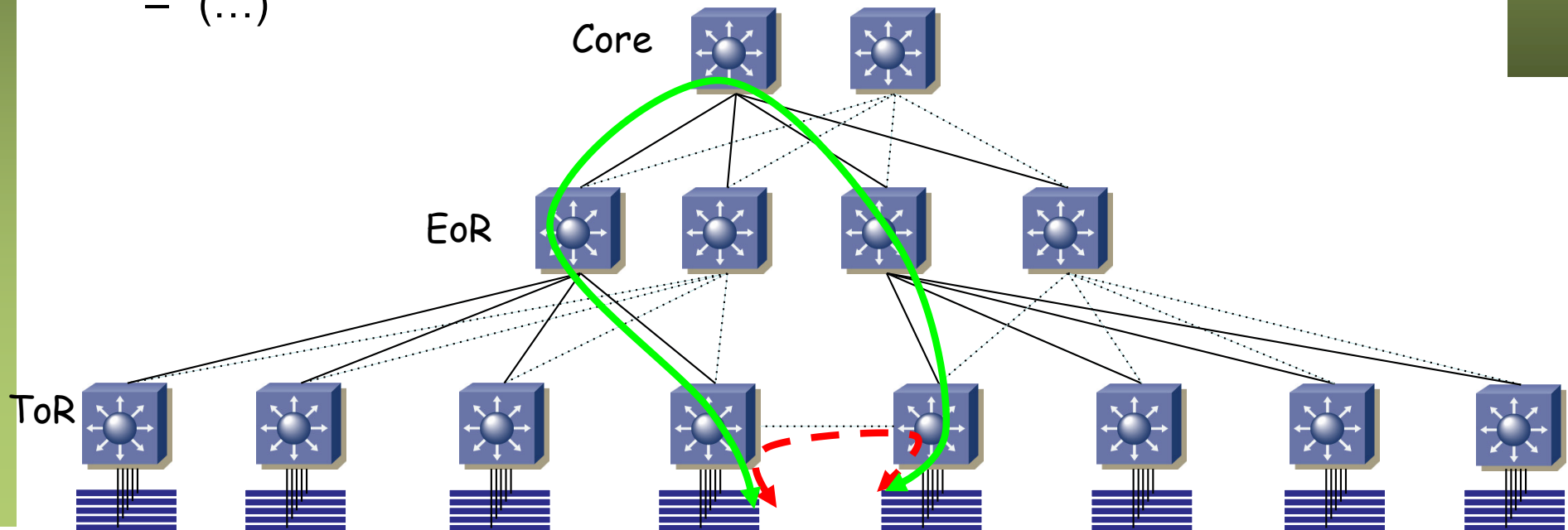


# Alternativas a STP

# STP

- Limitaciones de STP
  - No soporta multipath para una VLAN
  - Multipath entre diferentes VLANs requiere intervención manual
  - El camino es el más corto solo desde la perspectiva del root (...)
  - Largos tiempos de convergencia
  - Peligro de tormentas de inundación
  - Elección de la raíz no es segura
- Mejoras a STP
  - (...)



# STP

- Limitaciones de STP
  - No soporta multipath para una VLAN
  - Multipath entre diferentes VLANs requiere intervención manual
  - El camino es el más corto solo desde la perspectiva del root
  - Largos tiempos de convergencia
  - Peligro de tormentas de inundación
  - Elección de la raíz no es segura
- Mejoras a STP
  - RSTP, MSTP mejoran los tiempos de convergencia pero siguen en el rango de los segundos
  - Hay otras mejoras a la convergencia, muchas veces sin estandarizar (*Loopguard, BPDU guard, Rootguard, BPDU filter, Storm control*)
  - No cambian que STP desactiva puertos para formar un árbol
- Alternativas clásicas a STP
  - (...)

# STP

- Limitaciones de STP
  - No soporta multipath para una VLAN
  - Multipath entre diferentes VLANs requiere intervención manual
  - El camino es el más corto solo desde la perspectiva del root
  - Largos tiempos de convergencia
  - Peligro de tormentas de inundación
  - Elección de la raíz no es segura
- Mejoras a STP
  - RSTP, MSTP mejoran los tiempos de convergencia pero siguen en el rango de los segundos
  - Hay otras mejoras a la convergencia, muchas veces sin estandarizar (*Loopguard, BPDU guard, Rootguard, BPDU filter, Storm control*)
  - No cambian que STP desactiva puertos para formar un árbol
- Alternativas clásicas a STP
  - Multichassis LAG
  - Routing capa 3

# Alternativas a STP

- Conmutación capa 3 no ofrece continuidad en capa 2
- Eso es un problema para ciertas aplicaciones, en especial con funcionalidades de clustering
- Por ejemplo no permite la movilidad de las VMs
- Agregación multichasis está limitada a dominios en el orden de los miles de hosts
- (Habría que pensarse bien si es razonable una LAN con más de miles de hosts...)



# TRILL

# TRILL



- *Transparent Interconnection of Lots of Links*
- IETF, RFC 6325 (Perlman 2011) y otras
- Pretende sustituir a STP
- El conmutador que lo implementa se conoce como un RBridge (Routing Bridge)
- Lo básico
  - Los RBriges y los enlaces o LANs puenteadas que los interconectan forman un “campus” (el dominio de broadcast)
  - Transportan las tramas Ethernet por ese campus encapsulándolas en otras tramas Ethernet (MAC in MAC)
  - Esa cabecera adicional incluye una cuenta de saltos
  - El camino por el campus lo calcula IS-IS (permite ECMP)
  - Se desencapsula en el RBridge de salida hacia el destino
- Está especificado su transporte sobre Ethernet y sobre PPP

