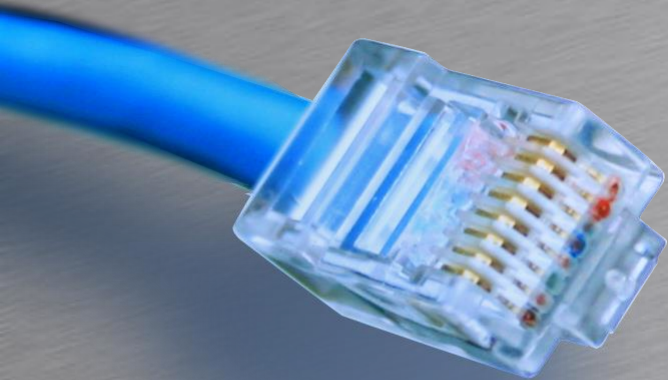


CoS



- en del af **mercantec**⁺

Class of Service

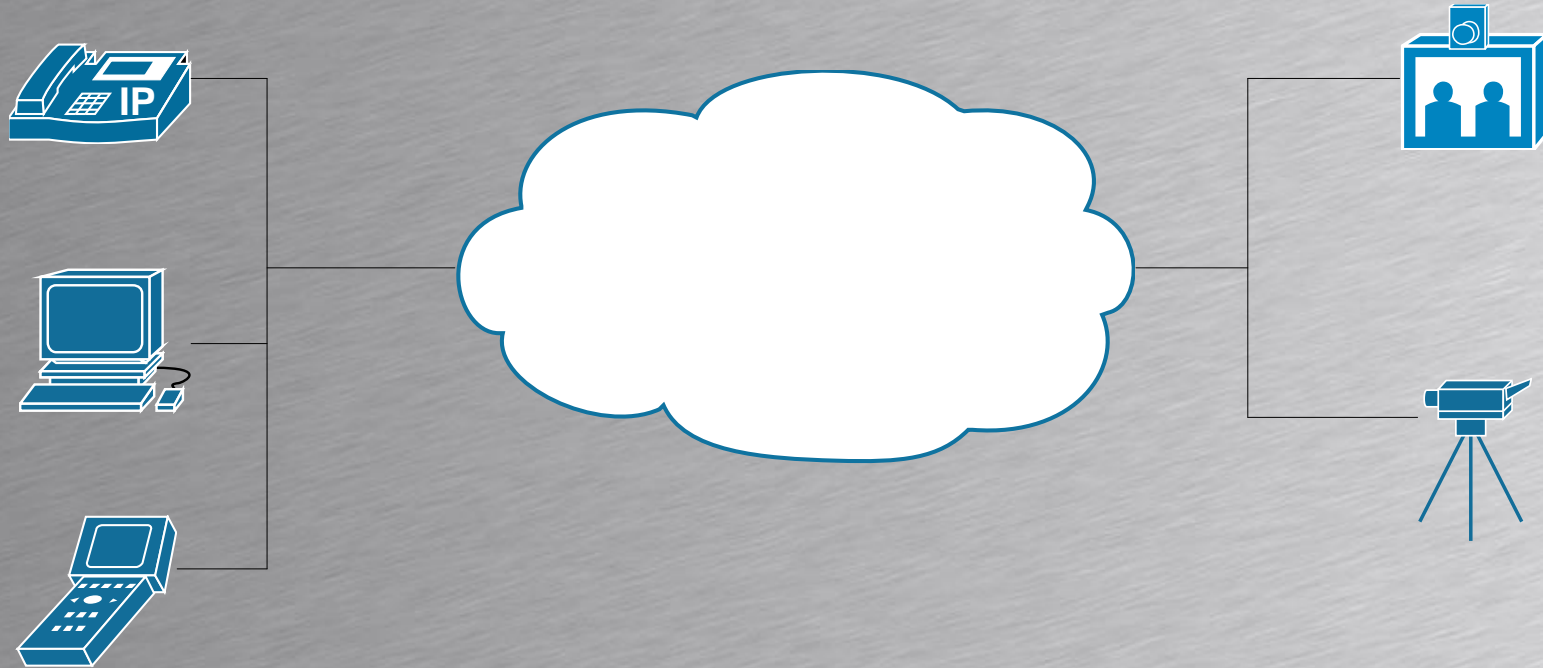
Rasmus Elmholt V1.0



CoS



- Converged networks





CoS

- Converged network ser god ud på papiret
 - Flere netværk bliver samlet i et bærenet
 - Maksimal return of investment
 - Men fordelene forsvinder hurtigt ved uproductive netværk
 - Hvis Mission Critical trafik ikke kommer frem pga. surf på facebook.



