

# Механизмы обеспечения качества оказываемых услуг



# Что такое QoS?

Quality of Service (QoS) — качество обслуживания.

Цели, которые преследует QoS:

- бесшовное качество передачи сервиса между клиентами;
- гарантированный сервис для выбранных IP/Ethernet пакетов;
- поддержка различных типов приложений и специфичных бизнес требований.

Традиционная передача пакетов:

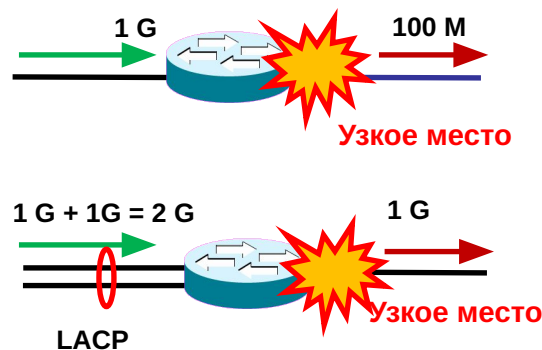
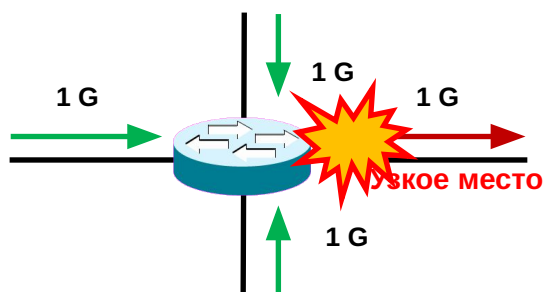
- Политика использования максимально возможного канала передачи без гарантии доставки и какой-либо классификации передаваемых пакетов.

Новые приложения, используемые в сетях передачи данных и требующие обеспечение QoS: Video-on-Demand (VOD), VOIP, видеоконференции.

# Проблемы требующие QoS

## Проблемы перегрузки:

- Перегрузка сети является ключевым фактором, снижающий скорость передачи информации
- Увеличивается задержка передачи, появляются потери пакетов. Клиенты, видя потерю пакетов, начинают слать информацию заново, тем самым еще увеличивая поток данных.



## Решения :

- Увеличение пропускной способности сети
- QoS.

# Модели реализации QoS в сети

Можно выделить три модели реализации QoS в сети:

- **Негарантированная доставка данных (Best Effort Service, BE)** – обеспечивает связь между узлами, но не гарантирует надежную доставку данных, время доставки, пропускную способность и приоритет.
- **Интегрированные услуги (Integrated Services, IntServ)** – эта модель RFC 1633, предполагает предварительное резервирование сетевых ресурсов с целью обеспечения предсказуемого поведения сети для приложений, требующих для нормального функционирования гарантированной выделенной полосы пропускания на всем пути следования трафика. Эту модель также часто называют *жестким QoS (hard QoS)* в связи с предъявлением строгих требований к ресурсам сети.
- **Дифференцированное обслуживание (Differentiated Service, DiffServ)** – эта модель RFC 2474, RFC 2475, предполагает разделение трафика на классы на основе требований к качеству обслуживания. Модель DiffServ занимает промежуточное положение между моделью BE и моделью IntServ и сама по себе не предполагает обеспечение гарантий предоставляемых услуг, поэтому ее называют *мягким QoS (soft QoS)*.



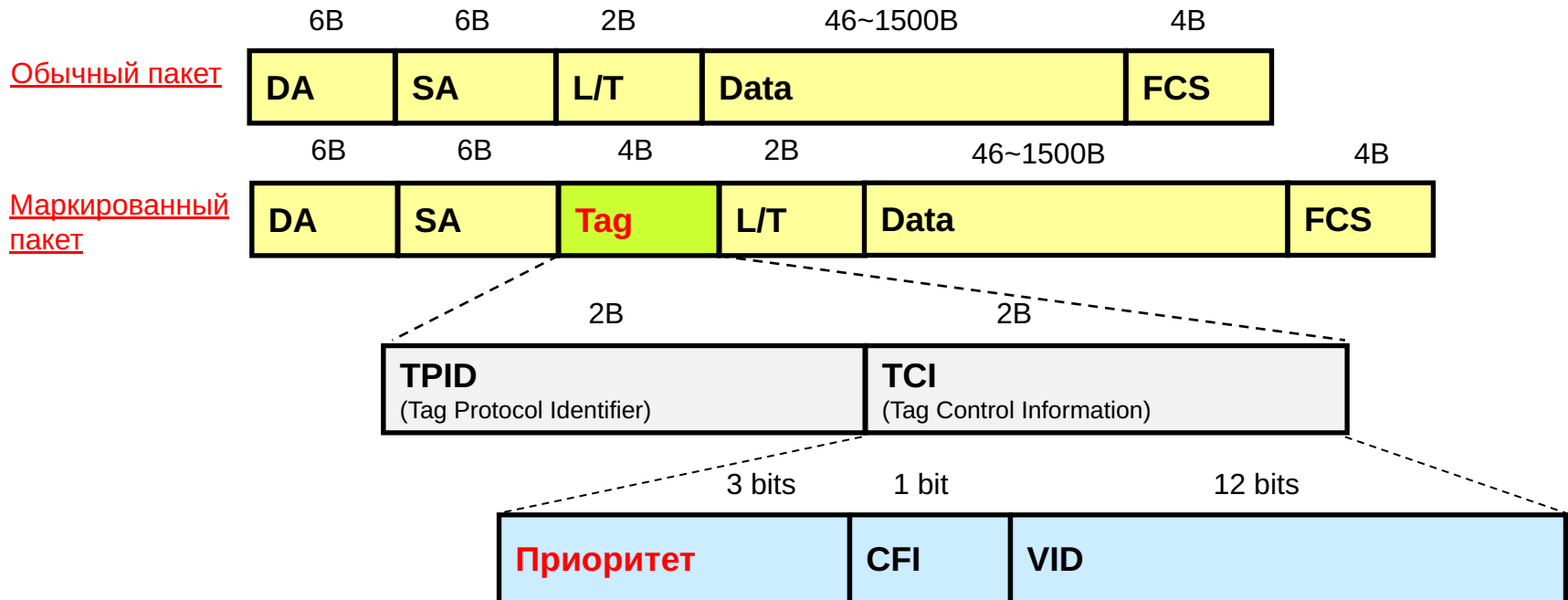
# Качество обслуживания на L2 (IEEE 802.1p)

# Приоритет 802.1p

## Особенности приоритета пакетов на L2:

- Для обеспечения QoS на канальном уровне модели OSI коммутаторы поддерживают стандарт IEEE 802.1p.
- Приоритет пакета на L2 также известен как Класс обслуживания (Class of Service).
- Стандарт IEEE 802.1p позволяет задать до 8 уровней приоритетов (от 0 до 7, или как еще принято обозначать от CS0 до CS7, где 7 – наивысший), определяющих способ обработки кадра, используя 3 бита поля приоритета тега IEEE 802.1Q.
- Поле приоритета 802.1p содержится в двухбайтном поле TCI заголовка 802.1Q.
- Разным типам трафика назначаются разные значения CoS.