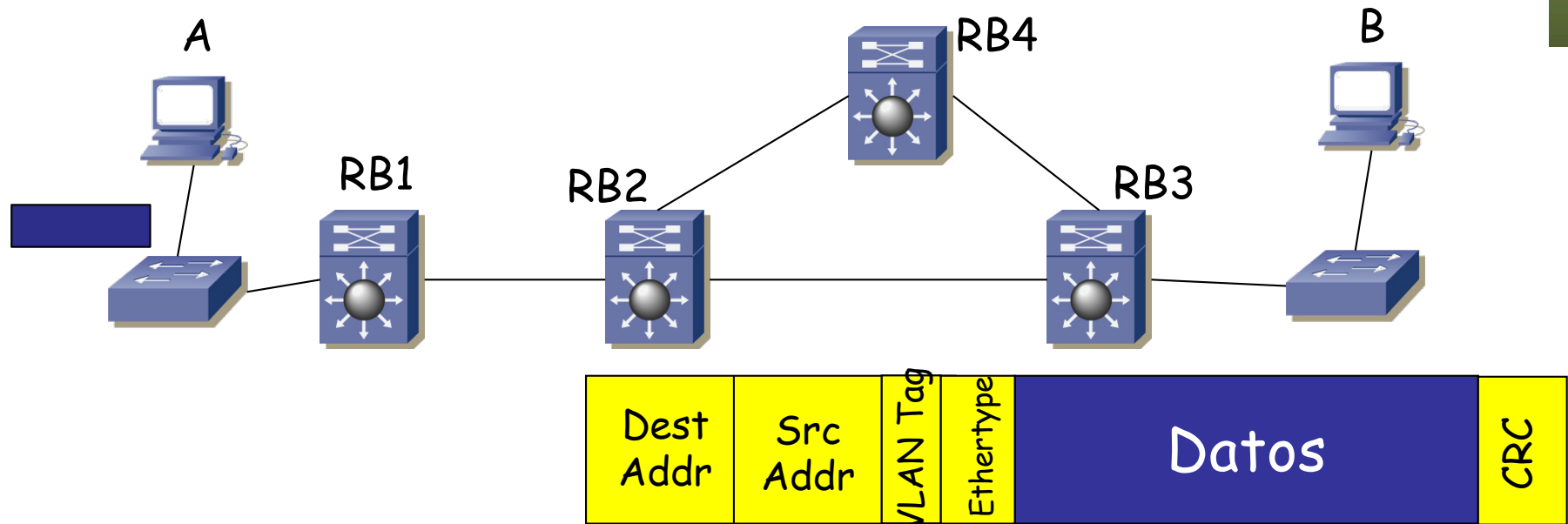
# TRILL data y control paths



# TRILL – Plano de datos

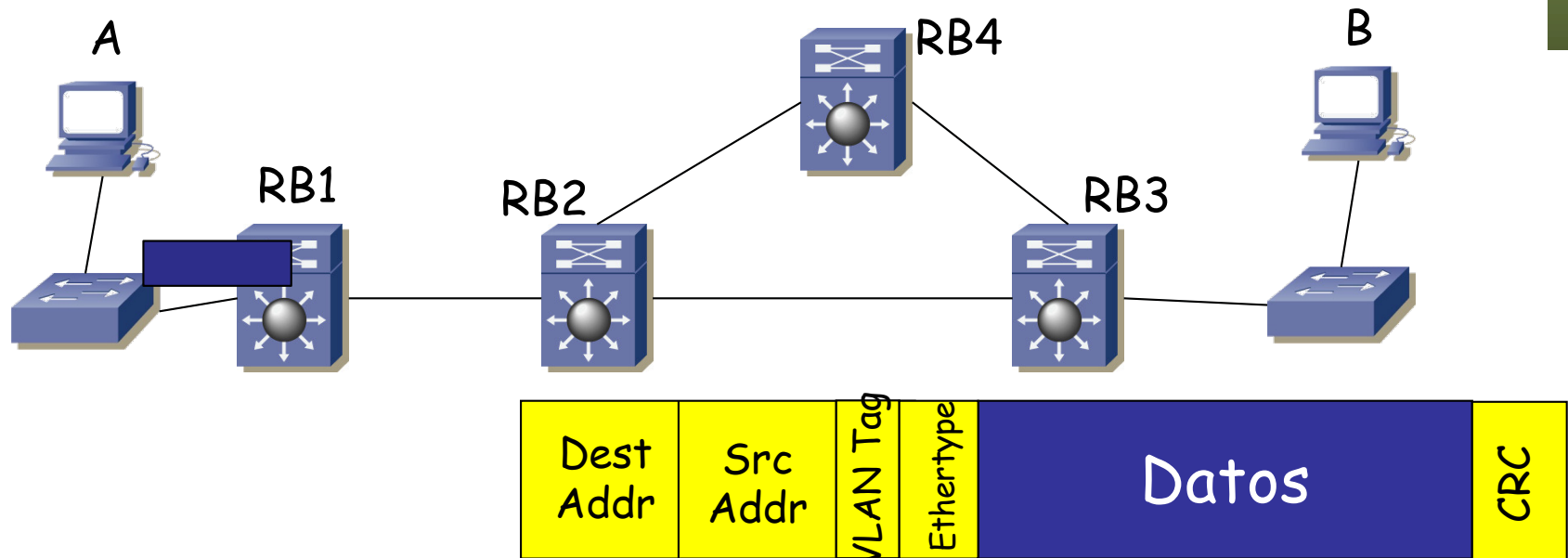
# TRILL data path

- Trama Ethernet original
  - Dirección MAC origen de A
  - Dirección MAC destino de B