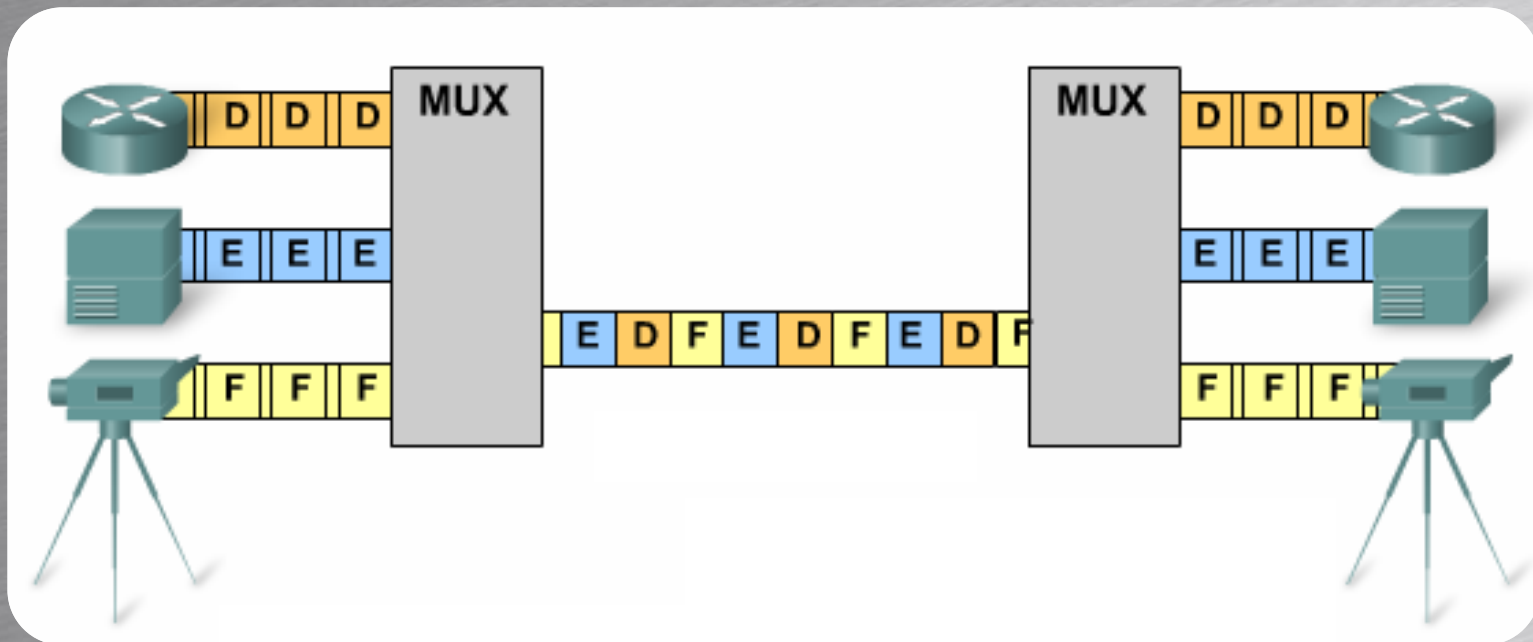
CoS

- Vi kommer fra en tid med kredsløbskoblede teknologier
 - Lavt delay
 - Fixed delay/jitter
 - Lavt pakketab
 - Dedikerede resourcer
 - Call Admission Control



CoS

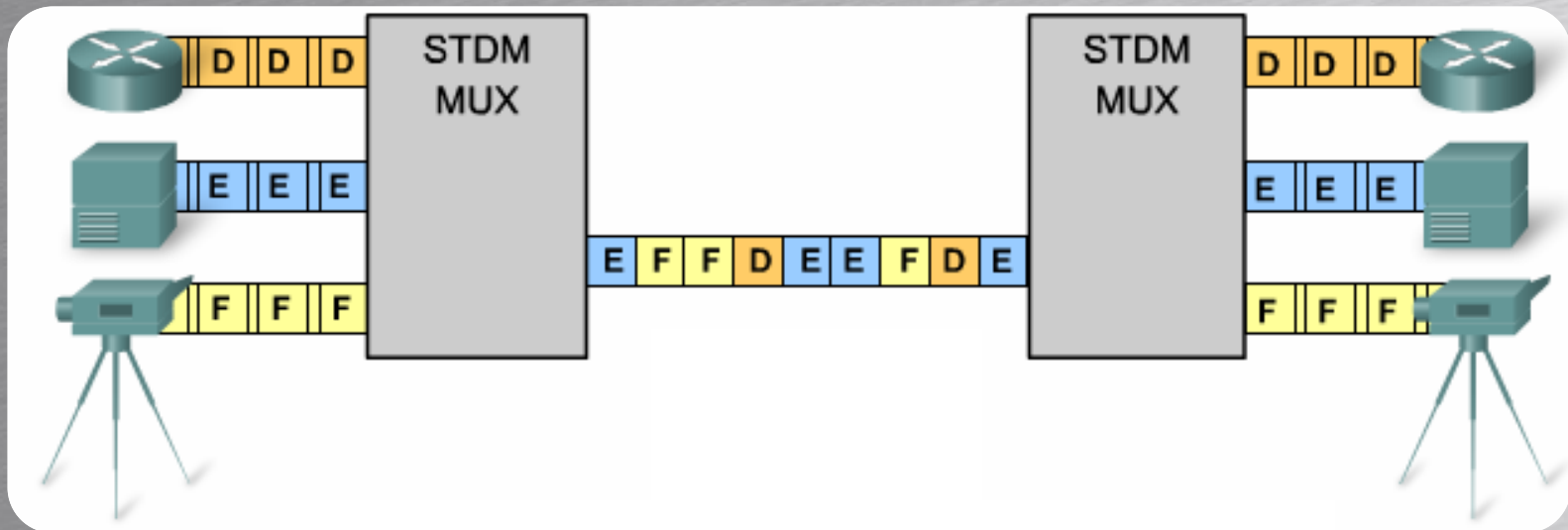
- Multiplexing
 - Time Division Multiplexing





CoS

- Statistical Multiplexing
 - Statistical Time Division Multiplexing





CoS

- Ethernet bruger STDM MUX
 - Udnytter båndbredden bedre
 - Bruger buffer til at gemme pakker
 - Uendelig buffer løser ikke vores problem, men tilfører bare delay
 - Har ikke samme jitterløse egenskaber som vi kender fra kredsløbskoblede netværk