# Расположение приоритета 802.1p



- **TPID:** Tag Protocol Identifier, 802.1Q TPID = 0x8100.
- **CFI:** Canonical Format Indicator, всегда 0 для Ethernet.
- **VID:** VLAN ID, 4096 VLAN используется 802.1Q для VLAN идентификации.

# QoS в MAN сетях

## Рекомендации:

- **VoIP** – QoS 5
- **IPTV** – QoS 4
- **Data:**
  - Management – QoS 7
  - Internet – QoS 3
  - Intranet (Local) – QoS 0

## Примечание:

Несмотря на рекомендованную раскраску трафика приоритетом QoS, администратор сети может сам выбрать оптимальный вариант для своей сети.

## Типовое использование:

Приоритет	Типовые приложения
111 (7)	Зарезервировано (управ. трафик)
110 (6)	Зарезервировано (управ. трафик)
101 (5)	Голосовые потоки
100 (4)	Видеоконференции
011 (3)	Телефонная сигнализация (SIP)
010 (2)	Высокоприоритетный трафик
001 (1)	Трафик со средним приоритетом
000 (0)	Обычный трафик



# Приоритет по умолчанию

Используется для того, чтобы добавить тег 802.1p/1q к нетегированному входящему кадру. Приоритет по умолчанию для каждого порта равен 0.

```
DGS-3620-28TC:4#show 802.1p default_priority
```

```
Command: show 802.1p default_priority
```

```
Port      Priority
```

```
-----
```

```
1         0
```

```
2         0
```

```
3         0
```

```
4         0
```

```
...
```

```
26        0
```

Поменять приоритет по умолчанию на портах:

```
config 802.1p default_priority [<portlist> | all] <priority 0-7>
```

# CoS Mapping

**CoS Mapping** – это привязка QoS (IEEE 802.1p) к CoS (очередям обработки). Используется для ассоциации пользовательского приоритета 802.1p входящего кадра с одной из аппаратных очередей приоритетов на коммутаторе.

Приоритет кадра внутри коммутатора определяется тем, к какой очереди он приписан, а не приоритетом 802.1p.

В коммутаторах D-Link используются следующие варианты привязок:

```
DGS-3620# show 802.1p
user_priority
QoS Class of Traffic

Priority-0 -> <Class-2>
Priority-1 -> <Class-0>
Priority-2 -> <Class-1>
Priority-3 -> <Class-3>
Priority-4 -> <Class-4>
Priority-5 -> <Class-5>
Priority-6 -> <Class-6>
Priority-7 -> <Class-7>
```

```
DES-3200-28# show 802.1p
user_priority
COS Class of Traffic

Priority-0 -> <Class-1>
Priority-1 -> <Class-0>
Priority-2 -> <Class-0>
Priority-3 -> <Class-1>
Priority-4 -> <Class-2>
Priority-5 -> <Class-2>
Priority-6 -> <Class-3>
Priority-7 -> <Class-3>
```

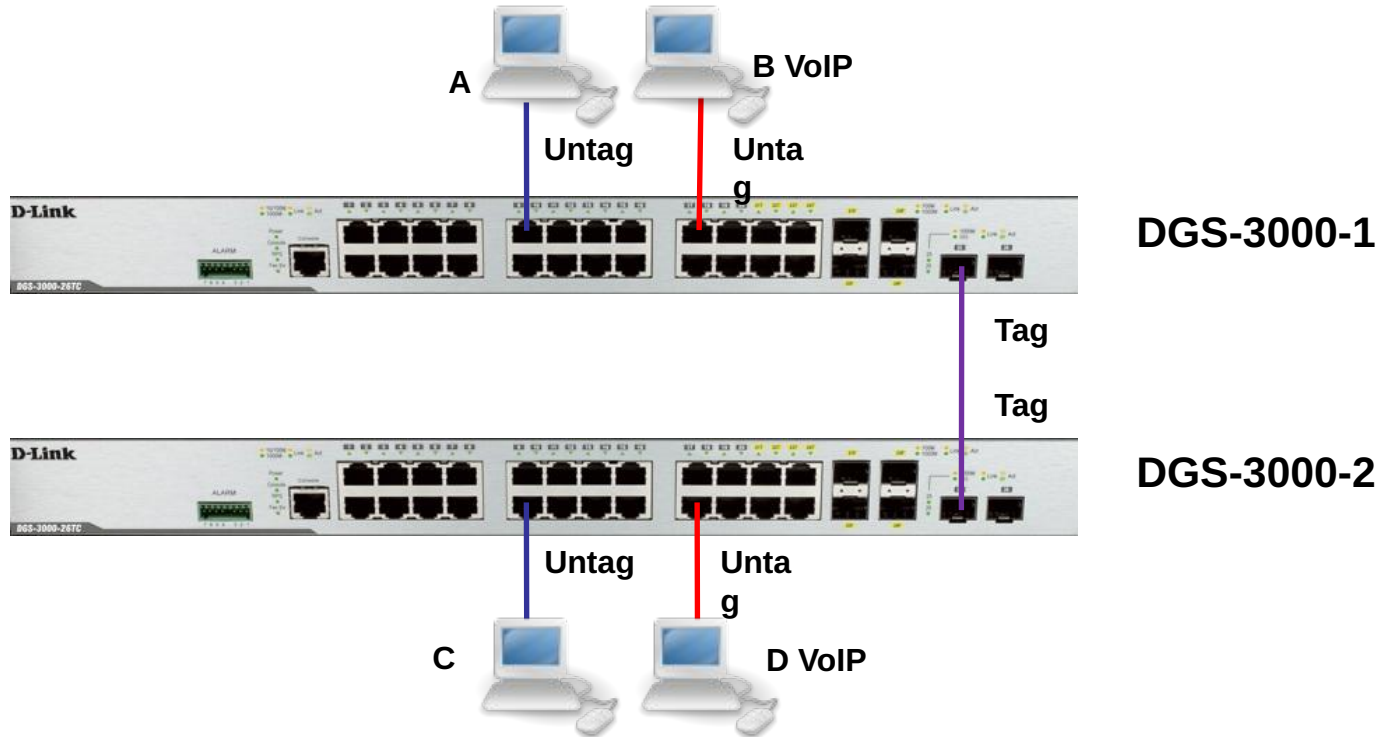
Настроить CoS mapping:

```
config 802.1p user_priority <priority 0-7> <class_id 0-7>
```