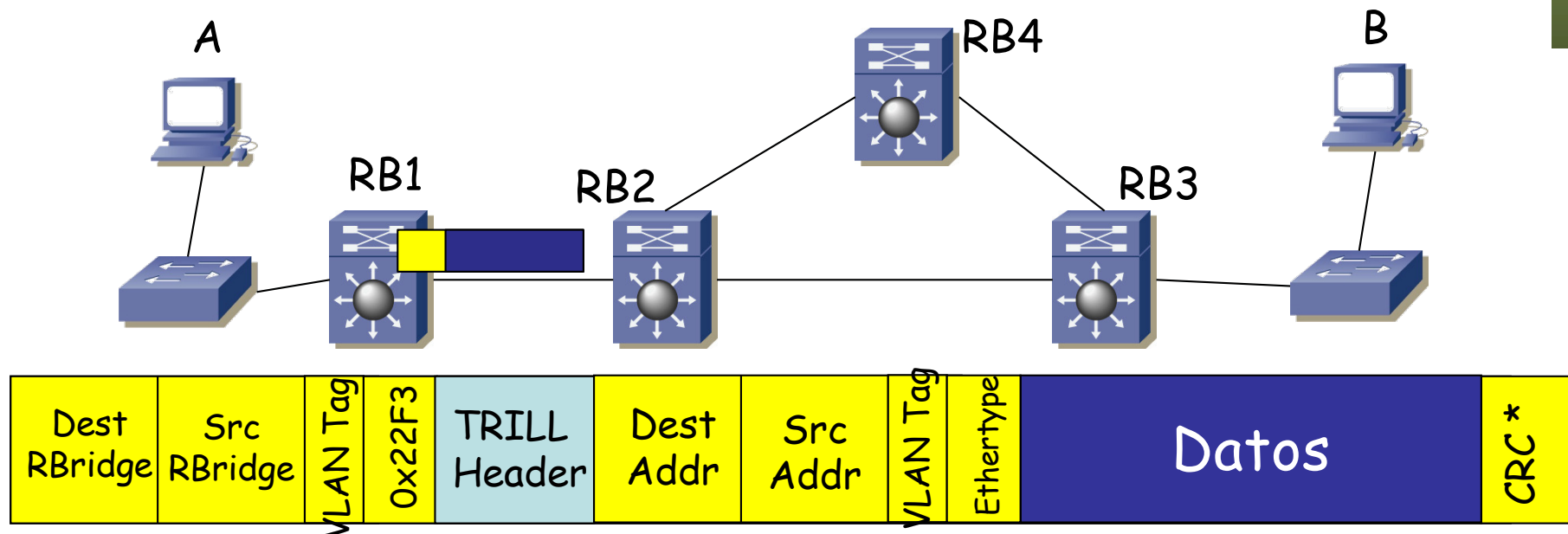
# TRILL data path

- La trama llega a RB1
- Calcula cuál es el siguiente salto en el campus TRILL hacia B
- Encapsula esa trama:
  - (...)



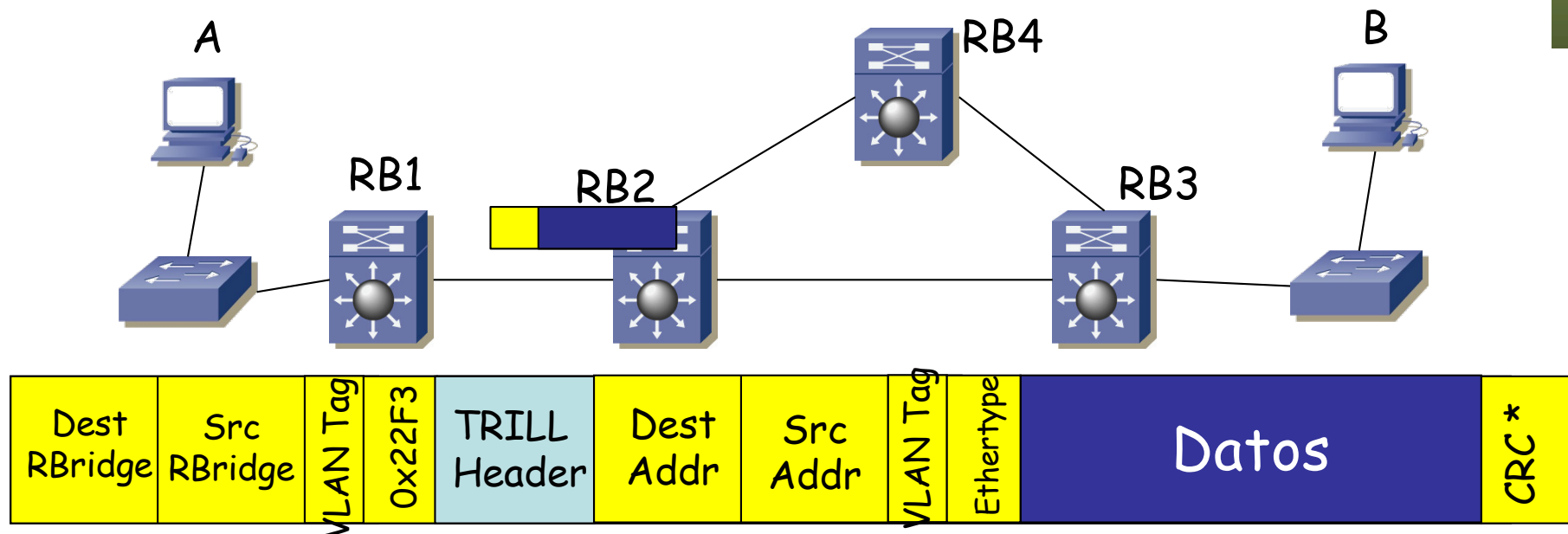
# TRILL data path

- La trama llega a RB1
- Calcula cuál es el siguiente salto en el campus TRILL hacia B
- Encapsula esa trama:
  - Dest RBridge = MAC de RB2
  - Src RBridge = MAC de RB1 en puerto hacia RB2
  - En TRILL header:
    - Egress RBridge Nickname = Nickname de RB3
    - Ingress RBridge Nickname = Nickname de RB1
    - TTL = n