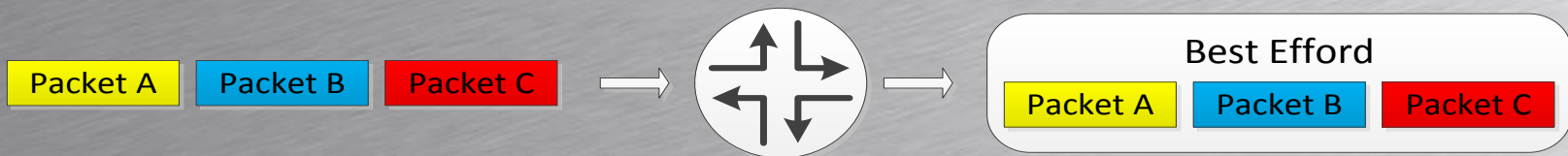
CoS

- Hvad kan man gøre for at undgå uproductive netværk?
 - Mere båndbrede
 - Bliver billigere og billigere
 - Skalerer kun til et vist punkt
 - Når vi kommer over 80% forbrug bliver CoS relevant



CoS

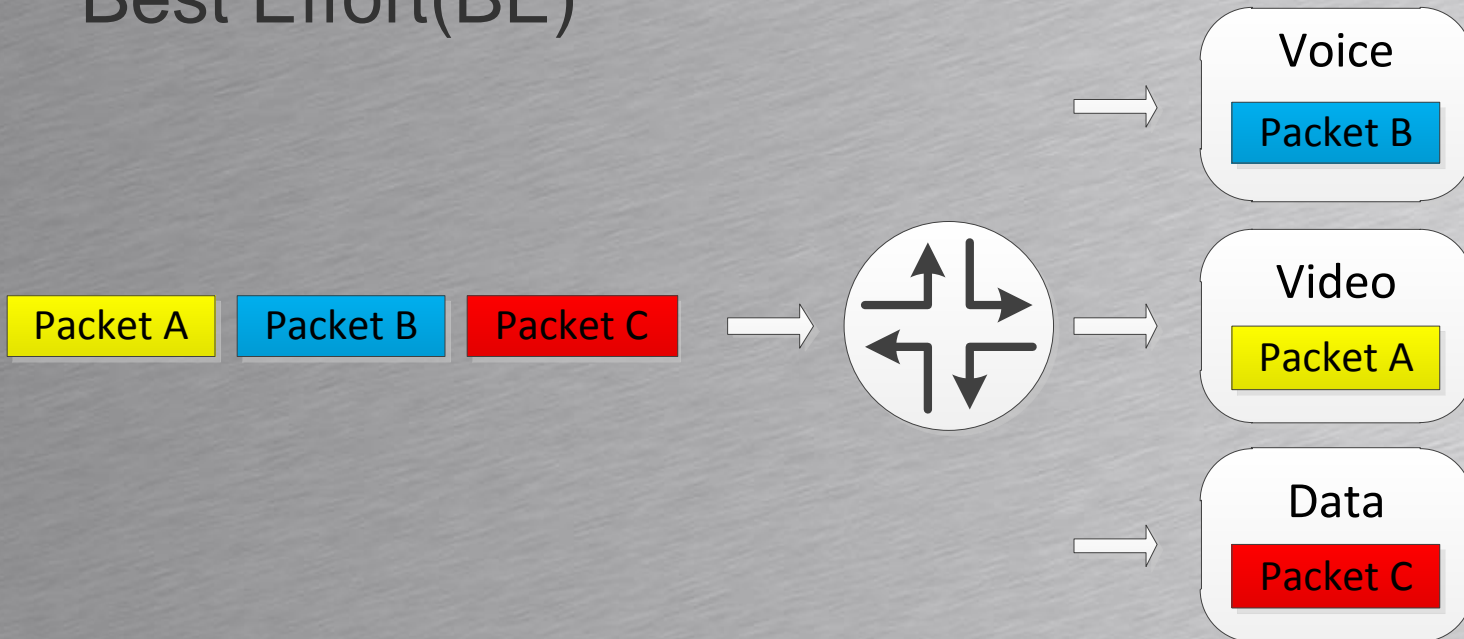
- Som Default bliver alle pakker håndteret ens
 - Alle pakker kommer i de samme input og output køer – FIFO
 - Dette kaldes Best Efford.





CoS

- På Juniper udstyr kan man aktivere CoS
 - Class of Service & Quality of Service er ca. det samme...
 - CoS bruges hvor der er behov for mere end Best Effort(BE)





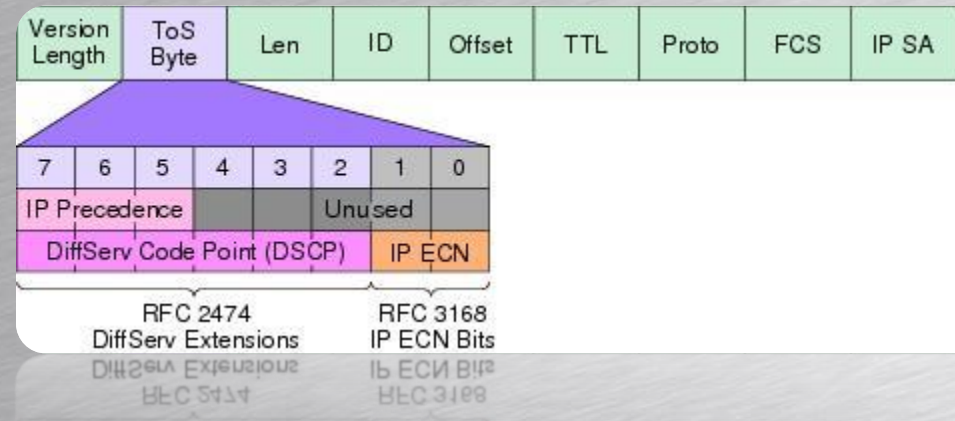
CoS Toolbox

- Class of Service
 - Markering af pakker så efterfølgende routere kan nøjes med at kigge på markeringen
 - Kan håndtere trafik typer forskelligt
 - Bandwidth
 - Delay
 - Delay Variation/Jitter
 - Loss
 - Loss pattern
 - Eller en kombination af flere af ovenstående



