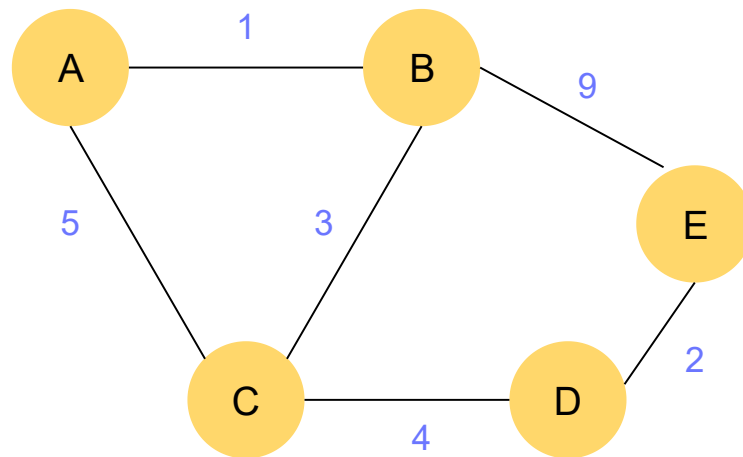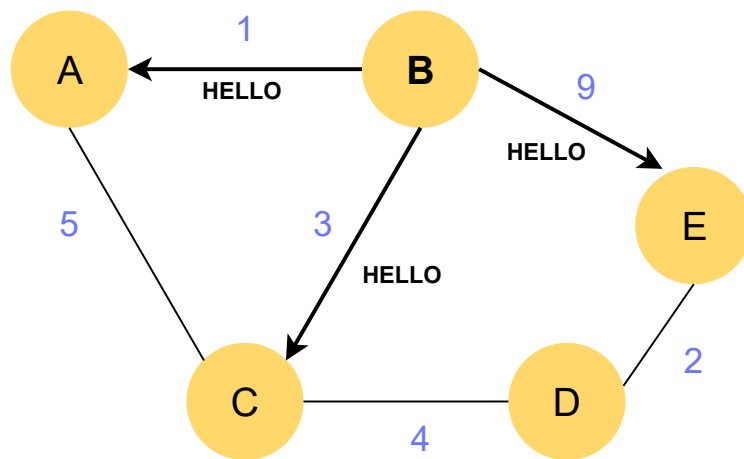
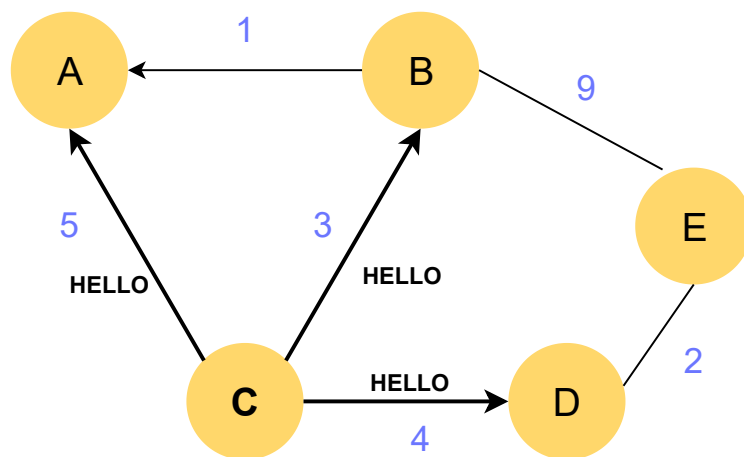**B** and **C** discover **A**

