

18. Задръстване и управление на потоците в мрежа

Задръстване

Когато броят на изпратените пакети в мрежата надвиши капацитета на мрежата се получава задръстване (**congestion**).

Задръстването може да се причини от няколко фактора.

Ако в даден маршрутизатор започват да постъпват пакети от три или четири линии и всички те трябва да се изпратят по една и съща линия ще се образува опашка. При недостатъчно памет ще се изгубят пакети.

Задръстване

Бавни процесори също могат да причинят задръстване.

Ако маршрутизаторите не могат да обработват достатъчно бързо пристигащите пакети отново ще се образуват опашки, въпреки че линиите не използват докрай своя капацитет.

Управление на натоварването vs. управление на потоците

Разликата между управление на натоварването и управление на потоците е:

- **управлението на натоварването** е свързано с това цялата мрежа да може да се справи с определен трафик - то е на глобално ниво, засяга всички възли в мрежата, съхраняването и препращането на пакетите в маршрутизаторите и всички останали фактори, които намаляват капацитета на мрежата.
- **управлението на потоците** е свързано с двупосочния трафик между даден източник и приемник. То е свързано с това бърз източник да не изпраща продължително данни, по-бързо отколкото приемника може да ги обработи.

натоварване vs. потоци

Управлението на потока обикновено се осъществява чрез обратна връзка от приемника към източника, по която се съобщава на източника за състоянието на приемника.

То се осъществява най-вече на транспортно ниво.

Недостиг на буфери

Броят на буферите в един възел на мрежата е ограничен.

Пристигнал пакет постъпва във входен буфер, след това преминава в изходен буфер, след това се изпраща потвърждение към източника.

Нека възелът има K изходни буфера, m е максималният брой буфери на една изходна линия, s е броят на изходните линии.

Ако $K = m$, тогава е възможно всички изходни буфери да се ангажират с една натоварена линия, което ще доведе до задръстване.

Оптимално натоварване на възела се постига при $m = K / \sqrt{s}$

С една изходна линия може също да се свърже минимален брой буфери, които не могат да се ползват от други линии.

Отстраняване на пакети (drop)

Друга стратегия за избягване на задръстването е маршрутизаторите да **отстраняват (drop) пакети** по желание.

Най-просто е да се отстраняват произволни пакети.

Друга възможност е пакети да се отстраняват в зависимост от **вида на предаваната информация**.

При **трансфер на файлове** има смисъл да се отстраняват по-новите пакети, тъй като отстраняването на стар пакет може да доведе до повторно предаване на следващите го пакети, ако получателят приема пакети само с поредни номера.

При трансфер на **мултимедия**, по-новите пакети са по-важни от по-старите, тъй като се цели минимално закъснение.

Ограничаване на пакетите в мрежата

Трета стратегия е ограничаването на общия брой пакети в мрежата.

За целта в мрежата се пускат служебни пакети, които представляват разрешения за изпращане на пакет.

За да изпрати пакет един хост, прилежащият му маршрутизатор трябва да улови разрешение.

Обратно, когато един хост получи пакет, разрешението се възстановява.

По този начин броят на пакетите в мрежата се контролира от броя на разрешенията. Този алгоритъм е сложен за поддръжка.

Регулиране на входа

- Последната стратегия е регулиране на входа при задръстване.
- Във всеки маршрутизатор се следи натовареността на изходните линии. Когато се надвиши някакъв праг на натовареност линията се слага в състояние на предупреждение.
- Когато линията е в такова състояние, в обратна посока се изпращат специални задръстващи пакети. Чрез тях изпращача разбира, че трябва да намали скоростта на предаване.
- В крайна сметка задръстващите пакети достигат до хоста, който генерира пакетите, довели до задръстването и го уведомяват да намали темпото на предаване.
- Проблем е, че регулирането на входа при задръстване не се прави на транспортно ниво, където е известен пътя между хостовете, и затова е трудно задръстващите пакети да намерят пътя си до хоста-източник.

deadlock

Друг голям проблем при управление на натоварването на мрежите е проблемът **deadlock**.