CoS Toolbox

- Classification
 - Classification kan ske ud fra 2 metoder.
- Behavioral Aggregate – BA
 - DSCP
 - IP Precedence
 - MPLS EXP
 - 802.1p
- Multifield Classification
 - Matcher trafik ud fra Firewall regler





CoS Toolbox

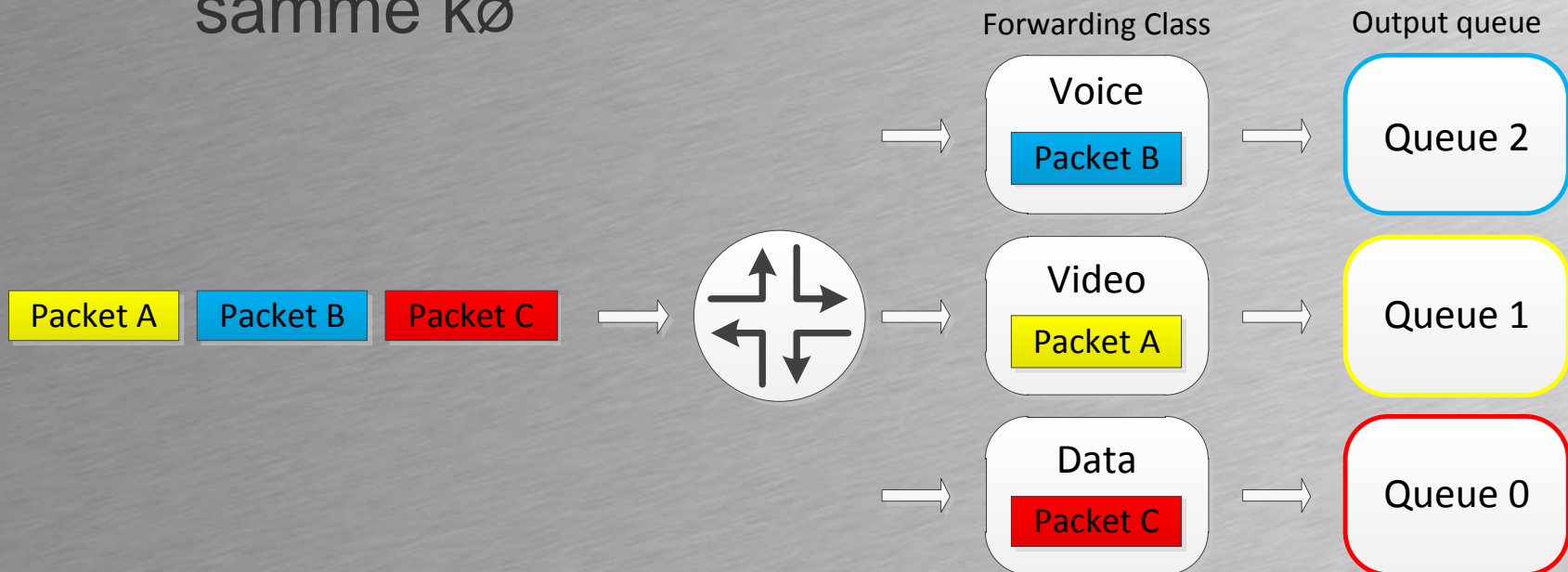
- Forwarding Classes
 - Forwarding Classes er et koncept Juniper bruger for at identificere trafik der skal have samme behandling – Behavioral Aggregates(BA)
 - Trafik bliver smidt i en Forwarding Class under Classification processen



CoS Toolbox

- Forwarding Classes

- Forwarding Classes bliver smidt i bestemte køer på egress interfacet
- Forwarding Classes er ikke det samme som en kø, da flere Forwarding Classes kan ende i samme kø