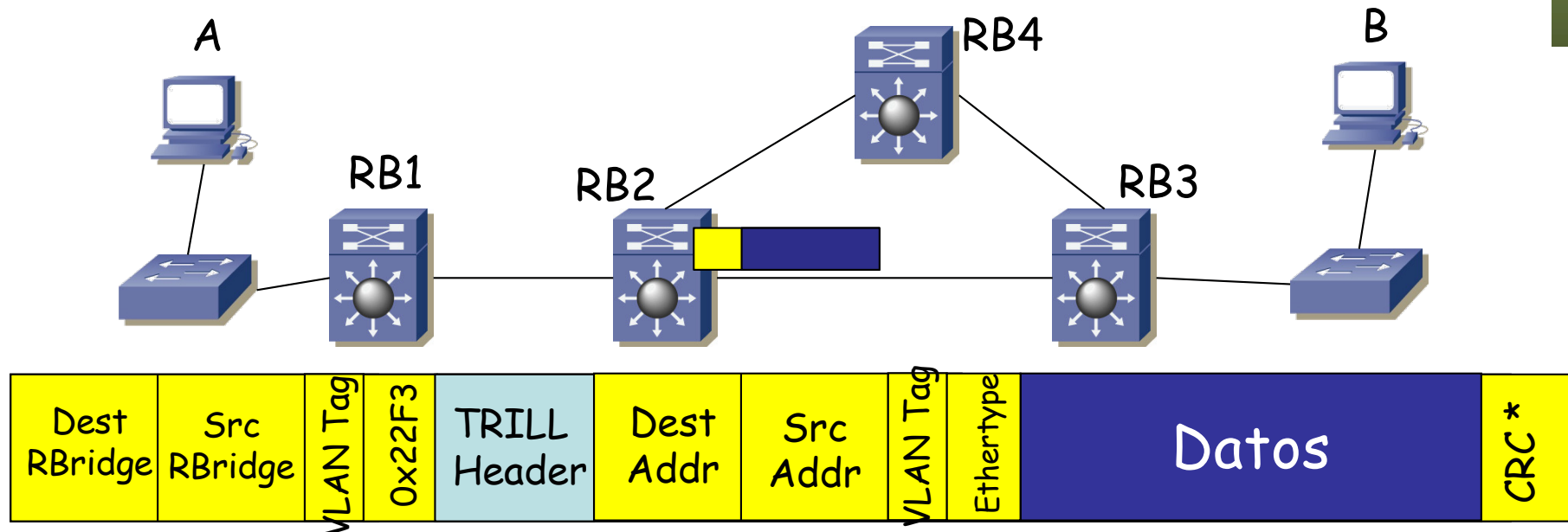
# TRILL data path

- La trama llega a RB2
- Calcula cuál es el siguiente salto en el campus TRILL hacia RB3
- Modifica esa trama:
  - (...)



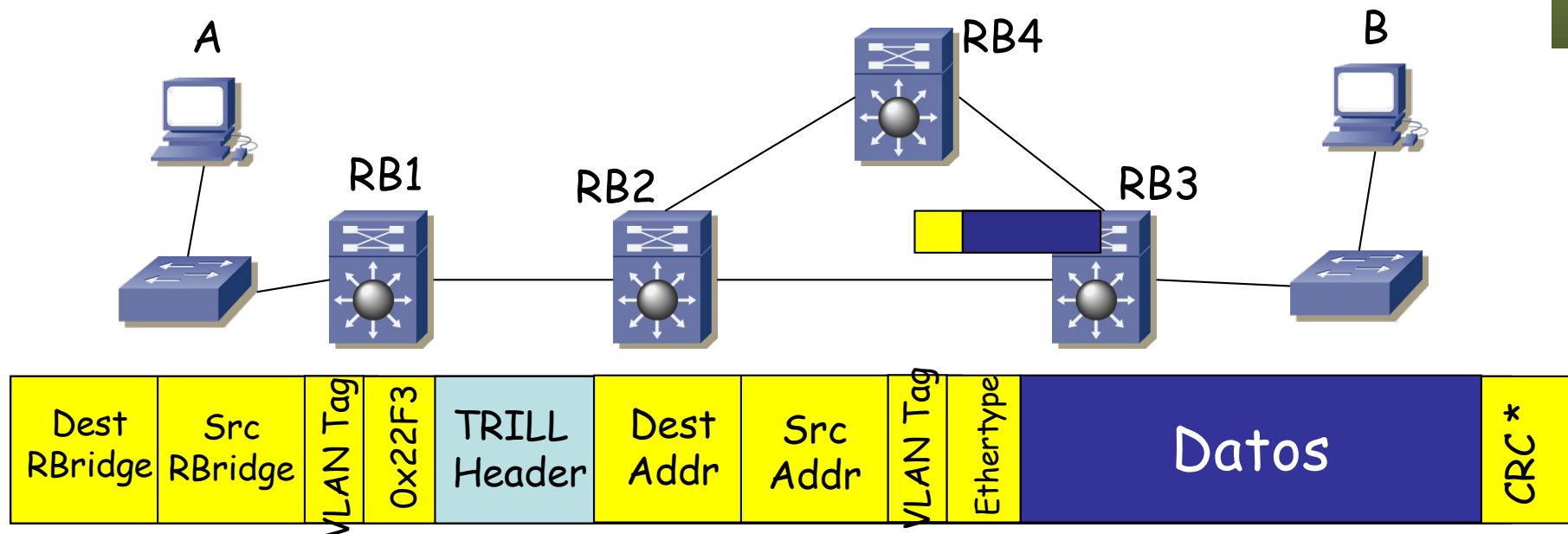
# TRILL data path

- La trama llega a RB2
- Calcula cuál es el siguiente salto en el campus TRILL hacia RB3
- Modifica esa trama:
  - Dest RBridge = MAC de **RB3**
  - Src RBridge = MAC de **RB2** en puerto hacia RB3
  - Egress RBridge Nickname = Nickname de RB3 (no cambia)
  - Ingress RBridge Nickname = Nickname de RB1 (no cambia)
  - TTL = TTL – 1 (se tira la trama si al recibirla tiene TTL=0)



