

画像情報特論 (7)

- アダプテーション (1) 同期再生

2001.05.29

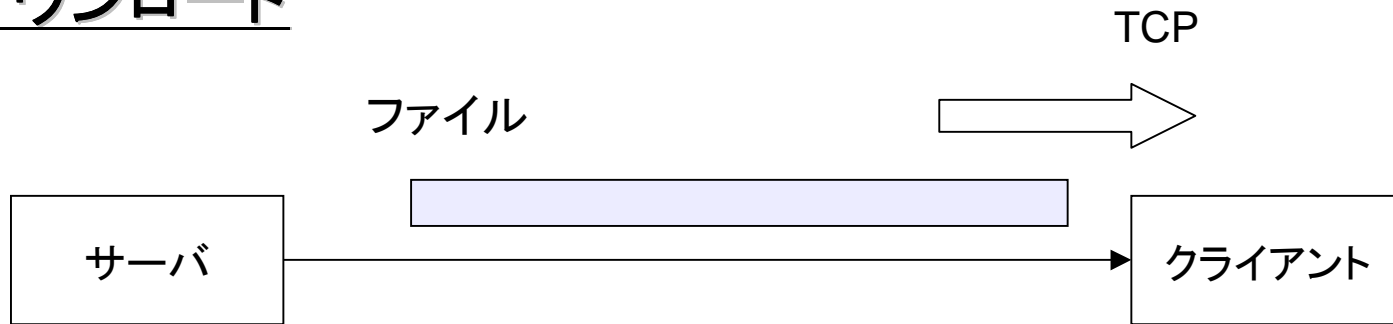
電子情報通信学科 甲藤二郎

E-Mail: katto@katto.comm.waseda.ac.jp

アダプテーション全般

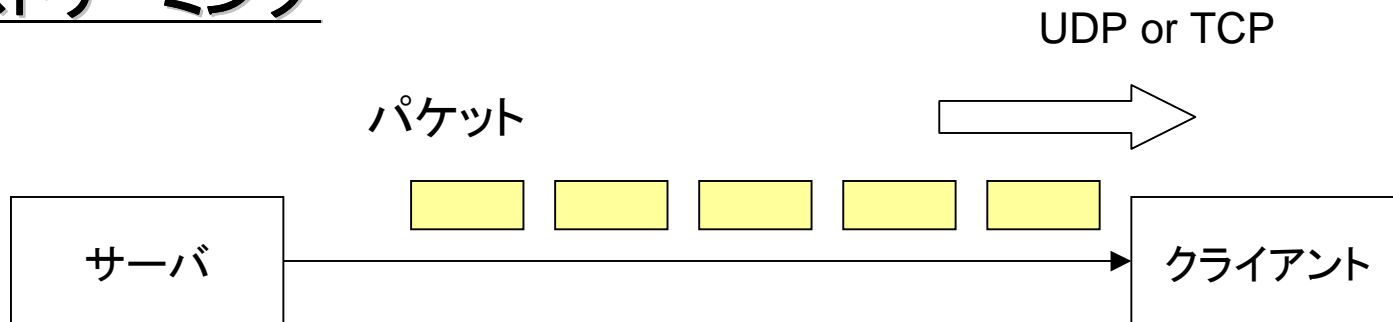
ダウンロードとストリーミング

ダウンロード



ダウンロードしてから同期再生
(待ち時間が我慢できない)

ストリーミング



受信しながら同期再生
(待ち時間がほとんどない)