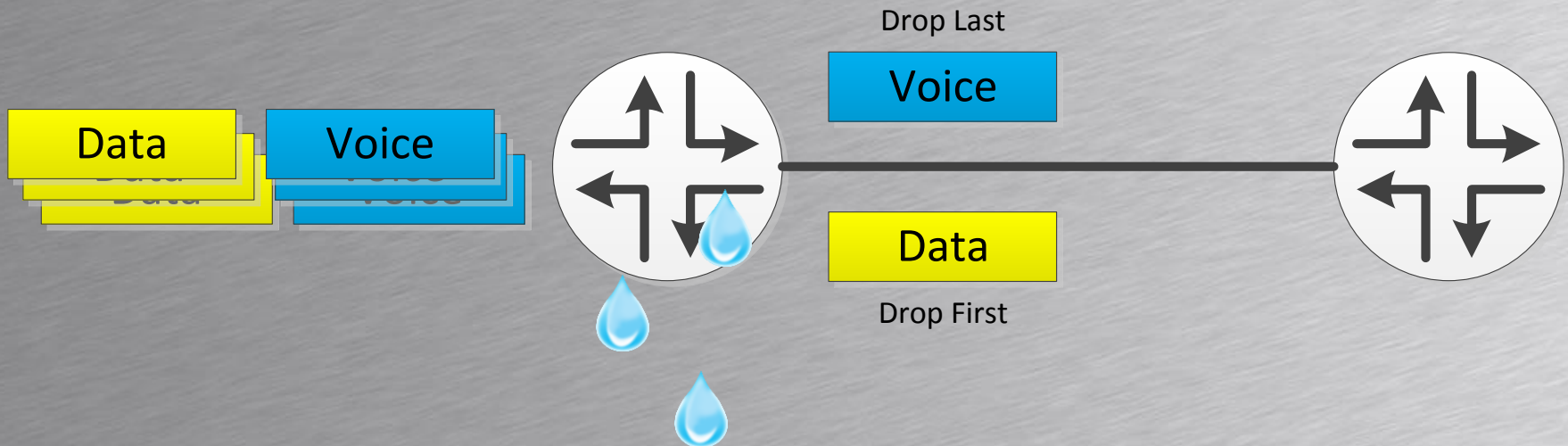




CoS Toolbox



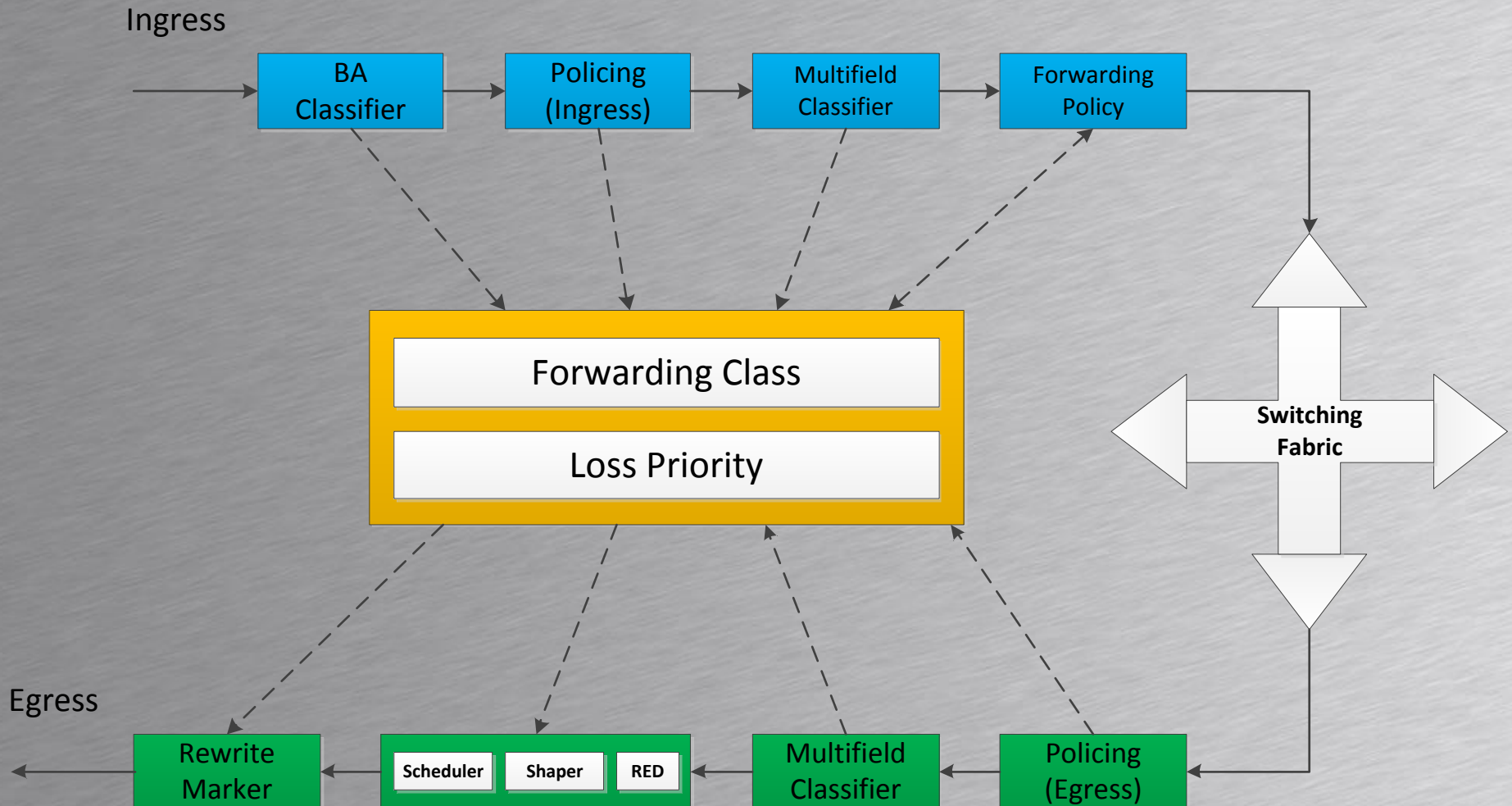
- Packet Loss Priority
 - Man kan tilføje en PLP til en pakke
 - PLP'en kan bruges i RED profiler til at definere drop probability





CoS Toolbox

- CoS Processering





CoS Toolbox

- CoS Processering
 - Multifield classifiers matcher ud fra firewall politikker
 - Bruges til at sætte Forwarding Class
 - Bruges til at sætte Loss Priority
 - Sættes i *then* klausulen
 - Forwarding politikker kan bruges for at ændre routing for bestemte CoS pakker – *Ikke en del af dette kursus*

