

停止-等待协议SW

回退N帧协议GBN

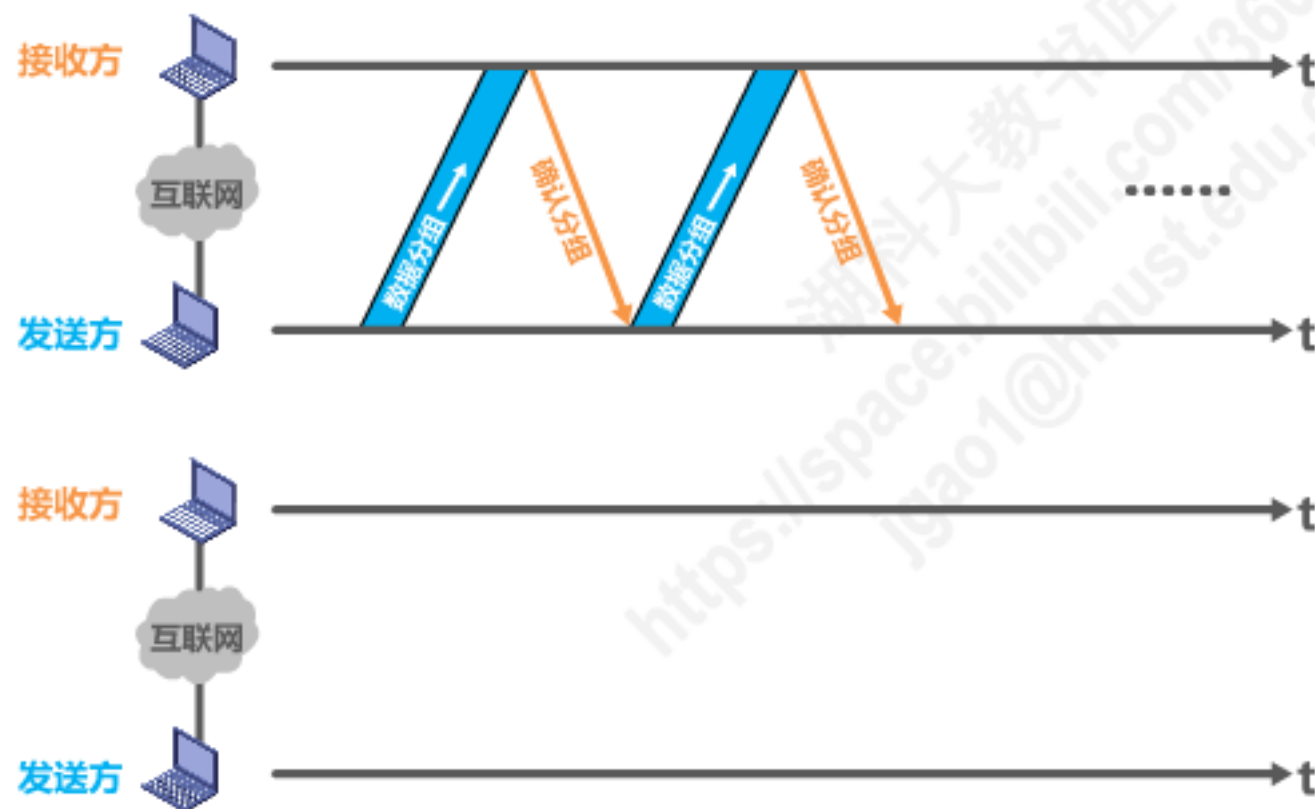
选择重传协议SR



停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

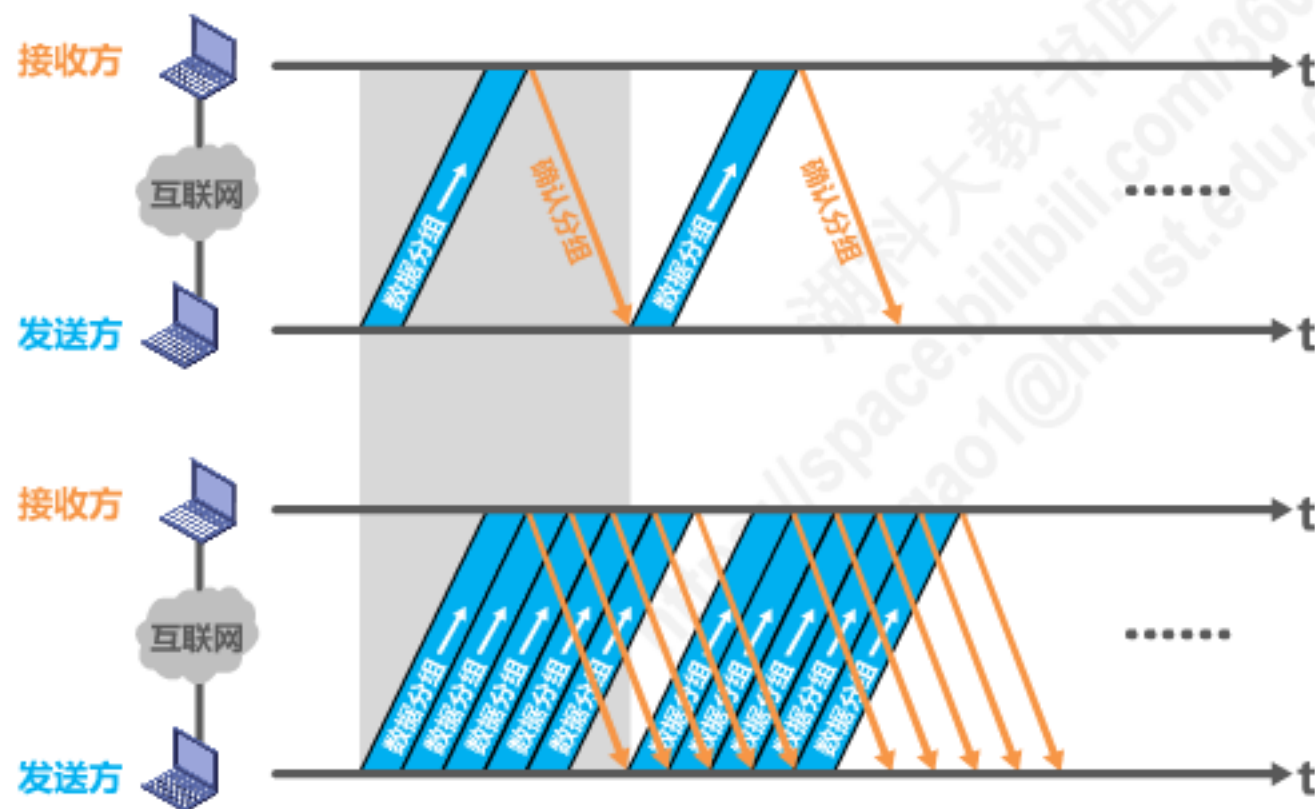


停止-等待协议的信道利用率很低
若出现超时重传，则信道利用率更低

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR



停止-等待协议的信道利用率很低
若出现超时重传，则信道利用率更低

采用流水线传输可提高信道利用率

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；

发送方



互联网



接收方

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

发送方



互联网

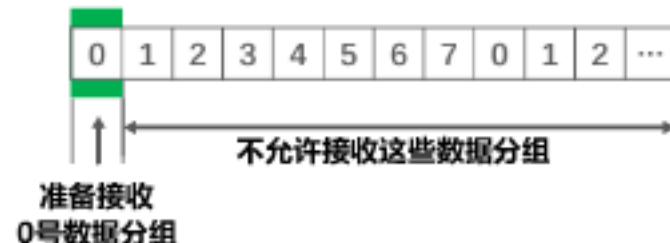


接收方

发送窗口
 $W_T=5$



接收窗口
 $W_R=1$



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；



发送窗口
 $W_T=5$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

接收窗口
 $W_R=1$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

无差错情况

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

发送方

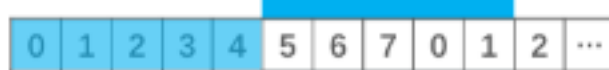


互联网



接收方

发送窗口
 $W_T=5$



接收窗口
 $W_R=1$



无差错情况

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



发送窗口
 $W_T=5$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

接收窗口
 $W_R=1$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

累积确认

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。

发送方



互联网



接收方



发送窗口
 $W_T=5$



接收窗口
 $W_R=1$



累积确认

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



累积确认

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。

发送方



互联网

接收方

发送窗口
 $W_T=5$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

接收窗口
 $W_R=1$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

累积确认

即使确认分组丢失，
发送方也可能不必重传！

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

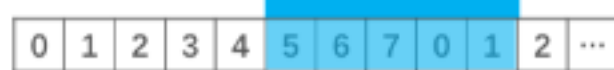
选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