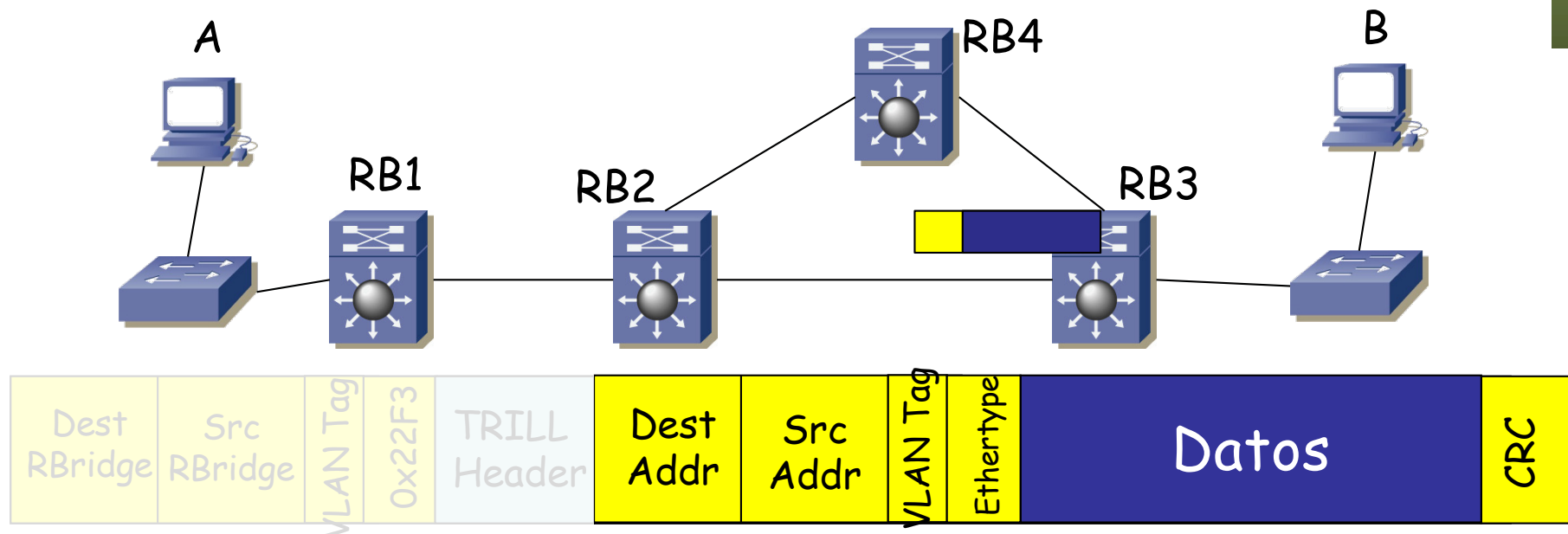
# TRILL data path

- La trama llega a RB3
- Desencapsula esa trama
- Calcula cuál es el siguiente salto en el campus TRILL hacia B
- (...)



# TRILL data path

- La trama llega a RB3
- Desencapsula esa trama
- Calcula cuál es el siguiente salto en el campus TRILL hacia B
- Los RBridges, en cierto modo, se han comportado como routers
- Pero la trama Ethernet ha atravesado el dominio TRILL sin ser modificada