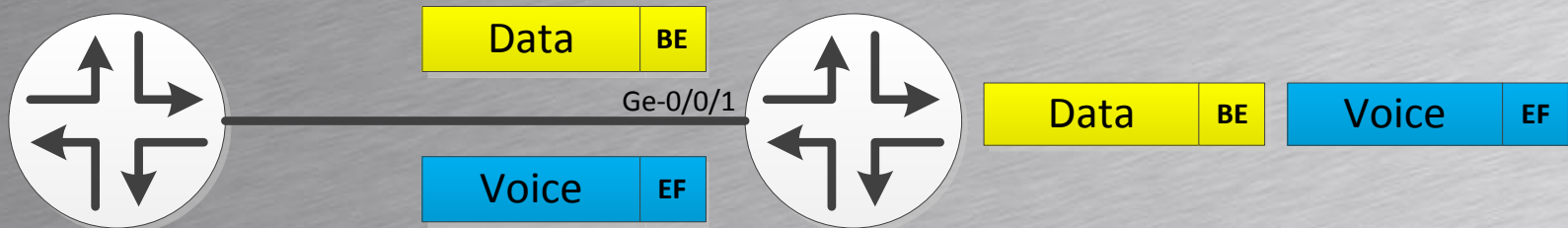


CoS Classification

- Behavior Aggregate Classifier – Eksempel
– Sættes på Ingress interface

```
[edit class-of-service]
root@SRX240# show
interfaces {
  ge-0/0/1 {
    unit 0 {
      classifiers {
        inet-precedence default;
      }
    }
  }
}
```

Vi stoler nu på trafik og sætter det i default Queue for trafik typen



```
show class-of-service classifier type dscp
```



CoS Classification

- BA Classifier - Default

DSCP	Forwarding Class	PLP
EF	expedited-forwarding	Low
AF11	assured-forwarding	Low
AF12	assured-forwarding	High
AF13	assured-forwarding	High
CS6	network-control	Low
CS7	network-control	Low
Other	best-effort	Low



CoS Classification

- Multifield Classifier - Eksempel

```
[edit firewall family inet filter APPLY-COS]
root@SRX240# show
term ADMIN-NET {
    from {
        source-address {
            192.168.146.0/24;
        }
    }
    then {
        forwarding-class expedited-forwarding;
        accept;
    }
}
term ALL-OTHER {
    then accept;
}
```

```
[edit interfaces ge-0/0/1 unit 0]
root@SRX240# show
family inet {
    filter {
        input APPLY-COS;
    }
    address 192.168.146.100/24;
}
```

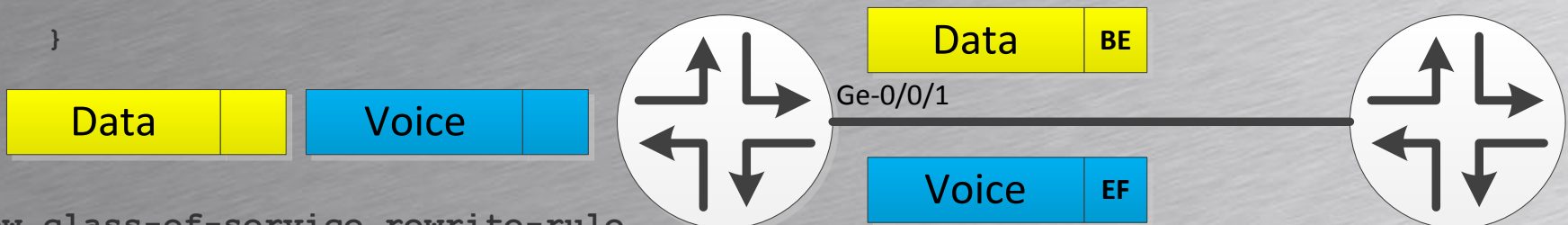


CoS Rewrite

- Behavior Aggregate Rewrite – Eksempel
– Sættes på Egress interface

```
[edit class-of-service]
root@SRX240# show
interfaces {
  ge-0/0/1 {
    unit 0 {
      rewrite-rules {
        inet-precedence default;
      }
    }
  }
}
```

Map from FC	PLP	DSCP/.1p/IPP
Expedited-forwarding	Low	EF
Expedited-forwarding	High	EF
Assured-Forwarding	Low	AF11
Assured-Forwarding	High	AF12
Best-Effort	Low	BE
Best-Effort	High	BE
Network-Control	Low	CS6
Network-Control	High	CS7



Show class-of-service rewrite-rule



CoS Queues

- Forwarding Classes bliver kædet sammen med en kø
- Default sammenkædning er ens på de fleste enheder med Junos OS:

Forwarding Class	Navn	Kø Nummer
0	Best-Efford	0
1	Expedited-Forwarding	1
2	Assured-Forwarding	2
3	Network-Control	3

```
[edit class-of-service]
```

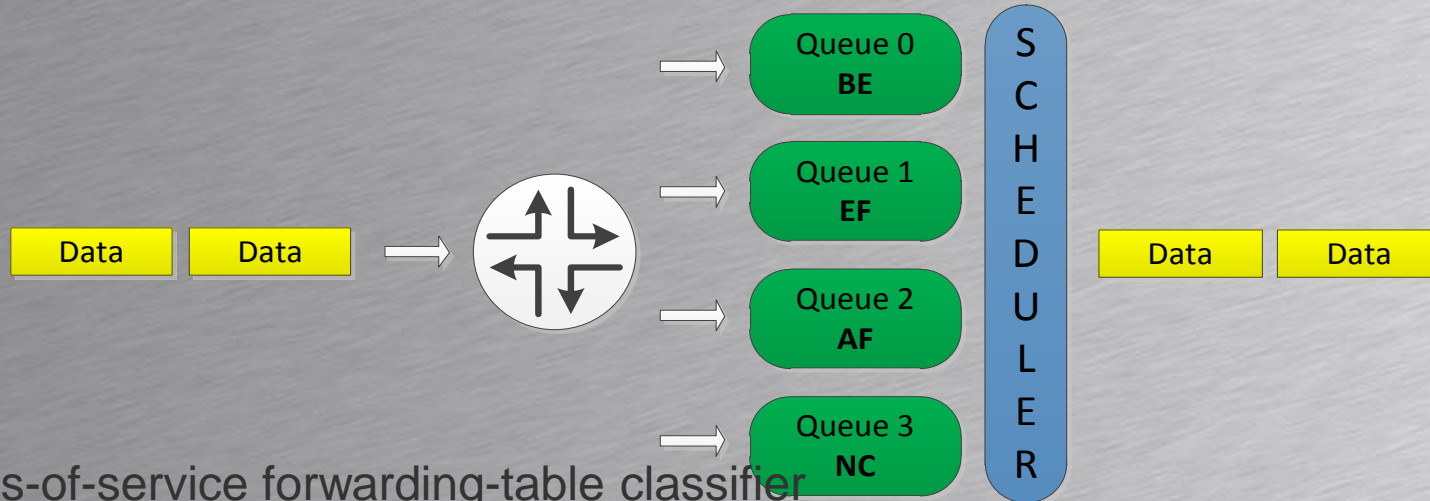
```
root@SRX240# set forwarding-classes queue 2 teacher-traffic
```

```
Show class-of-service forwarding-class
```