Да разгледаме пример с три възела *A*, *B* и *C*. Буферите и на трите възела са пълни. *A* има пакети за предаване към *B*, *B* има за *C*, а *C* има за *A*. По този начин трите възела са блокирани. Кръгът от блокирани възли може да е много по-голям.

Проблемът е, че трудно се установява наличието на deadlock в мрежата. Поради това deadlock трябва да се предотвратява.

Управление на потока на канално ниво (Flow Control)

Друг проблем, който е свързан с управлението на обмена на канално ниво е източникът да изпраща кадри по-бързо, отколкото те могат да бъдат приети от получателя.

Този проблем се решава чрез буфериране на кадрите (нареждане на опашка), но при запълване на опашката пристигналите кадри се губят.

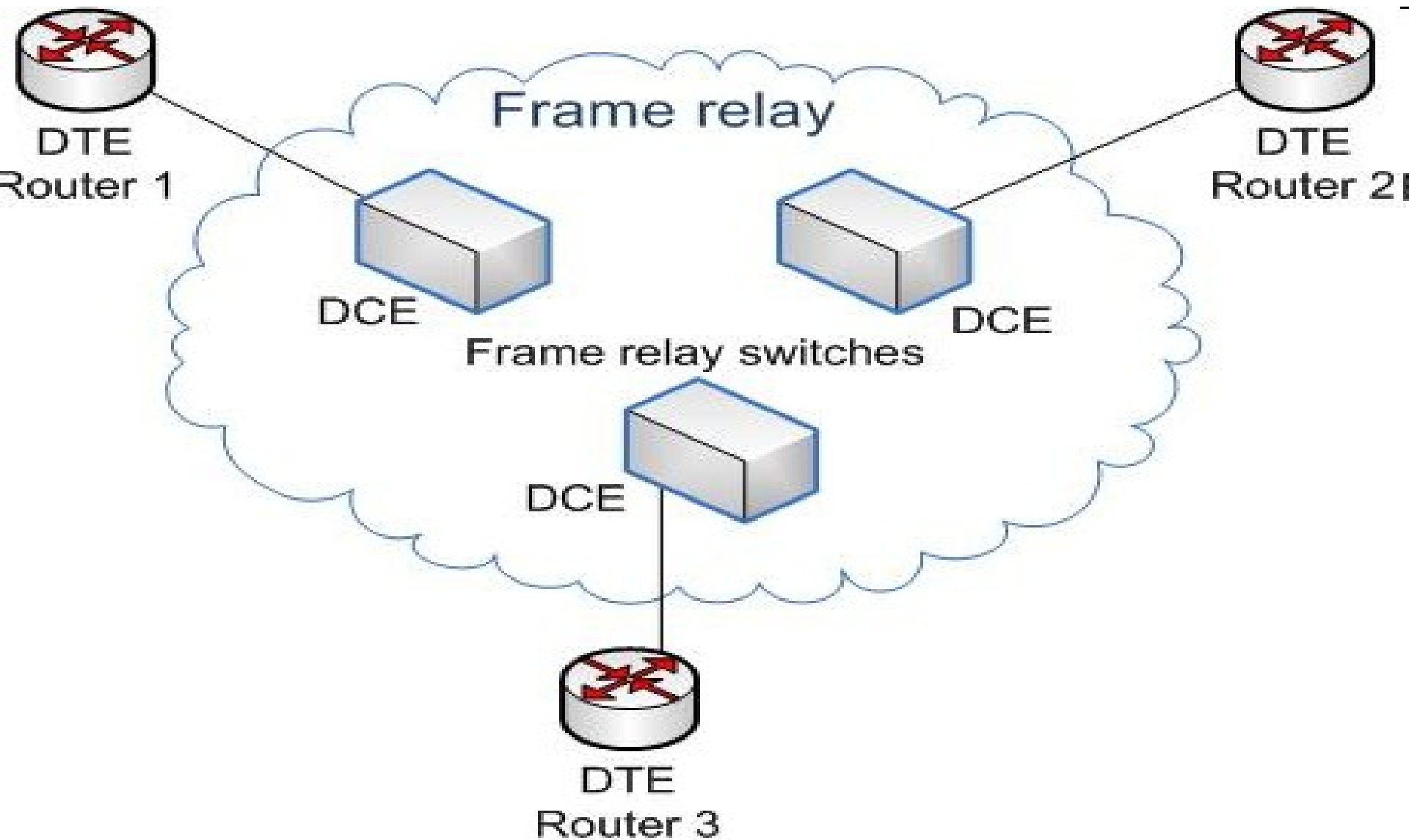
Управление на потока на канално НИВО

Въвеждат се механизми за управление на потока от кадри, който осигурява обратна информация на източника за темпа на предаване.

Обикновено механизмите по управление на обмена се изпълняват в транспортния слой над по-големи информационни единици, обхващащи последователност от сегменти.

Пример за Flow Control на канално ниво е Frame Relay (NBMA).

Frame Relay



Frame Relay. Flow Control

Frame Relay комутаторите създават виртуални канали за свързване на отдалечени LANs или отделни машини. Нарича се **Non Broadcast Multiple Access**, защото за потребителите е „облак“ - МА, но под него са комутаторите свързани „точка-точка“ в някаква топология - NB.

Евтина алтернатива на наетите линии (leased lines). Беше популярна преди появата на MAN.

В заглавието на кадъра (frame) има и следните битове, решаващи проблема със задръстванията (congestion control):

FECN=Forward Explicit Congestion Notification bit

BECN=Backward Explicit Congestion Notification bit

DE=Discard Eligibility bit

Frame Relay. Flow Control.