发送窗口
 $W_T=5$



接收窗口
 $W_R=1$



有差错情况

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

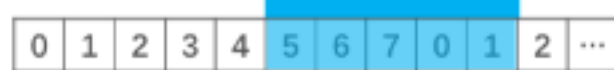
选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



发送窗口
 $W_T=5$



接收窗口
 $W_R=1$



有差错情况

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



发送方收到重复的确认，就知道之前所发送的数据分组出现了差错，于是可以不等超时计时器超时就立刻重传！

有差错情况

至于收到几个重复确认就立刻重传，由具体实现决定。

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

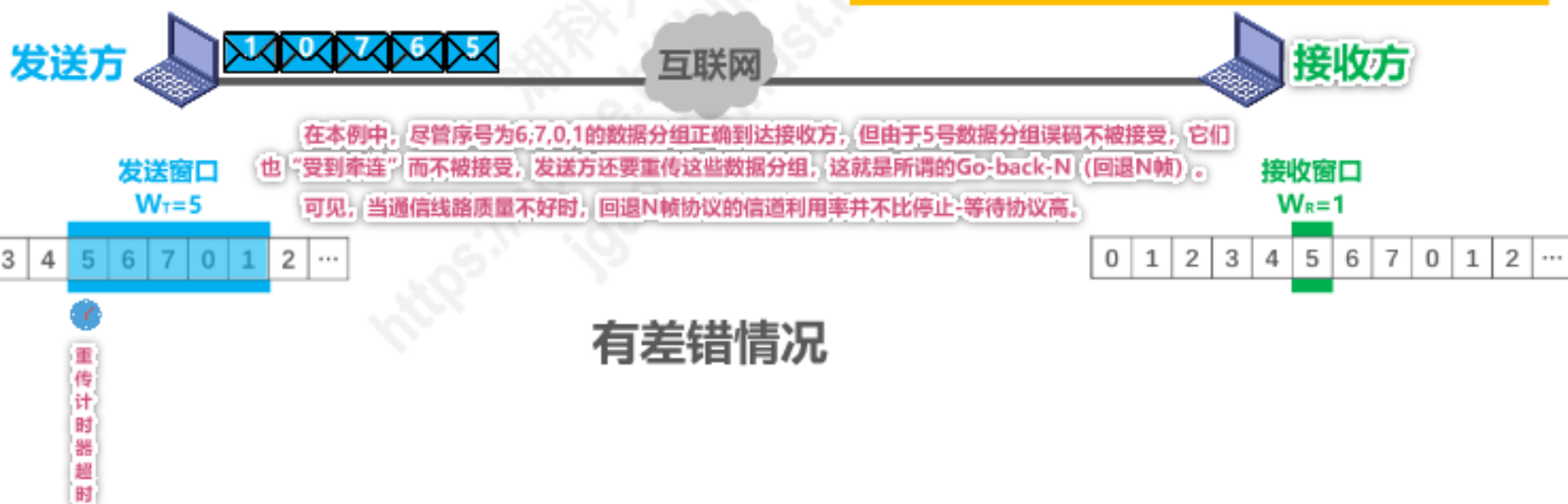
停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，本例取 $W_T=5$ ；
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

接收方不一定要对收到的数据分组逐个发送确认，而是可以在收到几个数据分组后（由具体实现决定），对按序到达的最后一个数据分组发送确认。ACKn表示序号为n及以前的所有数据分组都已正确接收。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，若 W_T 超过取值范围，例如 $W_T=8$ ，会出现什么情况？
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

发送方



互联网



接收方

发送窗口

$W_T=8$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

接收窗口

$W_R=1$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，若 W_T 超过取值范围，例如 $W_T=8$ ，会出现什么情况？
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；



发送窗口
 $W_T=8$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

接收窗口
 $W_R=1$

0	1	2	3	4	5	6	7	0	1	2	...
---	---	---	---	---	---	---	---	---	---	---	-----

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，若 W_T 超过取值范围，例如 $W_T=8$ ，会出现什么情况？
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；