ストリーミングの問題点

インターネット：もともとリアルタイムメディア用のネットワークではない

1. なめらかな同期再生

到着時間がばらばらのパケットからどのように同期再生するか。

2. パケット廃棄の問題

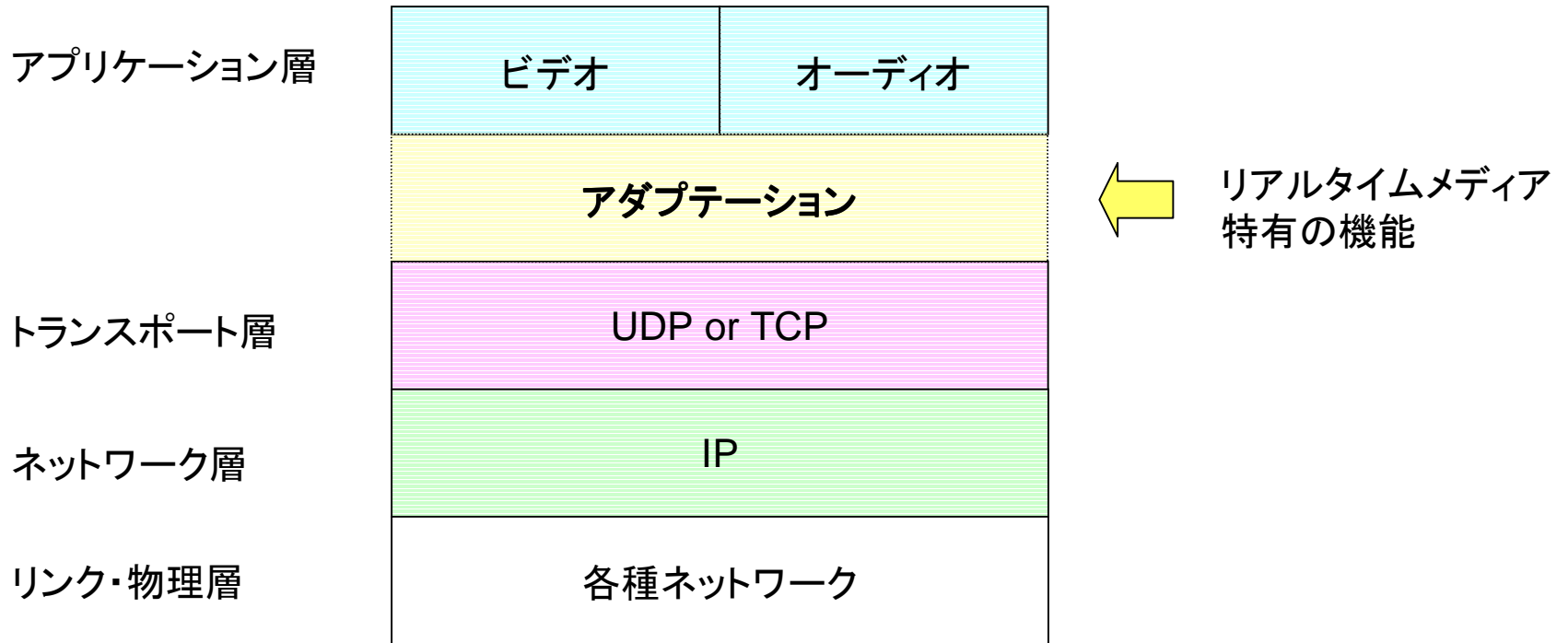
廃棄されたパケットの影響をどのように抑えるか。

3. ふくそうと品質

レートを上げすぎるとふくそうが起こり、下げると品質が劣化する。

アダプテーション

- 同期、廃棄対策、ふくそう制御の実現



具体的な対策

同期再生:

十分なバッファリング (ジッタの吸収)

タイムスタンプ

- 三階層同期 (メディア内同期、メディア間同期、システム間同期)

パケット廃棄対策 (TCPの場合は不要):

- (1) エラーコンシールメント
- (2) 誤りパケットの再送

ふくそう制御:

- (1) TCP フレンドリ (TCP フローとの公平な共存)
- (2) サービス階層化 (Differentiated Services)

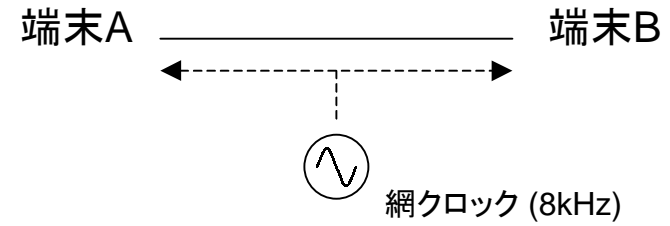
同期再生メカニズム

三種類の同期メカニズム

(1) 共通クロック (狭帯域回線交換)