# Пример использования L2 QoS

## Задача:

На компьютерах B и D запущены приложения VoIP, и им необходимо более высокое качество обслуживания (QoS) чем другим станциям с обычными приложениями.



# Пример использования L2 QoS

## Конфигурация DGS-3000-1:

1. Перевести порт, соединяющий DGS-3000-1 и DGS-3000-2 из “untag” в “tag” так, чтобы приоритеты смогли быть переданы между коммутаторами.

```
config vlan default delete 25  
config vlan default add tagged 25
```

2. Поменять приоритет по умолчанию порта 17, к которому подключено устройство VoIP, с 0 на 7.

```
config 802.1p default_priority ports 17 7
```

3. Пользовательский приоритет и метод обработки остаются по умолчанию.

## Конфигурация DGS-3000-2:

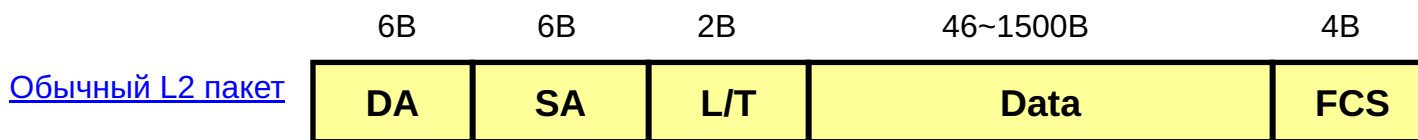
Аналогична DGS-3000-1, за исключением пункта 2, где вместо 17 порта, указываем 18.



# Качество обслуживания на L3 (Type of service)

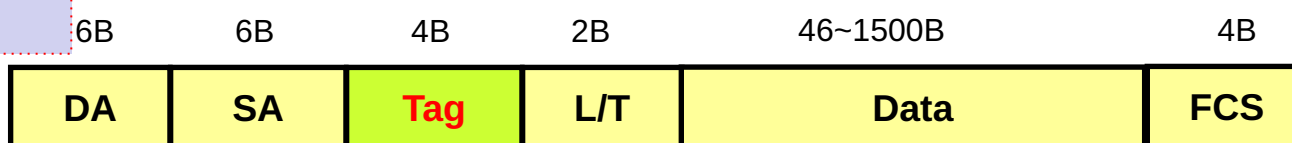
# QoS на L2 и L3

В случае снятия тега теряется информация о приоритете 802.1p. Следовательно, для передачи приоритета в нетегированном трафике, необходимо располагать информацией не в L2 кадре, а в L3 пакете.



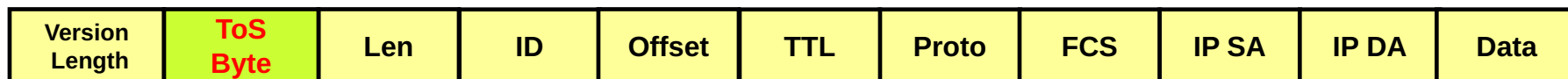
Добавление приоритета

L2 Кадр с 802.1Q



3 бита, используются для QoS

L3 Пакет IPv4

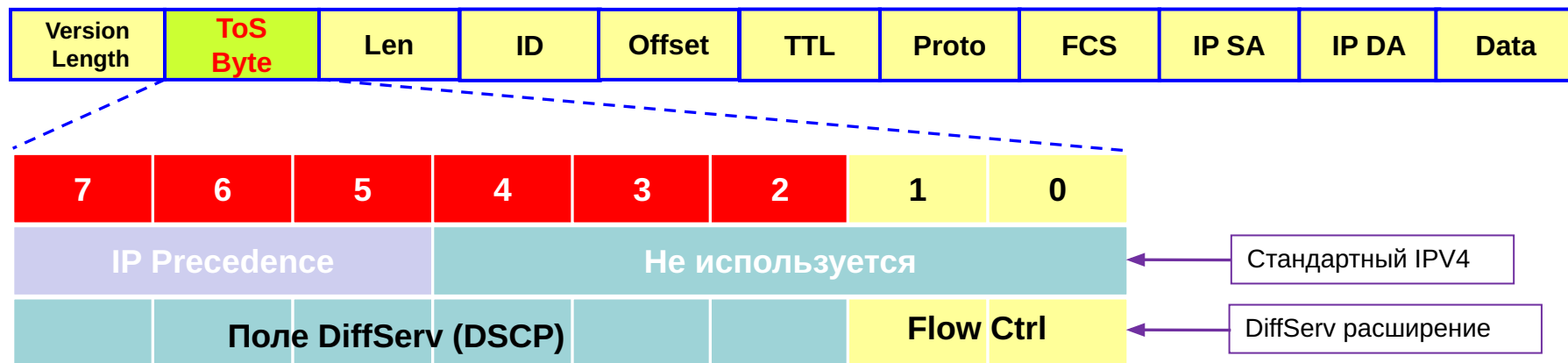


IP Precedence или DSCP (1 байт)

