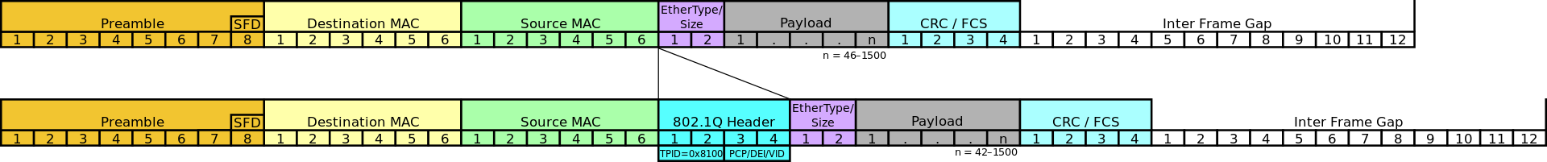
VLAN

= virtuelles LAN in der Norm **802.1Q (IEEE)**

Es fügt eine VLAN-ID in den Ethernet-Header ein

Es gibt ein „Start of frame“(SOF) am Anfang und ein „End of frame“(EOF) am ende

CRC ist eine Validation byte = 32bit

Im EtherType steht IPv4/IPv6

VLAN-IDs gehen von 1 bis 212 (4096)

Die ID ist eindeutig und identifiziert die VLAN

**Port-based VLAN**

Man stellt das auf den Port ein

**Egress-List**

Auf dem Switch gibt es dann eine **Egress-List**

Sagt auf welchem Interface sie den Switch wieder verlassen dürfen

z.B.:

Zwei IDs 100 und 200

|  |  |  |
| --- | --- | --- |
| VLAN-ID | Untagged | Tagged |
| 100 | Gig0/1, Gig0/3 | Gig0/24 |
| 200 | Gig0/2 | Gig0/24 |

**Tagged:**

Die haben einen VLAN-Header

Schickt es mit VLAN-Informationen

**Untagged:**

Die haben keinen VLAN-Header

Schickt es ohne VLAN-Informationen

**Trunk Ports** stehen in der Tagged Liste

Ports an den Clients stecken werden **Acces Points** genannt

802.1x

Router:

Der Router wird mit den **Untagged Ports** verbunden

Es gibt zwei Möglichkeiten:

1. Ein Router der Vlans kann (Cisco)
2. Ein Switch der auch routen und der auch Vlans kann (MikroTik)

MikroTik:

Das gerät hat einen virtuellen Switch, der eine Verbindung gleich mit einem Virtuellen Router verbunden wird 🡪 **Virtuelles Interface**

Auf dem Router virtuelle Vlan-Interfaces machen 🡪 Alles was zu diesem Interface geroutet wird bekommt einen Vlan Stempel z.B.: 100   
Denen kann man jetzt IP-Adressen gegeben

/interface/vlan/…

/interface/vlan ≠ /interface/bridge/vlan

Das eine ist Switching das andere Routing

Trunk 🡪 wo man mehrere VLANs zusammengibt

Native VLAN:

Trunk-Ports werden zwischen Switches verwendet, um die Übertragung von markiertem Verkehr zu unterstützen. Konkret fügt ein 802.1Q-Trunk-Port ein 4-Byte-Tag in den Header des Ethernet-Rahmens ein, um das VLAN zu identifizieren, zu dem der Rahmen gehört.