



RIP -

das Routing Information Protocol

# Motivation

- Router erstellen ihre Routing-Tabelle selbst, indem sie ihre Nachbarn fragen

-> Distance-Vector-Algorithmus

„Teile deinen Nachbarn mit, wie du die Welt siehst“

# Distance-Vector-Algorithmus

1. Erstelle deine Routing-Tabelle, die anfangs nur deine direkten Nachbarn enthält mit Angabe der Hops (=Anzahl der Weiterleitungen)
2. Erzeuge eine Kopie mit Informationen, welche Router du mit minimalen Hops erreichen kannst und schicke sie an alle Nachbarn
3. Warte auf Kopien deiner Nachbarn und rechne die Informationen in deine Routing-Tabelle mit ein (nur Verbesserungen und Neuigkeiten).
4. Ändern sich dadurch die minimalen Hops, mit denen du einen Router erreichen kannst, fahre mit Schritt 2 fort, sonst Schritt 3

# Das Routing Information Protocol

- Einbettung des Distance-Vector-Algorithmus in ein IP-kompatibles Protokoll
- Annahme:  
eine Verbindung zwischen zwei Routern = 1 Hop = Kosten von 1

# Aufwachen

## ROLLENSPIEL

- Schlüpfen Sie in die Rolle eines Routers und erzeugen sie sich ihre Routing-Tabelle nach dem Routing-Information-Protocol.

→ Siehe Material



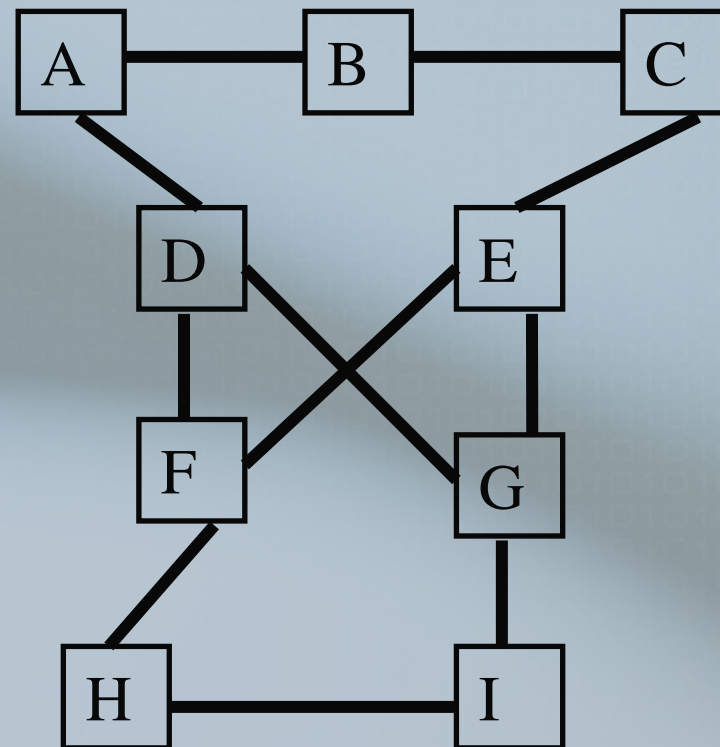
# Diskussion

- Zeit?
- Handling?
- Alle Routing-Tabellen gleich?
- Topologie?
- Test → Pakete werden geroutet

# Überprüfung...

- ... mittels Routing-Programm  
→ HSG-Homepage
- Vergleichen Sie Ihre Tabelle mit dem Ergebnis des [Routing-Programms](#).

# Topologie hier:





# Dynamik

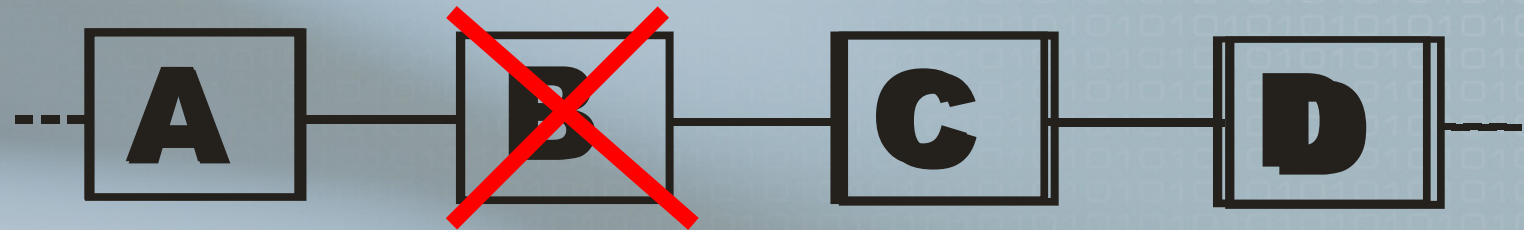
- Bisher: Statische Version
- Realität:
  - Einbinden von Dynamik, d.h.  
Berücksichtigung defekter als auch  
neuer Links

ROLLENSPIEL:

Router E und F verlieren ihren Link

# Probleme?

■ Bsp.:



■ Was passiert?

■ → Routing-Tabellen schauen sich auf

# Count to Infinity

- Das Aufschaukeln der Routing-Tabellen führt zu immer größer werdenden Hops
- Abhilfe:
  - Angabe eines maximalen Hop-Counts
  - Sobald dieser Count to Infinity erreicht ist, geht der Router von einer weggebrochenen Route aus.

# Noch Unklarheiten?

- Wie wärs hiermit ?