CoS Queues

- Når trafikken rammer et Egress interface bliver hver Forwarding Class smidt i en kø.
 - Antal mulige køer varierer pga. HW.
 - Som default sendes Routing Protokoller bl.a. andet i Queue 3(NC)
 - Alt andet trafik smides i Queue 0(BE)



show class-of-service forwarding-table classifier



CoS Schedulers

- For at bestemme hvordan trafik bliver afsendt skal der konfigureres nogle schedulers
 - Priority
 - Transmission rate
 - Buffer size
 - RED
- Junos OS Software Default
 - Queue 0 får 95% buffer og båndbredde
 - Queue 3 får 5% buffer og båndbredde



CoS Schedulers

- Priority angiver rækkefølgen hvori køer bliver tømt.
- Den tømmer altid den med højeste prioritet først og arbejder sig nedefter.
 - Strict-high - PQ
 - High
 - Medium-high
 - Medium-low
 - Low

Køerne skal have positiv credit for denne prioritering virker.



CoS Schedulers

- Schedulers - Eksempel

```
[edit class-of-service schedulers]
```

```
root@SRX240# show
```

```
scheduler-best-efford {
```

```
    transmit-rate percent 50;
```

```
    buffer-size percent 50;
```

```
    priority low;
```

```
}
```

```
scheduler-assured-forwarding {
```

```
    transmit-rate percent 10 exact;
```

```
    buffer-size percent 10;
```

```
    priority high;
```

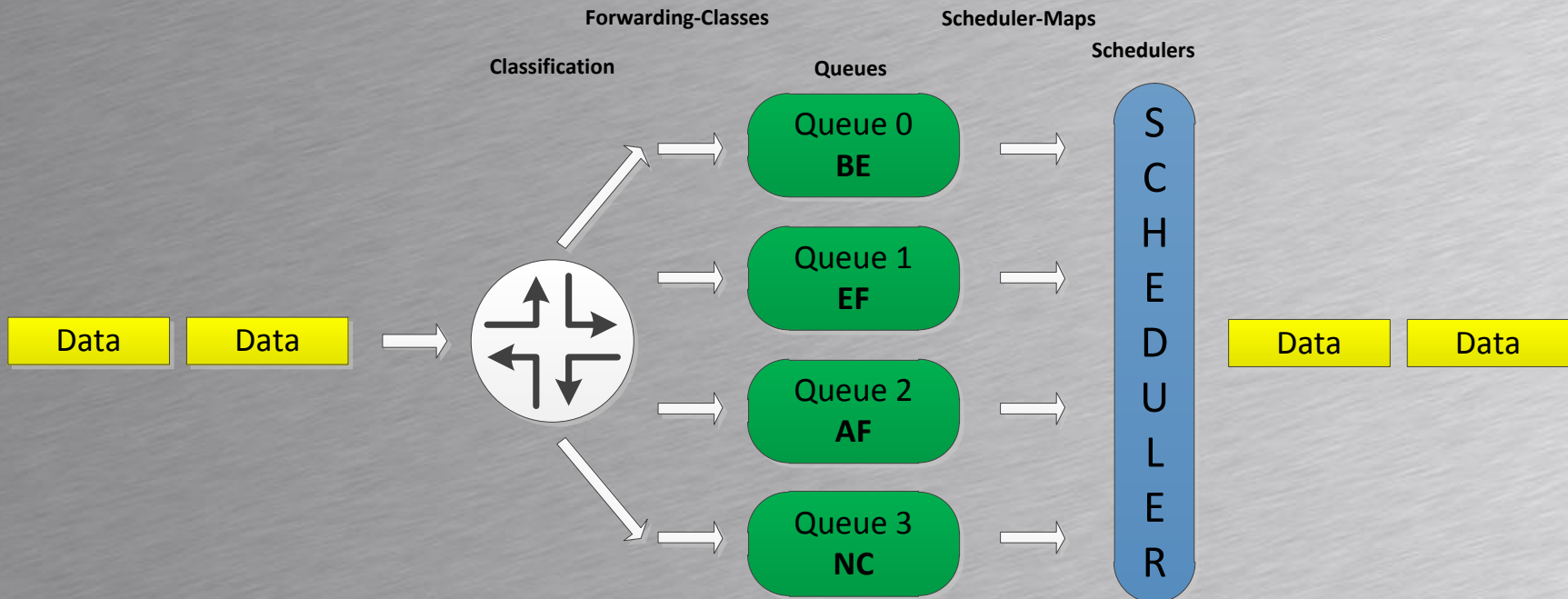
```
}
```

← Køen må ikke overskride 10%



CoS Scheduler-maps

- Scheduler-maps kæder forwarding-classes sammen med Schedulers og deres køer



show class-of-service scheduler-map



CoS Scheduler-maps

- Scheduler-maps - Eksempel

```
[edit class-of-service scheduler-maps]
root@SRX240# show
GE-0/0/0-OUT {
    forwarding-class best-effort scheduler scheduler-best-efford;
    forwarding-class assured-forwarding scheduler scheduler-assured-forwarding;
}
```