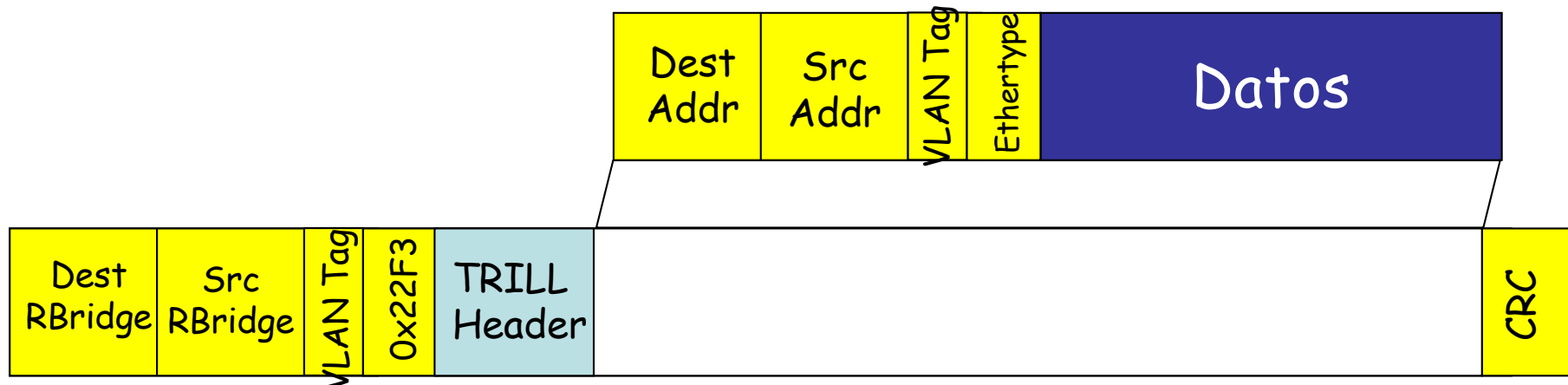


# TRILL – Plano de datos

# TRILL header

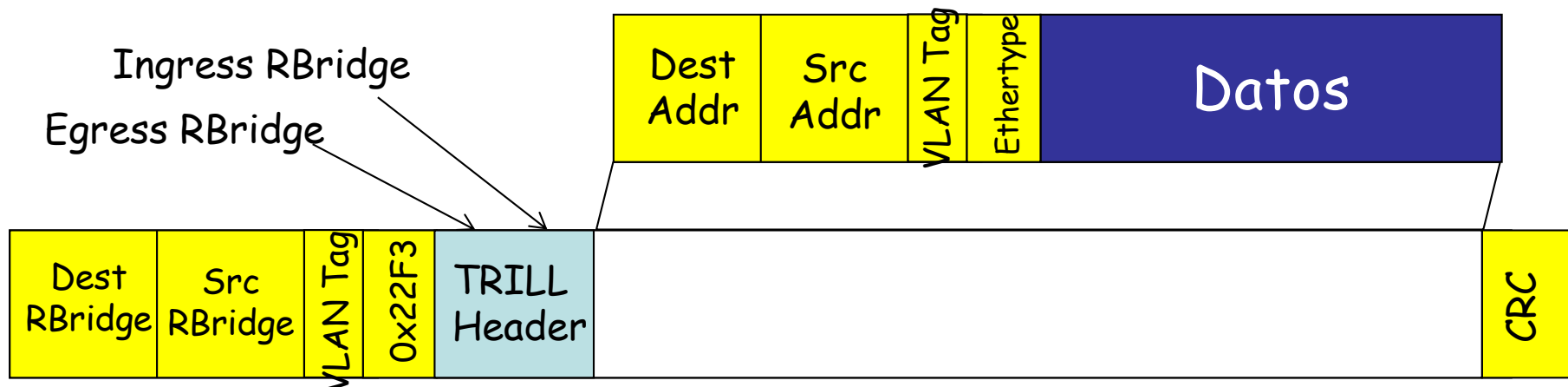
# TRILL sobre Ethernet

- MAC origen y destino son de RBridges adyacentes
- Puede llevar etiqueta de VLAN si los conmutadores del campus TRILL la necesitan
- Ethertype 0x22F3
- TRILL añade su propia cabecera (la vemos más adelante)
- A continuación la trama que ha llegado al RBridge frontera
- Si la trama original no llevaba etiqueta de VLAN se le añade
- Los conmutadores del Campus TRILL no RBridges van a reenviar en base a las direcciones de la cabecera exterior
- RBridges reenvían en función del RBridge Nickname destino



# TRILL sobre Ethernet

- En cada salto entre RBridges las direcciones MAC más exteriores son de los RBridges que envían y reciben esa trama
- Es decir, “Dest RBridge” es la dirección del siguiente salto
- “Src RBridge” es la dirección del salto anterior
- Parecido a que los RBridges fueran routers
- Los RBridges frontera (entrada a la campus y salida) están indicados en la cabecera de TRILL



# TRILL Header

- Nicknames
  - Cada RBridge posee un *nickname* con el que se le hace referencia en las PDUs de TRILL y sirve para identificarlo de cara a IS-IS
  - Los nicknames son números de 2 bytes (máx. 64K RBridges)
  - Los nicknames se eligen mediante un proceso automático con información añadida a los mensajes de IS-IS
- Campos de la cabecera
  - (...)

|                         |                          |   |   |        |     |
|-------------------------|--------------------------|---|---|--------|-----|
| TRILL Ethertype         | V                        | R | M | ExtLng | Hop |
| Egress RBridge Nickname | Ingress RBridge Nickname |   |   |        |     |

# TRILL Header

- Nicknames
  - Cada RBridge posee un *nickname* con el que se le hace referencia en las PDUs de TRILL y sirve para identificarlo de cara a IS-IS
  - Los nicknames son números de 2 bytes (máx. 64K RBridges)
  - Los nicknames se eligen mediante un proceso automático con información añadida a los mensajes de IS-IS
- Campos de la cabecera
  - V = Version (2 bits), R = Reserved (2 bits)
  - M = Multi-Destination (1 bit)
  - ExtLng = Length of TRILL Header Extensions
  - Hop = Hop Limit (6 bits)
  - Egress RBridge Nickname = nickname del RBridge de salida del campus hacia el host destino
  - Ingress RBridge Nickname = nickname del RBridge de entrada al campus de la trama desde el host origen

|                         |                          |   |   |        |     |
|-------------------------|--------------------------|---|---|--------|-----|
| TRILL Ethertype         | V                        | R | M | ExtLng | Hop |
| Egress RBridge Nickname | Ingress RBridge Nickname |   |   |        |     |