IP Precedence: 3 Most Significant Bits (MSBs) ToS

DSCP: 6 MSBs ToS

# Layer 3: IP ToS байт



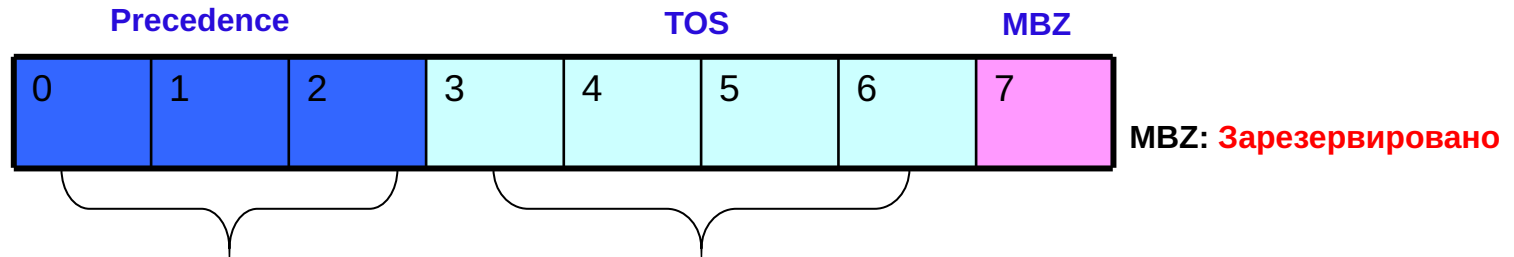
## Поле IP Precedence:

Имеет размерность 3 бита и может принимать значения от 0 до 7. Оно используется для указания относительного приоритета обработки пакета на сетевом уровне. Другие биты не используются.

## DiffServ код (DSCP):

Было стандартизировано IETF с появлением модели DiffServ. Оно занимает 6 старших бит байта ToS и позволяют задать до 64 уровней приоритетов (от 0 до 63). По сути, код DSCP является расширением 3-битового поля IP Precedence и обладает обратной совместимостью с IP-приоритетом.

# ToS - оригинальная версия



Высокий

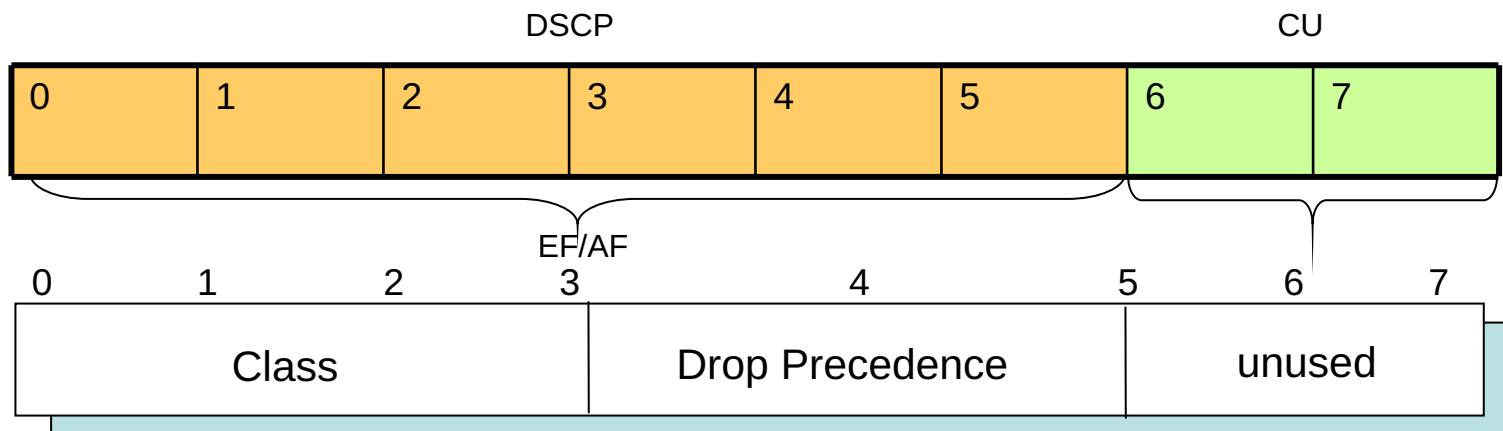
Приоритет

Низкий

IP Precedence	IP Precedence	TOS: Указывает на тип сервиса	Типичные приложения
111 (7)	Network control	1000	Минимальные задержки
110 (6)	Internetwork control	0100	Максимальная полоса
101 (5)	Critical	0010	Максимальная надёжность
100 (4)	Flash Override	0001	Минимальная стоимость
011 (3)	Flash	0000	Обычное обслуживание
010 (2)	Immediate		
001 (1)	Priority		
000 (0)	Routine		



# ToS тип 2 – DiffServ Code Point (DSCP)



Ex: **001010**

- 001** □ определяет класс
- 01** □ возможность отброса
- 0** □ всегда 0

	Class 1	Class 2	Class 3	Class 4
Низкие потери	001010 (DSCP 10) AF 11	010010 (DSCP 18) AF 21	011010 (DSCP 26) AF 31	100010 (DSCP 34) AF 41
Средние потери	001100 (DSCP 12) AF 12	010100 (DSCP 20) AF 22	011100 (DSCP 28) AF 32	100100 (DSCP 36) AF 42
Высокие потери	001110 (DSCP 14) AF 13	010110 (DSCP 22) AF 23	011110 (DSCP 30) AF 33	100110 (DSCP 38) AF 43

# Привязка CoS и IP Precedence к DSCP

## IP Precedence and DSCP Values

3-bit IP Precedence	6 MSb <sup>1</sup> of ToS						6-bit DSCP		3-bit IP Precedence	6 MSb <sup>1</sup> of ToS						6-bit DSCP		3-bit IP Precedence	6 MSb <sup>1</sup> of ToS						6-bit DSCP									
	8	7	6	5	4	3				8	7	6	5	4	3				8	7	6	5	4	3		8	7	6	5	4	3			
0	0	0	0	0	0	0	0		4	1	0	0	0	0	0	32		2	0	1	0	0	0	0	16		6	1	1	0	0	0	0	48
	0	0	0	0	0	1	1			1	0	0	0	0	1	33			0	1	0	0	0	1	17			1	1	0	0	0	1	49
	0	0	0	0	1	0	2			1	0	0	0	1	0	34			0	1	0	0	1	0	18			1	1	0	0	1	0	50
	0	0	0	0	1	1	3			1	0	0	0	1	1	35			0	1	0	0	1	1	19			1	1	0	0	1	1	51
	0	0	0	1	0	0	4			1	0	0	1	0	0	36			0	1	0	1	0	0	20			1	1	0	1	0	0	52
	0	0	0	1	0	1	5			1	0	0	1	0	1	37			0	1	0	1	0	1	21			1	1	0	1	0	1	53
	0	0	0	1	1	0	6			1	0	0	1	1	0	38			0	1	0	1	1	0	22			1	1	0	1	1	0	54
	0	0	0	1	1	1	7			1	0	0	1	1	1	39			0	1	0	1	1	1	23			1	1	0	1	1	1	55
1	0	0	1	0	0	0	8		5	1	0	1	0	0	0	40		3	0	1	1	0	0	0	24		7	1	1	1	0	0	0	56
	0	0	1	0	0	1	9			1	0	1	0	0	1	41			0	1	1	0	0	1	25			1	1	1	0	0	1	57
	0	0	1	0	1	0	10			1	0	1	0	1	0	42			0	1	1	0	1	0	26			1	1	1	0	1	0	58
	0	0	1	0	1	1	11			1	0	1	0	1	1	43			0	1	1	0	1	1	27			1	1	1	0	1	1	59
	0	0	1	1	0	0	12			1	0	1	1	0	0	44			0	1	1	1	0	0	28			1	1	1	1	0	0	60
	0	0	1	1	0	1	13			1	0	1	1	0	1	45			0	1	1	1	0	1	29			1	1	1	1	0	1	61
	0	0	1	1	1	0	14			1	0	1	1	1	0	46			0	1	1	1	1	0	30			1	1	1	1	1	0	62
	0	0	1	1	1	1	15			1	0	1	1	1	1	47			0	1	1	1	1	1	31			1	1	1	1	1	1	63