CoS Scheduler-maps

- Scheduler-maps - default

```
root@SRX07# run show class-of-service scheduler-map
Scheduler map: <default>, Index: 2
```

```
Scheduler: <default-be>, Forwarding class: best-effort, Index: 21
  Transmit rate: 95 percent, Rate Limit: none, Buffer size: 95 percent,
  Buffer Limit: none, Priority: low
  Excess Priority: low
```

Drop profiles:

Loss priority	Protocol	Index	Name
Low	any	1	<default-drop-profile>
Medium low	any	1	<default-drop-profile>
Medium high	any	1	<default-drop-profile>
High	any	1	<default-drop-profile>

```
Scheduler: <default-nc>, Forwarding class: network-control, Index: 23
  Transmit rate: 5 percent, Rate Limit: none, Buffer size: 5 percent,
  Buffer Limit: none, Priority: low
  Excess Priority: low
```

Drop profiles:

Loss priority	Protocol	Index	Name
Low	any	1	<default-drop-profile>
Medium low	any	1	<default-drop-profile>
Medium high	any	1	<default-drop-profile>
High	any	1	<default-drop-profile>



CoS Scheduler-maps

- Før det hele virker skal scheduler-maps sættes på et interface
 - Tilføjes på Egress interface
 - Man kan bruge wildcards

```
[edit class-of-service interfaces]
root@SRX240# show
fe-* {
    scheduler-map FE-OUT;
}
ge-0/0/0 {
    scheduler-map GE-0/0/0-OUT;
}
```



CoS Shaping

- Hvis man vil lave en båndbredde begrænsning kan shaping bruges i udgående retning af et interface
- Her begrænses til 2Mbit/s

```
[edit]
root@SRX07# show | compare rollback 1

[edit interfaces ge-0/0/0]
+   per-unit-scheduler;

[edit class-of-service interfaces ge-0/0/0
unit 10]
+       shaping-rate 2m;
```



CoS Policing

- Hvis man vil forhindre kunderne i at sende for meget EF trafik kan vi police
- Policing kan bruges på 2 måder:
 - Traffic rate limiting
 - CoS traffic conforming
 - Alt trafik der overskrider den definerede båndbredde bliver nedmarkeret.

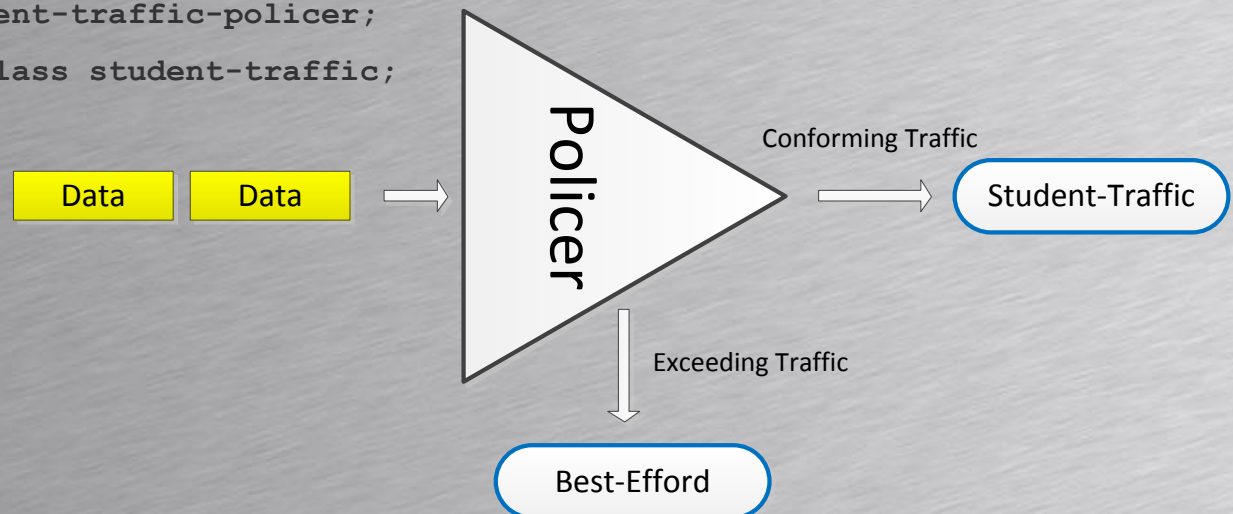


CoS Policing

- CoS Policing - Eksempel

```
family inet {  
  filter if-cos-filter {  
    term student-traffic {  
      from {  
        source-address {  
          10.0.0.0/24;  
        }  
      }  
      then {  
        policer student-traffic-policer;  
        forwarding-class student-traffic;  
        accept;  
      }  
    }  
    term all-other {  
      then accept;  
    }  
  }  
}
```

```
[edit firewall]  
root@SRX240# show  
policer student-traffic-policer {  
  if-exceeding {  
    bandwidth-limit 10m;  
    burst-size-limit 3k;  
  }  
  then forwarding-class best-effort;  
}
```





CoS Toolbox

- CoS Processering - Summary