# TRILL header

# TRILL – Plano de control



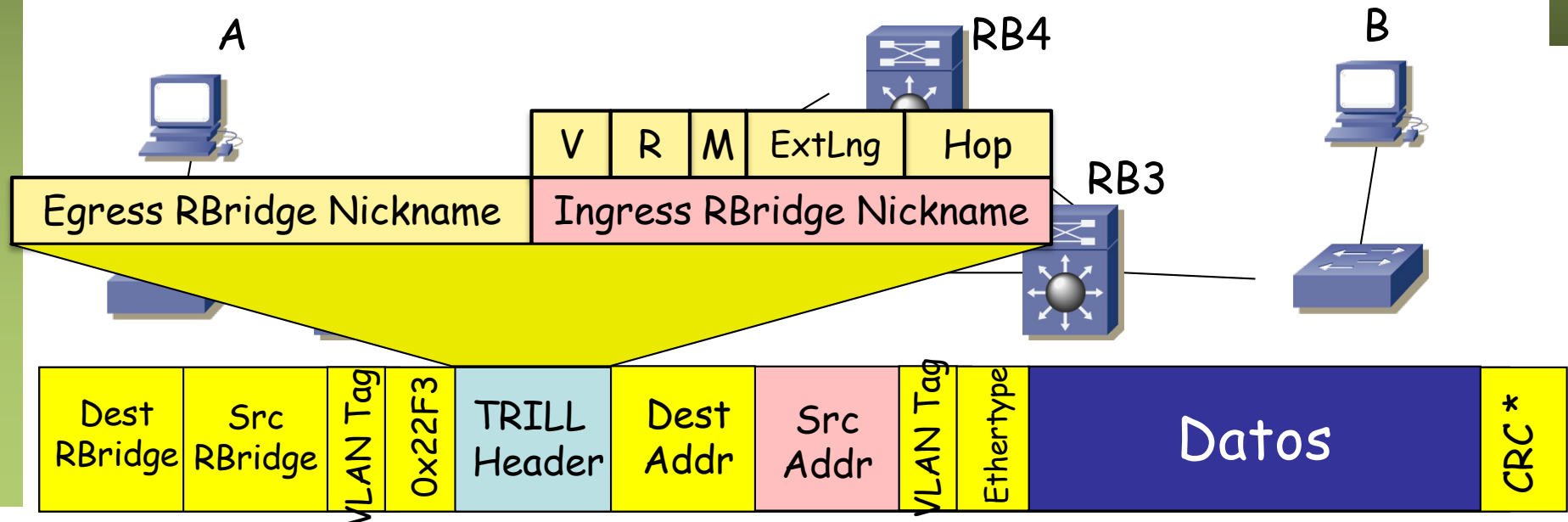
# TRILL control path

- IS-IS directamente sobre el nivel de enlace
- Ethertype 0x22F4
- Todos los mecanismos típicos de un protocolo link-state
- Más añadidos específicos para TRILL (por ejemplo en el tema de routers designados)



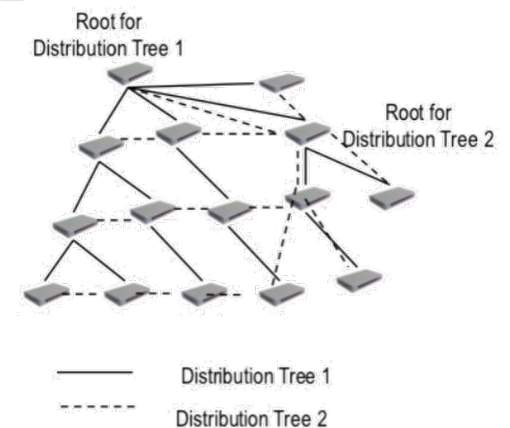
# Aprendizaje

- RBridge frontera aprende direcciones MAC de hosts remotos junto con:
  - RBridge por el que acceden al campus
  - RBridge siguiente salto hacia ese egress RBridge
- Lo hace principalmente en base a los paquetes de TRILL que recibe
- Solo los R Bridges frontera necesitan aprender direcciones MAC de los hosts
- Pueden aprender también mediante ESADI (opcional)
  - *End-Station Address Distribution Information*
  - Un RBridge puede anunciar MACs de hosts a otros R Bridges
  - Se transporta en tramas TRILL



# Multidestination

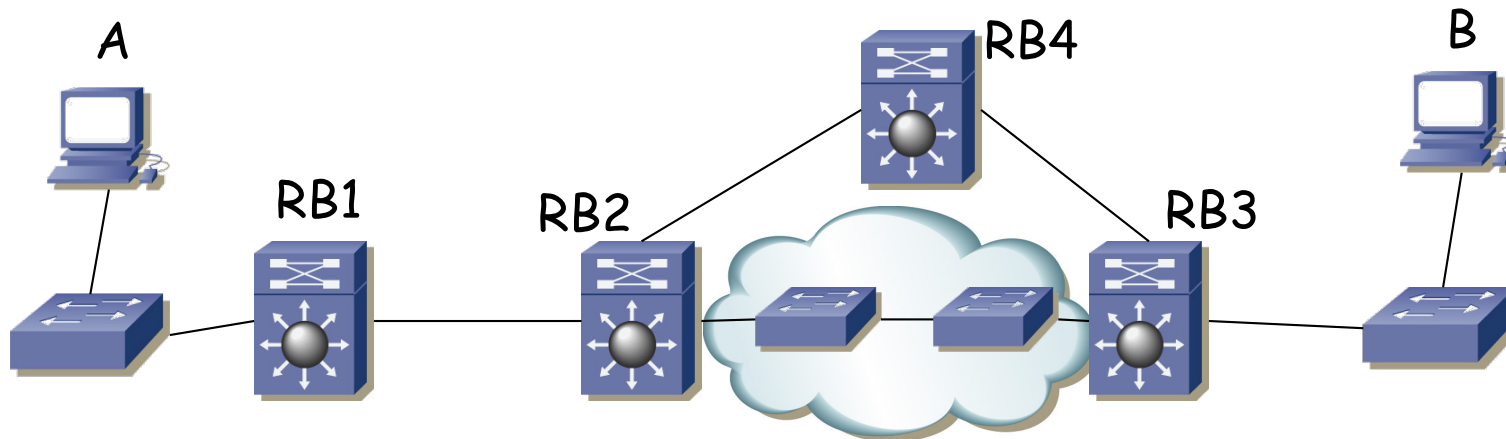
- Casos (BUM)
  - Tramas Broadcast
  - Tramas unicast para las que no se conoce dónde está el destino (Unknown unicast)
  - Tramas Multicast
- Los RBridges construyen árboles de distribución para las tramas multicast (bidireccionales)
- Sería suficiente con un árbol pero calcula múltiples, lo cual le permite multipath también para el multicast
- Lo hace con el mismo IS-IS (no hace falta otro protocolo)
- Cada árbol incluye todos los RBridges del campus y las VLANs
- Puede hacer *pruning*
- Se marcan las tramas con un bit en la cabecera de TRILL
- El Nickname del egress RBridge especifica el árbol
- Los nodos hacen una comprobación de RPF