## Привязка CoS и IP Precedence к DSCP

CoS/IP Precedence	0	1	2	3	4	5	6	7
DSCP	0	8	16	24	32	40	48	56

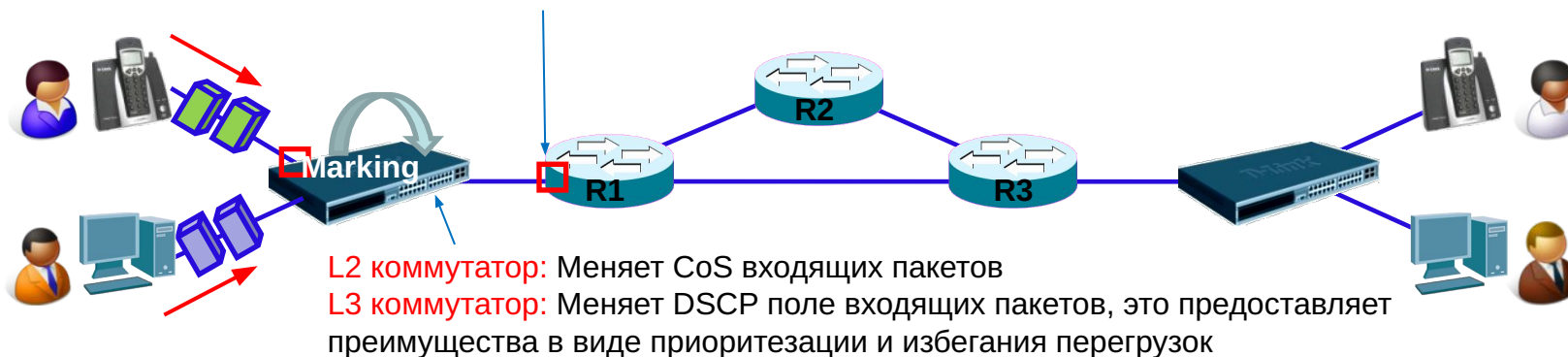
# Маркировка пакетов приоритетом

- Маркировка даёт возможность изменить QoS биты (DSCP, CoS, или IP Precedence) входящих кадров.
- Позволяет изменить обработку коммутатором маркированных кадров
- В первую очередь следует определить в каком месте сети классифицировать и маркировать пакеты. Главное правило гласит:

**Маркируйте пакеты как можно ближе к источнику**

**Пример:** Маркировка голосового трафика значением 40 DSCP, после этого коммутатор будет обрабатывать трафик с высоким приоритетом.

L2 кадр с CoS полем, R1 может привязать значение CoS к Precedence или DSCP полю.  
L3 пакет с DSCP/IP Precedence полем будет перемаркирован.



# DSCP маркировка

## DSCP Mapping:

Используется для ассоциации приоритета DSCP с приоритетом кадра на коммутаторе. Приоритет кадра внутри коммутатора определяется тем, к какой очереди он приписан.

Настроить DSCP Mapping на коммутаторах D-Link:

```
config dscp map [<portlist> | all] [ dscp_priority <dscp_list> to <priority 0-7> |  
dscp_dscp <dscp_list> to <dscp 0-63>]
```

По умолчанию в коммутаторах D-Link используются следующая схема привязки:

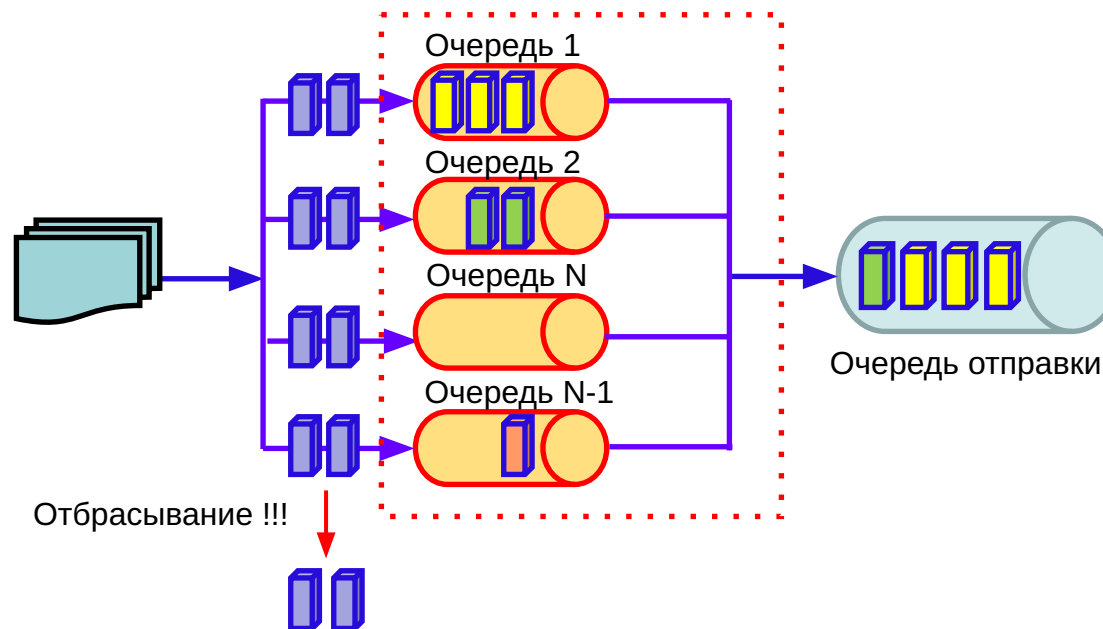
```
config dscp map all dscp_priority 0-7 to 0  
config dscp map all dscp_priority 8-15 to 1  
config dscp map all dscp_priority 16-23 to 2  
config dscp map all dscp_priority 24-31 to 3  
config dscp map all dscp_priority 32-39 to 4  
config dscp map all dscp_priority 40-47 to 5  
config dscp map all dscp_priority 48-55 to 6  
config dscp map all dscp_priority 56-63 to 7
```