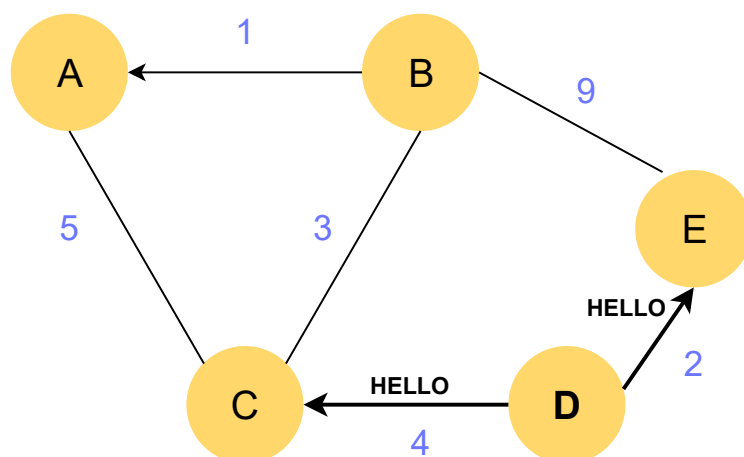## A, C and E discover B

A, B and D discover C

E and C discover D

A ←— 1 —— B

9

HELLO

E

HELLO

5   3

2

C —— 4 —— D
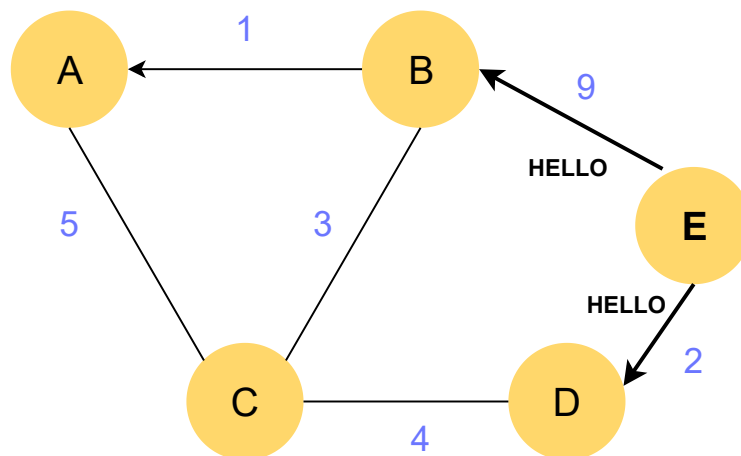
LSPs

Once a router has discovered its neighbours, it must reliably distribute its local links to all routers in the network to allow them to compute their local view of the network topology. For this, each router builds a link-state packet (LSP) that contains the following information:

- `LSP.Router` : identification (address) of the sender of the LSP
- `LSP.age` : age or remaining lifetime of the LSP
- `LSP.seq` : sequence number of the LSP
- `LSP.Links[]` : links advertised in the LSP. Each directed link is represented with the following information:
  - `LSP.Links[i].Id` : identification of the neighbour
  - `LSP.Links[i].cost` : cost of the link

Flooding Algorithm

The routers will construct their routing tables based on the LSPs.

The **Flooding Algorithm** is used to efficiently distribute the LSPs of all routers. Each router that implements flooding maintains a **link state database (LSDB)** that contains the most recent LSP sent by each router. When

a router receives an LSP, it:

- Verifies whether this LSP is already stored inside its LSDB. If so, the router has already distributed the LSP earlier and it **does not need to forward it**.
- Otherwise, **the router forwards the LSP on all links** except the link over which the LSP was received.

To ensure that all routers receive all LSPs, even when there are transmission errors, link state routing protocols use **reliable flooding**, which involves acknowledgments, and if necessary, retransmissions to ensure that all link state packets are successfully transferred to all neighbouring routers.

> **What We Have so Far:** Routers combine the received LSPs with their own LSP to compute the entire network topology.

Then in phase II, the routers apply shortest path algorithms to compute the most efficient path.

Handling Link Failure

- When a link fails, the two routers attached to the link **detect the failure by the lack of HELLO messages** received in the last k×N seconds.

- Once a router has detected a local link failure, it generates and **floods a new LSP that no longer contains the failed link**, and the new LSP replaces the previous LSP in the network.

- As the two routers attached to a link do not detect this failure exactly at the same time, the status of the link may be advertised differently by one of the routers.

Two-way Connectivity Check

- When a link is reported in the LSP of only **one** of the attached routers, the rest consider the link to have failed in **both directions** and remove it from the directed graph that they compute from their LSDB! This check allows link failures to be flooded quickly, as a single LSP is sufficient to announce such bad news.

- Furthermore, a link can only be used once the two attached routers have sent their LSPs.

Handling Router Failure #

- The two-way connectivity check also allows for dealing with **router failures**. When a router fails, all its links fail by definition. Unfortunately, it doesn't send a new LSP to announce its failure. The two-way connectivity check ensures that the failed router is removed from the graph.

- When a router fails, its LSP must be removed from the LSDB of all routers. This can be done by using the **age field** that is included in each LSP. When a router generates an LSP, it sets the `LSP.age` to a value called the LSP's **lifetime** (usually measured in seconds). All routers regularly decrement the age of the LSPs in their LSDBs and an LSP is discarded once its age reaches 0.

Quick Quiz!

In the next lesson, we'll study Dijkstra's algorithm!