Frame Relay мрежата е с опростен протокол, липсва flow-control връзка по връзка. Затова производителността на мрежата като цяло е податлива на „скокове“ в натоварването от някои услуги - **bursts**. Затова се нуждаят от механизми за **congestion control** – управление на задръстванията.

Congestion control включва следните елементи:

Admission Control. Мрежата решава дали да приеме нова заявка за връзка в зависимост от описания в заявката трафик (**traffic descriptor**), сравнявайки го с остатъчния капацитет. **Traffic descriptor** се състои от три компонента:

Frame Relay. Flow Control.

- **Committed Information Rate (CIR)**. Средната скорост (в bit/s), при която мрежата гарантира трансфера на информационни единици по време на интервал T : $T = Bc/CIR$.
- **Committed Burst Size (Bc)**. Максималният брой предадени информационни единици през интервала T .
- **Excess Burst Size (BE)**. Максималният брой непотвърдени информационни единици (в bits), които мрежата ще се опита да пренесе по време на интервала.

Frame Relay. Flow Control.

След като мрежата е установила връзката, граничният възел трябва да следи трафика, така че да не се надвишават зададените стойности. Frame relay позволява мрежата да налага информационната скорост за крайните потребители и да изхвърля информация, когато се надвиши заявената скорост.

Политиката на избягване на задръстванията включва непосредствено уведомяване. Така че да се постигне определено качество на услугата **Quality of Service (QOS)**.

Frame Relay. Flow Control.

За това служат специалните битове за контрол на задръстванията: **FECN** и **BECN**.

FECN - Forward Explicit Congestion Notification.

Установява се в 1, когато се наблюдава задръстване в посоката на предаване на кадъра (frame), т.е информира **дестинацията**, че има задръстване.

BECN - Backwards Explicit Congestion Notification.

Установява се в 1, когато се наблюдава задръстване в посоката, обратна на предаване на кадъра (frame), т.е информира **източника**, че има задръстване.