ネットワーク、あるいは外部から、「信頼できる共通クロック」を提供する → 狭帯域ネットワーク

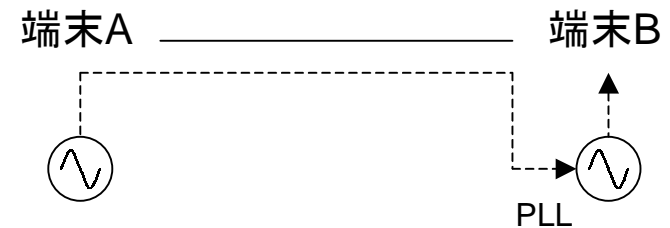
(例) 電話網、ISDN、モバイル



(2) 搬送クロック (広帯域回線交換)

クロックを相手端末に搬送する。受信側は、PLL でクロックを生成 → 広帯域ネットワーク

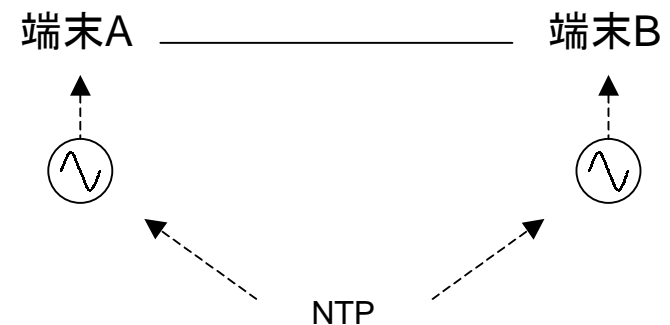
(例) ATM、デジタル放送



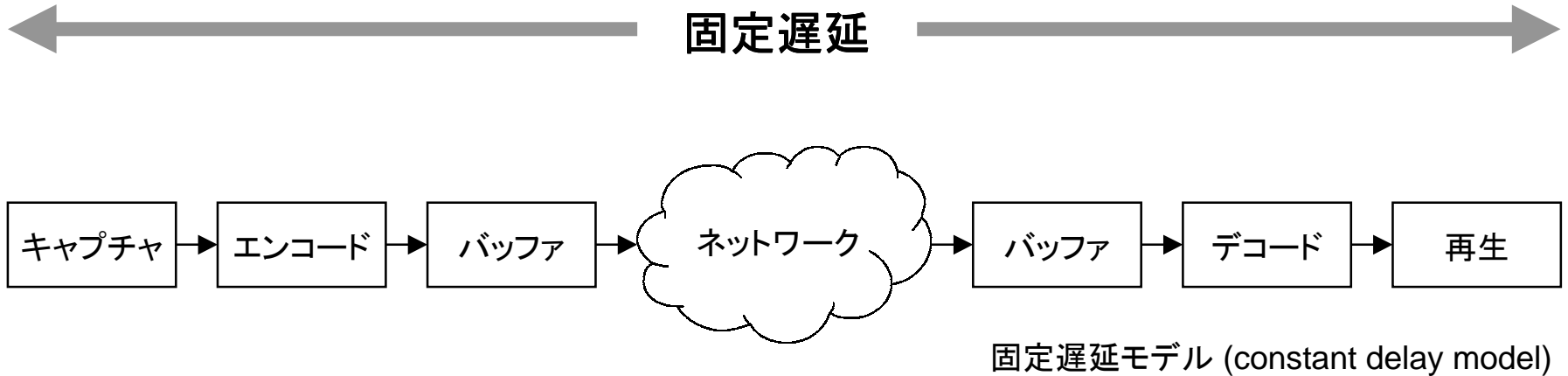
(3) 自走クロック (インターネット)