# Решение проблемы перегрузки

# Механизмы управления перегрузками

- Для управления перегрузками и их предотвращения используются множественные исходящие очереди
- Предотвращение перегрузки – это свойство очереди, поэтому каждая очередь может иметь свои собственные настройки
- Механизм предотвращения перегрузок в своём составе имеет несколько алгоритмов управления очередью.



# Алгоритмы управления очередями

**FIFO** (First Input First Output) – коммутатор отправляет пакеты в том порядке, в котором они приходят.

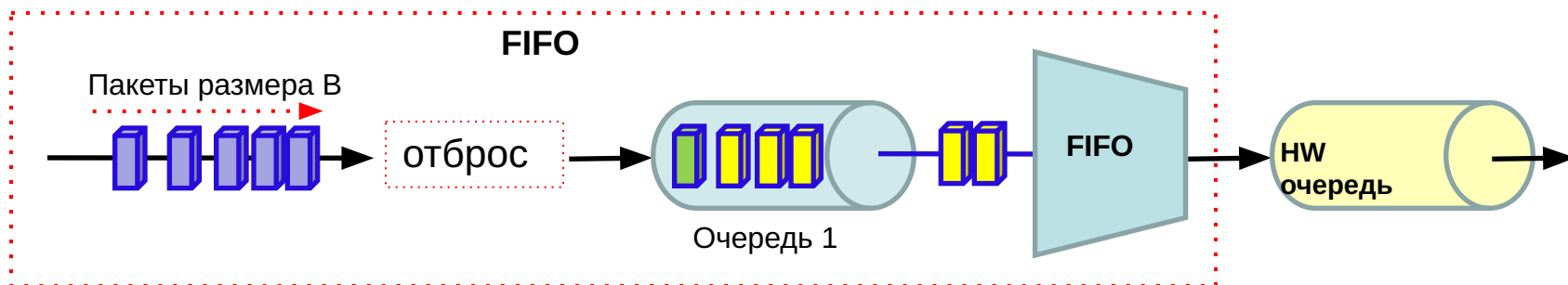
**Priority queuing** (Приоритетная очередь) – предполагают передачу трафика строго в соответствии с приоритетом выходных очередей.

**Weighted round robin (WRR)** (Взвешенный круговой режим) – обработка начинается с очереди с наивысшим приоритетом, потом переходит к более низкому и так далее, а в конце возвращается к наивысшему приоритету, и всё повторяется опять.

**Custom queuing** (Очередь, настраиваемая пользователем) – обработка настраивается пользователем, для каждого приоритета отдельно (с возможностью комбинаций приоритетной очереди и взвешенного кругового режима).

# FIFO и приоритетная очередь

**FIFO** не использует классификацию, все поступающие пакеты будут попадать в один класс. Коммутатор отправляет пакеты в том порядке, в котором они приходят.



**Priority queuing** – если настроен строгий приоритет в одной из очередей, коммутатор обрабатывает пакеты из этой очереди до тех пор пока в ней есть пакеты.

Обработка других очередей приостанавливается.

Очередь с приоритетами полезна для передачи голосового трафика.



# Обработка приоритетов Strict Priority

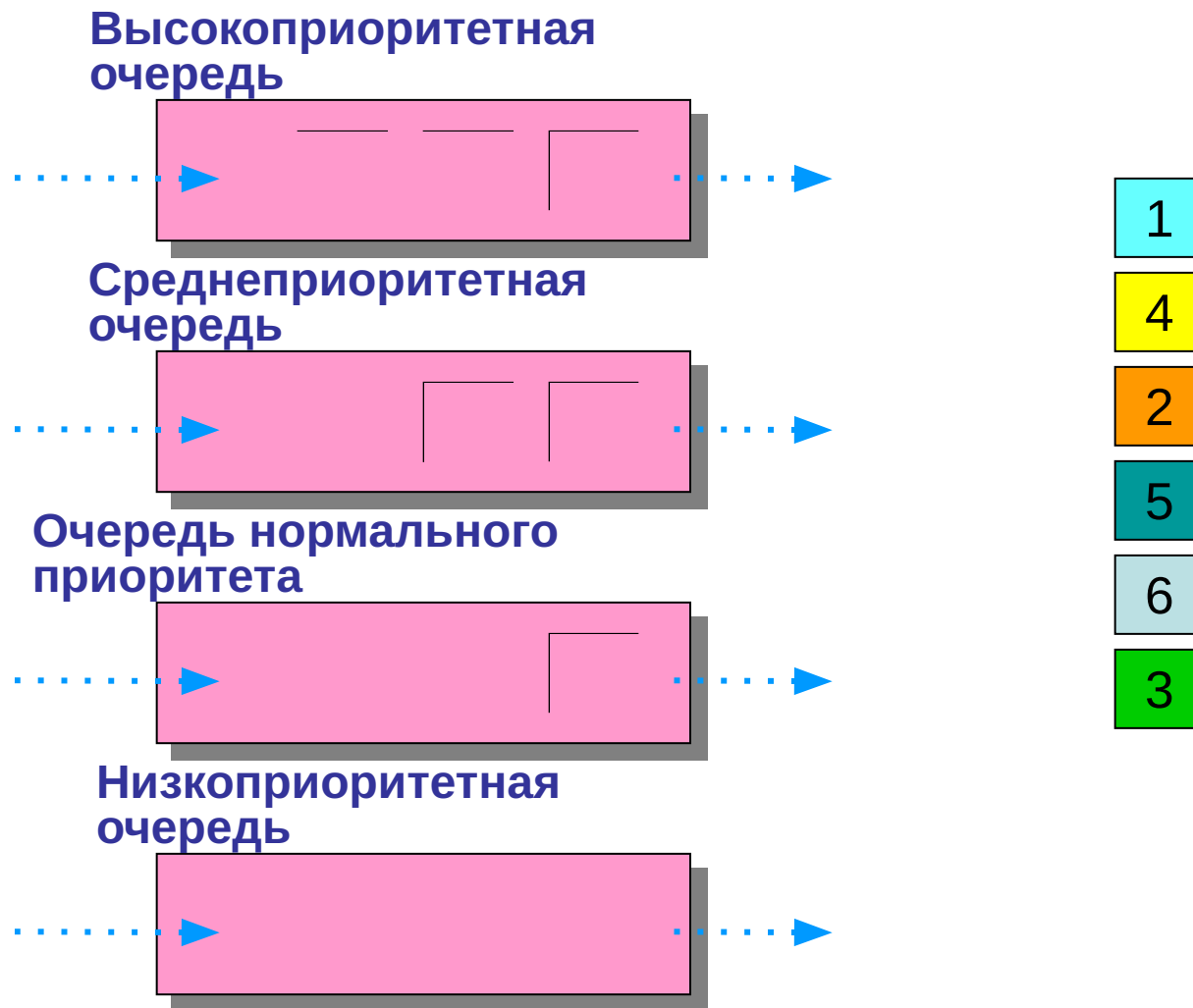
Обработка приоритетов производится в соответствии с одним из методов, строгий или по весу.