发送方



互联网



接收方

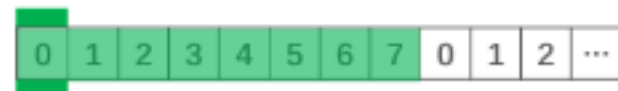
发送窗口

$W_T=8$



接收窗口

$W_R=1$



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

1. 采用3个比特给分组编序号，即序号0~7；
2. 发送窗口的尺寸 W_T 的取值： $1 < W_T \leq 2^3 - 1$ ，若 W_T 超过取值范围，例如 $W_T=8$ ，会出现什么情况？
3. 接收窗口的尺寸 W_R 的取值： $W_R=1$ ；



重传计时器超时

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

发送方

接收方

湖科大教书匠
<https://space.bilibili.com/360951512>
jgao1@hust.edu.cn

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

发送方

■ 发送窗口尺寸 W_T 的取值范围是 $1 < W_T \leq 2^n - 1$

其中, n 是构成分组序号的比特数量。

- ☐ $W_T = 1$ 停止-等待协议
- ☐ $W_T > 2^n - 1$ 接收方无法分辨新、旧数据分组

接收方

■ 接收方的接收窗口尺寸 W_R 的取值范围是 $W_R = 1$
因此接收方只能按序接收数据分组。

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

发送方

- 发送窗口尺寸 W_T 的取值范围是 $1 < W_T \leq 2^n - 1$
其中, n 是构成分组序号的比特数量。
 - ☐ $W_T = 1$ 停止-等待协议
 - ☐ $W_T > 2^n - 1$ 接收方无法分辨新、旧数据分组
- 发送方可在未收到接收方确认分组的情况下, 将序号落在发送窗口内的多个数据分组全部发送出去;
- 发送方只有收到对已发送数据分组的确认时, 发送窗口才能向前相应滑动;
- 发送方收到多个重复确认时, 可在重传计时器超时前尽早开始重传, 由具体实现决定。
- 发送方发送窗口内某个已发送的数据分组产生超时重发时, 其后续在发送窗口内且已发送的数据分组也必须全部重传, 这就是回退N帧协议名称的由来。

接收方

- 接收方的接收窗口尺寸 W_R 的取值范围是 $W_R = 1$
因此接收方只能按序接收数据分组。
- 接收方只接收序号落在接收窗口内且无误码的数据分组, 并且将接收窗口向前滑动一个位置, 与此同时给发送方发回相应的确认分组。为了减少开销, 接收方不一定每收到一个按序到达且无误码的数据分组就给发送方发回一个确认分组,
 - ☐ 而是可以在连续收到好几个按序到达且无误码的数据分组后(由具体实现决定), 才针对最后一个数据分组发送确认分组, 这称为累积确认;
 - ☐ 或者可以在自己有数据分组要发送时才对之前按序接收且无误码的数据分组进行捎带确认;
- 接收方收到未按序到达的数据分组, 除丢弃外, 还要对最近按序接收的数据分组进行确认;

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

【2009年 题35】数据链路层使用后退N帧（GBN）协议，发送方已经发送了编号为0~7的帧。当计时器超时，若发送方只收到0、2、3号帧的确认，则发送方需要重发的帧数是 **C**

A. 2

B. 3

C. 4

D. 5

【解析】

(1) “发送方只收到0、2、3号帧的确认”表明接收方正确接收了0~3号帧，并针对它们中的每一个发送了确认帧，只不过针对1号帧的确认帧丢失了（这是题目中的陷阱，但又没有相应的选项，所以迷惑性并不是很大）；

(2) 截止到计时器超时，发送方只收到了针对0~3号帧的确认，而发送方之前已经发送了0~7号帧，因此应该从4号帧开始重传，即重传之前已经发送过的4、5、6、7号帧，共计重传4个帧。

3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

【2009年 题35】数据链路层使用后退N帧（GBN）协议，发送方已经发送了编号为0~7的帧。当计时器超时，若发送方只收到0、2、3号帧的确认，则发送方需要重发的帧数是 **C**

A. 2

B. 3

C. 4

D. 5

【解析】

(1) “发送方只收到0、2、3号帧的确认”表明接收方正确接收了0~3号帧，并针对它们中的每一个发送了确认帧，只不过针对1号帧的确认帧丢失了（这是题目中的陷阱，但又没有相应的选项，所以迷惑性并不是很大）；