まず、端末自身の自走クロックを信頼する。端末間同期はNTPを利用 → 精度に問題

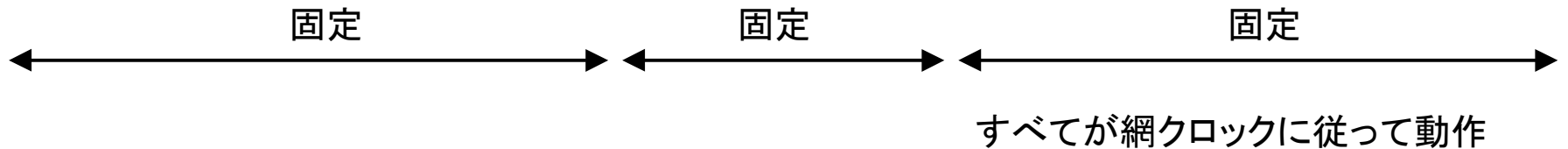
(例) インターネット



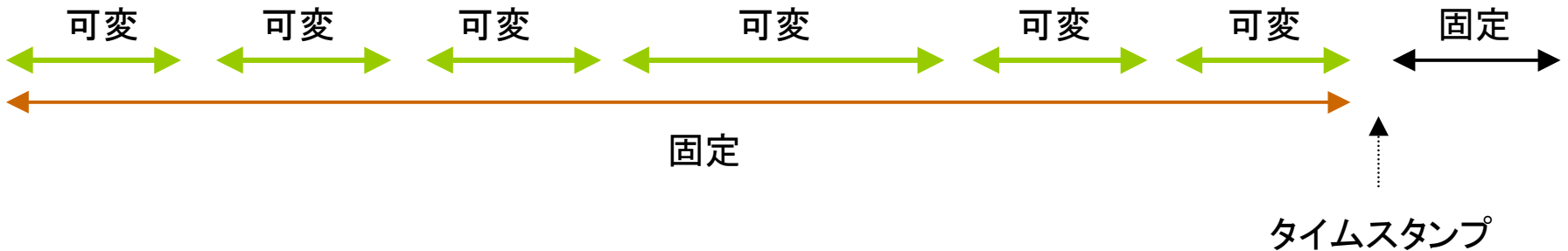
固定遅延モデル (1)



(1) 共通クロック方式 (回線交換)



(2) タイムスタンプ方式 (パケット交換)



固定遅延モデル (2)

- キャプチャから再生までの遅延時間が一定

(1) 共通クロック方式 (電話網、デジタル放送)

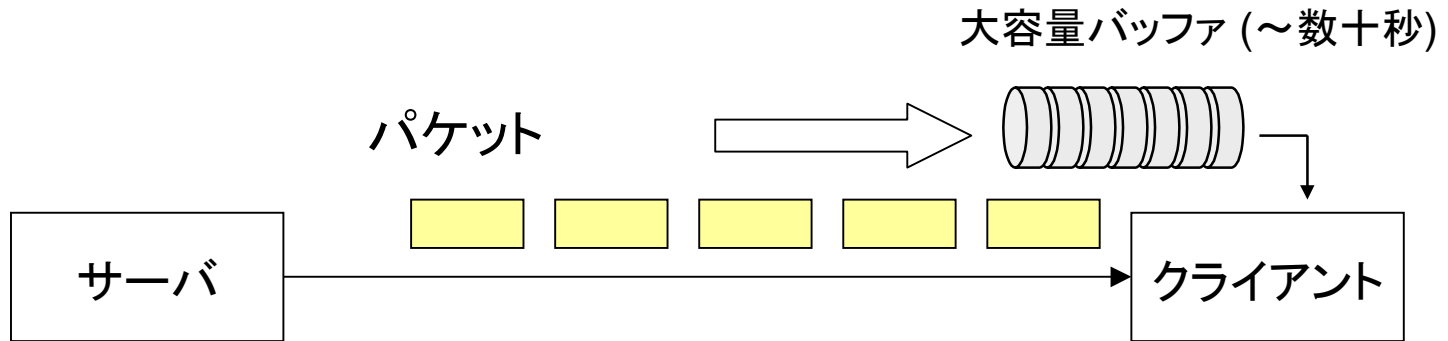
エンコード、デコード、ネットワークの処理時間 (遅延) が一定
→ キャプチャから再生まで固定遅延

(2) 自走クロック方式 (インターネット)