При строгом методе, кадры в очередях с высоким приоритетом обрабатываются первыми. Только тогда, когда эти очереди пусты, могут быть обработаны кадры с более низким приоритетом. Кадры с высоким приоритетом всегда получают предпочтение независимо от количества кадров в других очередях в буфере и времени, прошедшего с момента передачи последнего кадра с низким приоритетом. По умолчанию коммутатор настроен как раз на этот режим.

```
config scheduling_mechanism ports <portlist> {strict |wrr}
```

Этот тип приоритезации может привести к “застою” пакетов в других, неприоритетных очередях

# Строгий приоритет Strict Priority



# Взвешенный круговой режим (WRR)

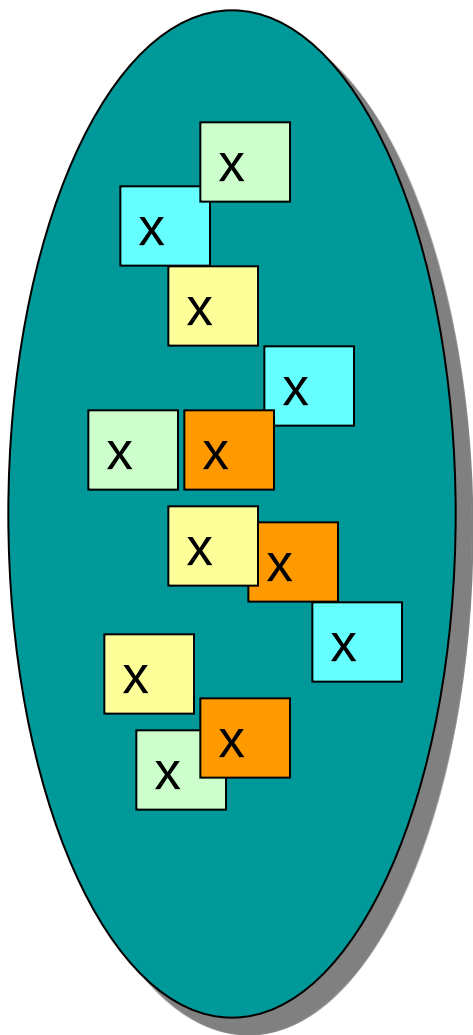
**WRR** – обработка начинается с очереди с наивысшим приоритетом, потом переходит к более низкому и т.д., а в конце возвращается к наивысшему приоритету, и всё повторяется по кругу.

Такой режим исключает главный недостаток строгого режима. Очередь с минимальным приоритетом уже не страдают от переполнения, поскольку всем очередям предоставляется часть пропускной способности для передачи. Это достигается заданием максимального числа кадров, которые можно передать из данной очереди приоритетов, перед тем как перейти к следующей.

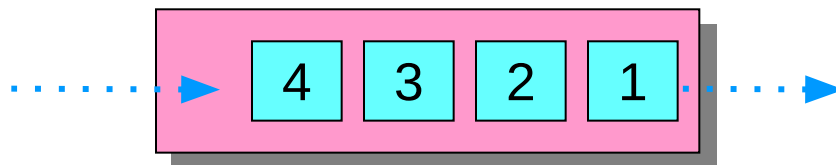
```
config scheduling <class_id 0-6> {weight <value 1-127>}
```

Для использования этой схемы, параметры `weight` не должны иметь значение 0. Он задаёт весовой коэффициент для количества кадров в определённой очереди, которое может быть передано за один раз (цикл). Это обеспечивает поддержку CoS, и также даёт возможность передавать кадры из всех очередей. Это значение можно изменять в диапазоне от 1 до 127 кадров для каждой очереди приоритетов.

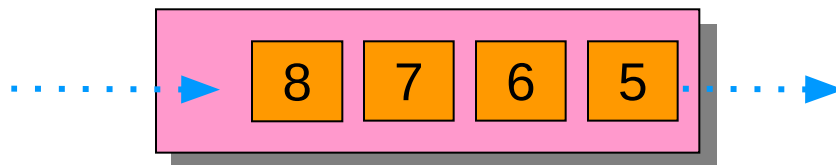
# Взвешенный круговой режим (WRR)



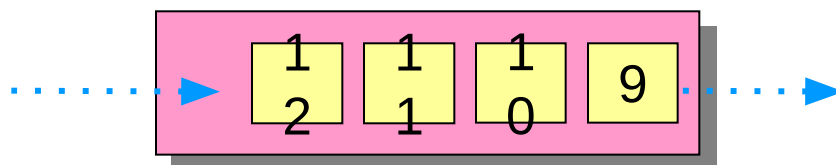
Высокоприоритетная очередь (40)