# Más sobre TRILL

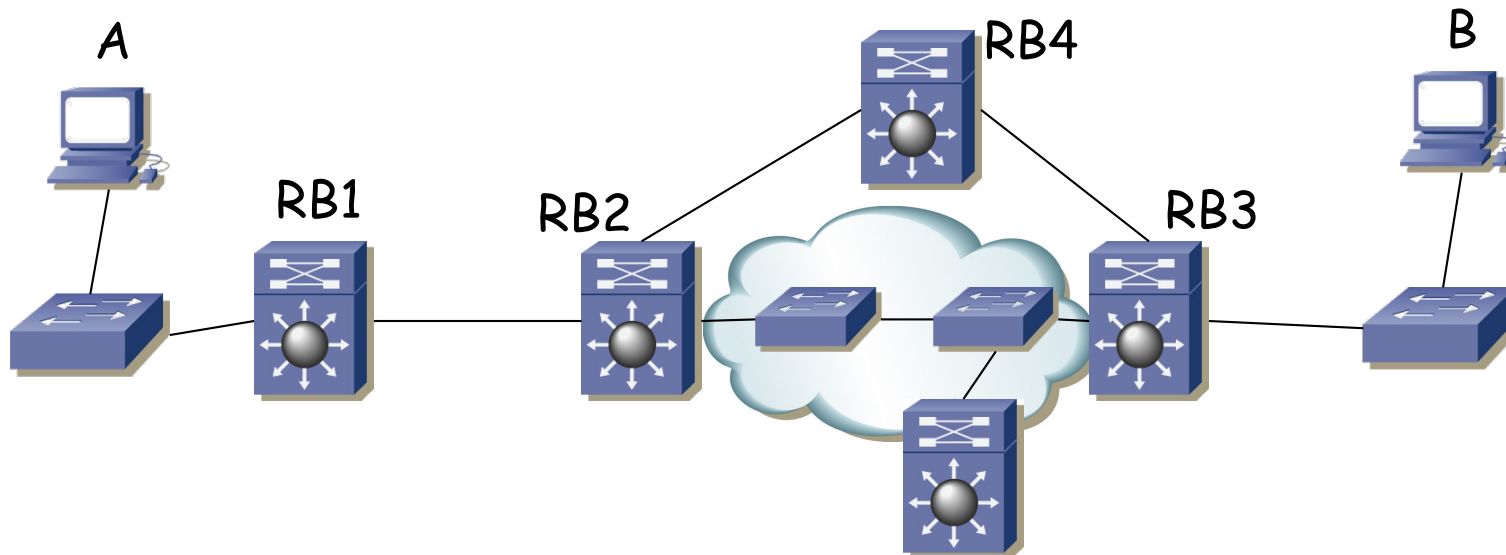
# TRILL y puentes

- Entre dos RBridges puede haber un enlace directo o una LAN con puentes



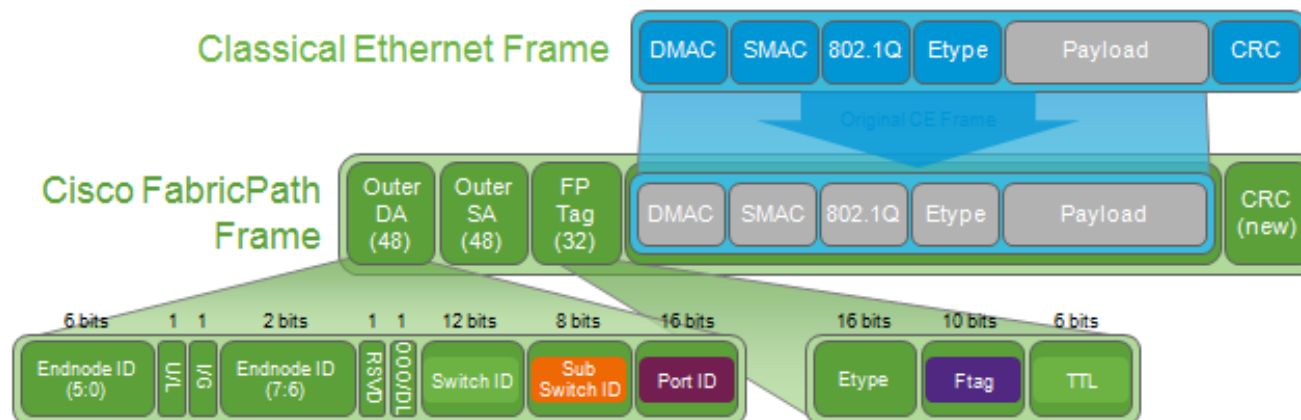
# TRILL y puentes

- Entre dos RBridges puede haber un enlace directo o una LAN con puentes
- Puede haber varios RBridges en una LAN con puentes



# FabricPath y TRILL

- FabricPath es propietario de Cisco
- El plano de control es como en TRILL, es decir, IS-IS sobre L2
- El plano de datos es similar por emplear encapsulación MAC in MAC
- Las direcciones MAC son asignadas localmente, jerárquicamente
  - SwitchID es el identificador único del switch (manual o automático)
  - SubSwitchID para vPC+
  - PortID puede usarse para indicar el puerto en que está el host
  - EndnodeID se puede emplear para distinguir al host origen/destino
  - OOO/DL indica si se puede emplear balanceo por paquete
  - FTag (*Forwarding Tag*) indica una topología lógica que debe emplear
  - Ethertype 0x9003
- Emplea el SwitchID y el FTAG para las decisiones de reenvío



# Más sobre TRILL