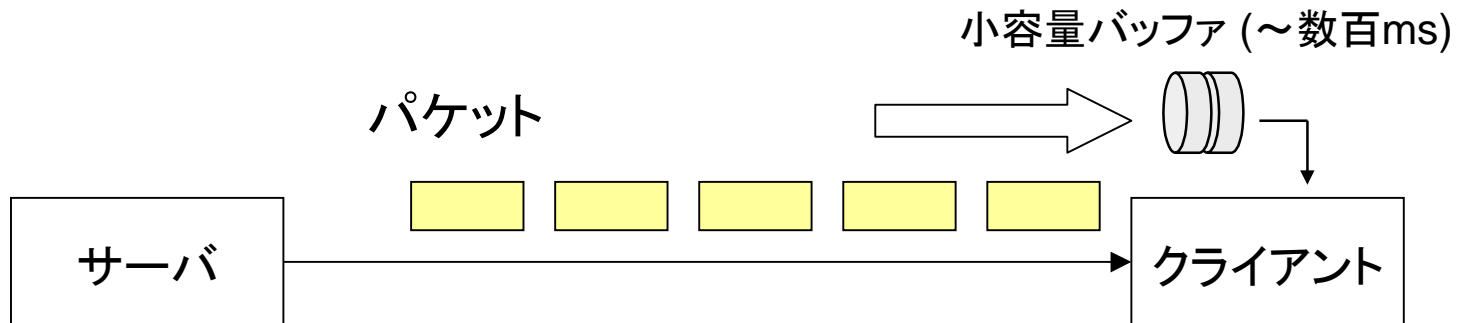
エンコード、デコード、ネットワークの処理時間 (遅延) が可変
→ タイムスタンプ (キャプチャ時刻) で再生時刻を決定
→ 遅延のばらつきを吸収
→ キャプチャから再生まで固定遅延

バッファリング (1)

インターネット放送 (片方向)



インターネット電話 (双方向)



バッファリング (2)

- ジッタ (遅延のばらつき) の吸収

(1) インターネット放送

片方向なので、数十秒間のバッファリングを行ってもユーザは気にならない

→ ユニキャスト時はTCPが利用可能 (パケット廃棄対策が不要)

(注) マルチキャスト時はUDPを使用 (パケット廃棄対策が必要)

(2) インターネット電話

双方向なので、あまり長時間のバッファリングを行うと会話にならない

→ TCPは利用困難

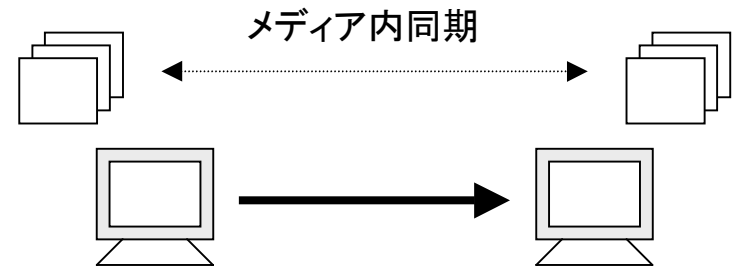
→ UDPを使用 (パケット廃棄対策が必要)

三階層のメディア同期

(1) メディア内同期

一つのメディア (ビデオ、あるいはオーディオ) の同期再生の実現

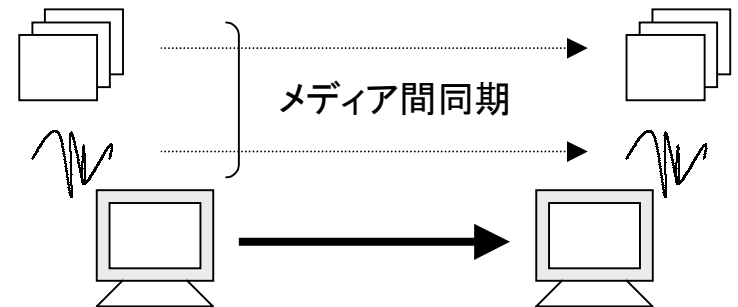
(手段) タイムスタンプ (固定遅延モデル)



(2) メディア間同期

複数のメディア (ビデオとオーディオ) の同期再生の実現 (リップシンク)