(2) 截止到计时器超时，发送方只收到了针对0~3号帧的确认，而发送方之前已经发送了0~7号帧，因此应该从4号帧开始重传，即重传之前已经发送过的4、5、6、7号帧，共计重传4个帧。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

【2009年 题35】数据链路层使用后退N帧（GBN）协议，发送方已经发送了编号为0~7的帧。当计时器超时，若发送方只收到0、2、3号帧的确认，则发送方需要重发的帧数是 **C**

A. 2

B. 3

C. 4

D. 5

【解析】

(1) “发送方只收到0、2、3号帧的确认”表明接收方正确接收了0~3号帧，并针对它们中的每一个发送了确认帧，只不过针对1号帧的确认帧丢失了（这是题目中的陷阱，但又没有相应的选项，所以迷惑性并不是很大）；

(2) 截止到计时器超时，发送方只收到了针对0~3号帧的确认，而发送方之前已经发送了0~7号帧，因此应该从4号帧开始重传，即重传之前已经发送过的4、5、6、7号帧，共计重传4个帧。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

回退N帧协议GBN

选择重传协议SR

【2009年 题35】数据链路层使用后退N帧（GBN）协议，发送方已经发送了编号为0~7的帧。当计时器超时，若发送方只收到0、2、3号帧的确认，则发送方需要重发的帧数是 **C**

A. 2

B. 3

C. 4

D. 5

【解析】

(1) “发送方只收到0、2、3号帧的确认”表明接收方正确接收了0~3号帧，并针对它们中的每一个发送了确认帧，只不过针对1号帧的确认帧丢失了（这是题目中的陷阱，但又没有相应的选项，所以迷惑性并不是很大）；

(2) 截止到计时器超时，发送方只收到了针对0~3号帧的确认，而发送方之前已经发送了0~7号帧，因此应该从4号帧开始重传，即重传之前已经发送过的4、5、6、7号帧，共计重传4个帧。



3.4.3 可靠传输的实现机制 —— 回退N帧协议GBN(Go-Back-N)

停止-等待协议SW

发送方

- 发送窗口尺寸 W_T 的取值范围是 $1 < W_T \leq 2^n - 1$
其中, n 是构成分组序号的比特数量。
 - ☐ $W_T = 1$ 停止-等待协议
 - ☐ $W_T > 2^n - 1$ 接收方无法分辨新、旧数据分组
- 发送方可在未收到接收方确认分组的情况下, 将序号落在发送窗口内的多个数据分组全部发送出去;
- 发送方只有收到对已发送数据分组的确认时, 发送窗口才能向前相应滑动;
- 发送方收到多个重复确认时, 可在重传计时器超时前尽早开始重传, 由具体实现决定。
- 发送方发送窗口内某个已发送的数据分组产生超时重发时, 其后续在发送窗口内且已发送的数据分组也必须全部重传, 这就是回退N帧协议名称的由来。

回退N帧协议GBN

接收方

- 接收方的接收窗口尺寸 W_R 的取值范围是 $W_R = 1$
因此接收方只能按序接收数据分组。
- 接收方只接收序号落在接收窗口内且无误码的数据分组, 并且将接收窗口向前滑动一个位置, 与此同时给发送方发回相应的确认分组。为了减少开销, 接收方不一定每收到一个按序到达且无误码的数据分组就给发送方发回一个确认分组。
 - ☐ 而是可以在连续收到好几个按序到达且无误码的数据分组后 (由具体实现决定), 才针对最后一个数据分组发送确认分组, 这称为累积确认;
 - ☐ 或者可以在自己已有数据分组要发送时才对之前按序接收且无误码的数据分组进行捎带确认;
- 接收方收到未按序到达的数据分组, 除丢弃外, 还要对最近按序接收的数据分组进行确认;

选择重传协议SR



回退N帧协议在流水线传输的基础上利用发送窗口来限制发送方连续发送数据分组的数量, 是一种连续ARQ协议。

在协议的工作过程中发送窗口和接收窗口不断向前滑动, 因此这类协议又称为滑动窗口协议。

由于回退N帧协议的特性, 当通信线路质量不好时, 其信道利用率并不比停止-等待协议高。