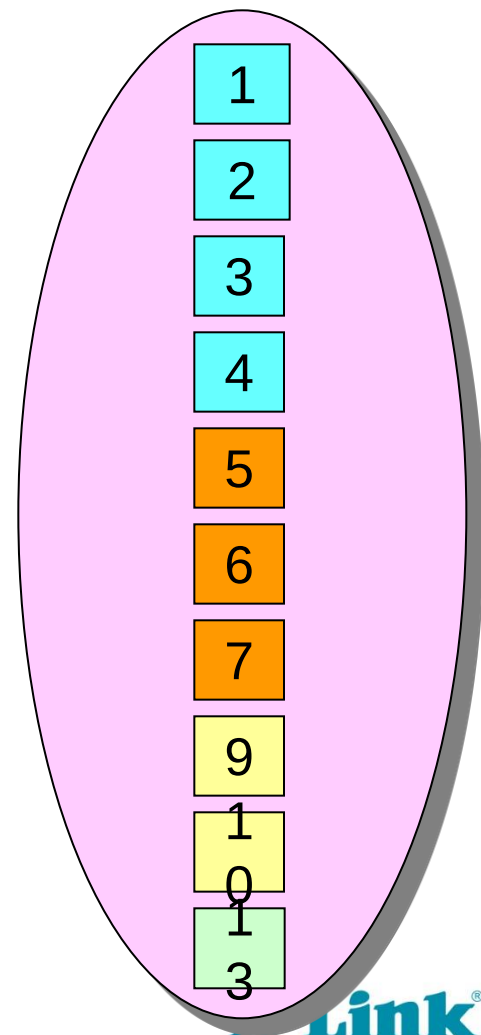
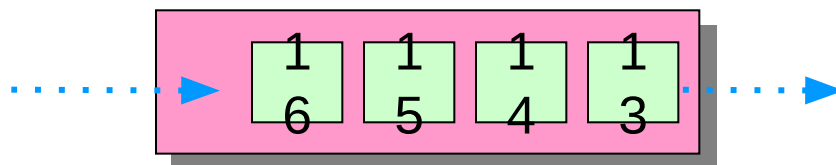
Среднеприоритетная очередь (30)



Очередь нормального приоритета (20)



Низкоприоритетная очередь (10)



# Механизм предотвращения перегрузок

Механизм предотвращения перегрузок (Congestion avoidance) – это процесс выборочного отбрасывания пакетов с целью избежания перегрузок в сети в случае достижения выходными очередями своей максимальной длины (в пакетах).

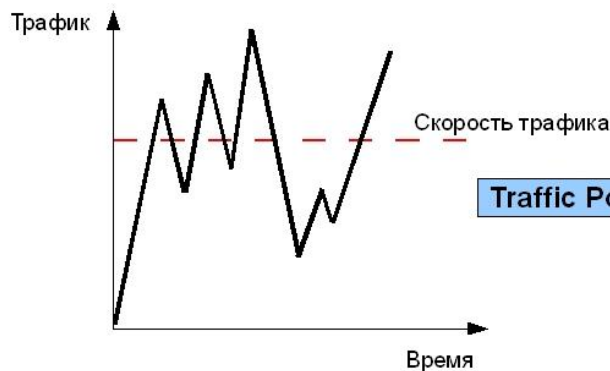
## Алгоритмы предотвращения перегрузок:

- Алгоритм «отбрасывания хвоста» (Tail-Drop);
- Алгоритм произвольного раннего обнаружения (Random Early Detection, RED);
- Простой алгоритм произвольного раннего обнаружения (Simple Random Early Detection, SRED);
- Взвешенный алгоритм произвольного раннего обнаружения (Weighted Random Early Detection, WRED).

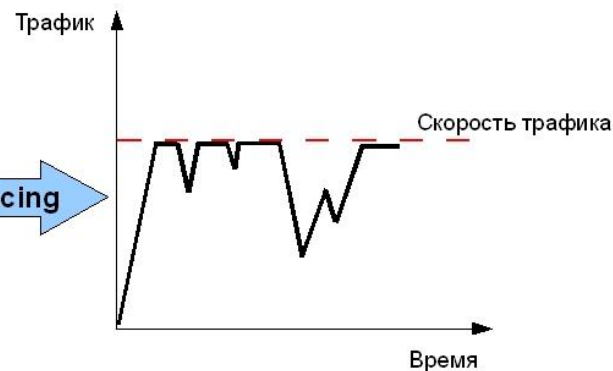
# Контроль полосы пропускания

Механизмы **Traffic Policing** (ограничение трафика) и **Traffic Shaping** (выравнивание трафика) – позволяют регулировать интенсивность трафика с целью обеспечения функций качества обслуживания.

Без Traffic Policing

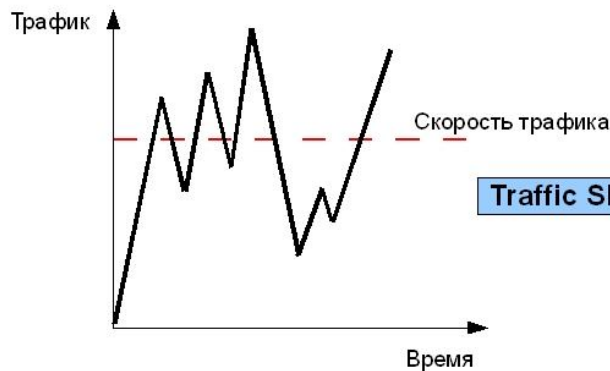


С Traffic Policing



Traffic Policing

Без Traffic Shaping



С Traffic Shaping



Traffic Shaping

# Управление полосой пропускания

**Bandwidth control** – служит для управления полосой пропускания входящего и исходящего трафика на портах коммутаторов D-Link (использует для ограничения скорости механизм **Traffic Policing**).


Администратор может вручную устанавливать требуемую скорость соединения на порте с шагом от 8 Кбит/с.

```
config bandwidth_control [<portlist> | all] {rx_rate [no_limit | <value 8-10240000>]  
| tx_rate [no_limit | <value 8-10240000>]}
```

**Per-flow bandwidth control** – более гибкое решение, позволяет ограничивать полосу пропускания не всему трафику на интерфейсе, а конкретным потокам данных, определенным администратором сети.

Функция per-flow Bandwidth control использует механизм списков управления доступом (ACL) для просмотра определенного типа трафика и ограничения для него полосы пропускания.





**Успехов  
в применении!  
Спасибо  
за внимание!**