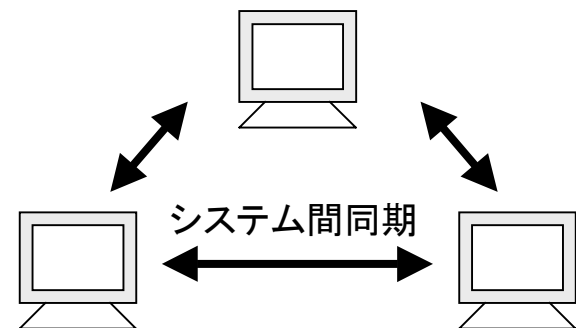
(手段) タイムスタンプ間の関連付け



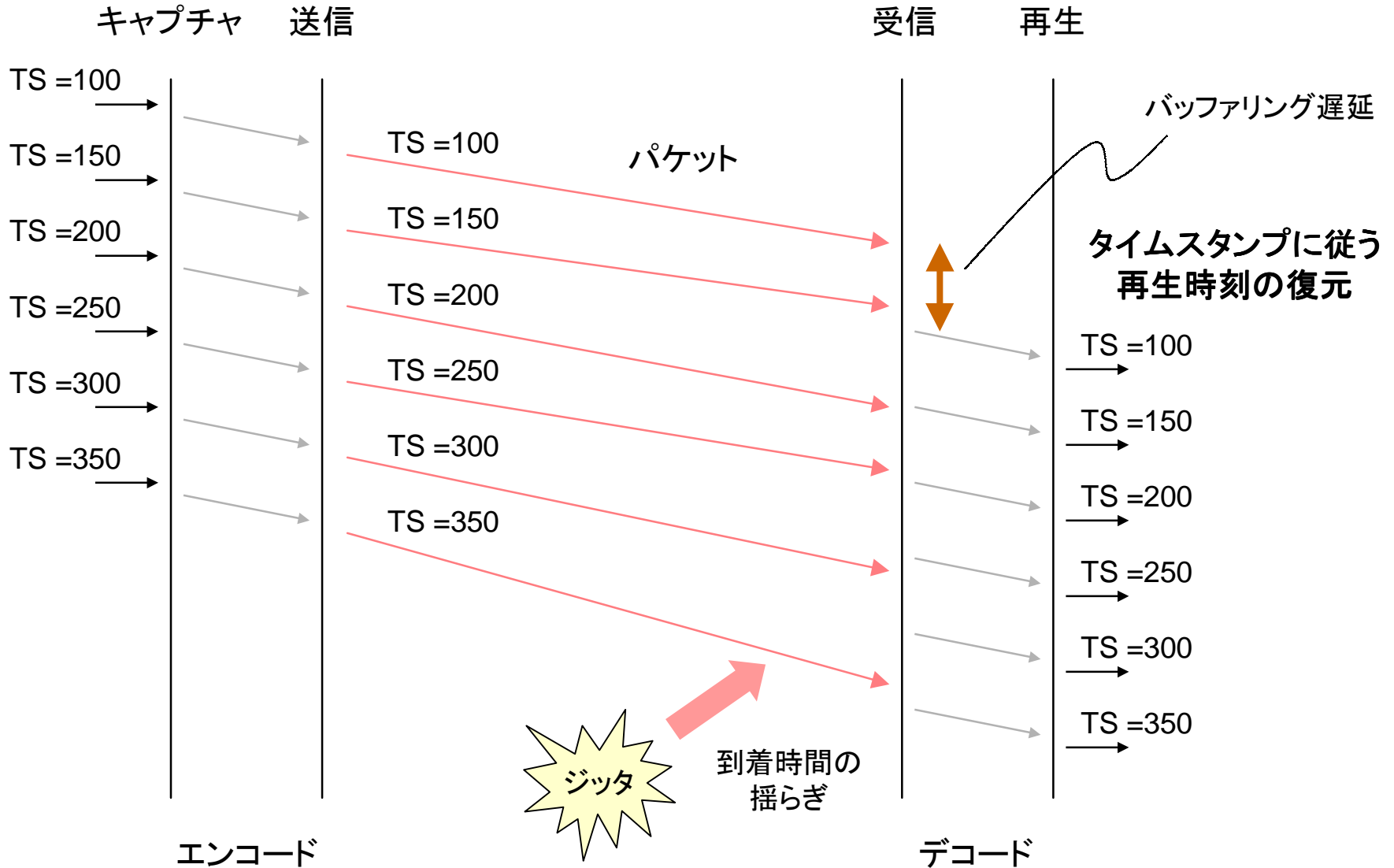
(3) システム間同期

端末毎のシステム時刻の調整

(手段) ネットワーク・タイム・プロトコル

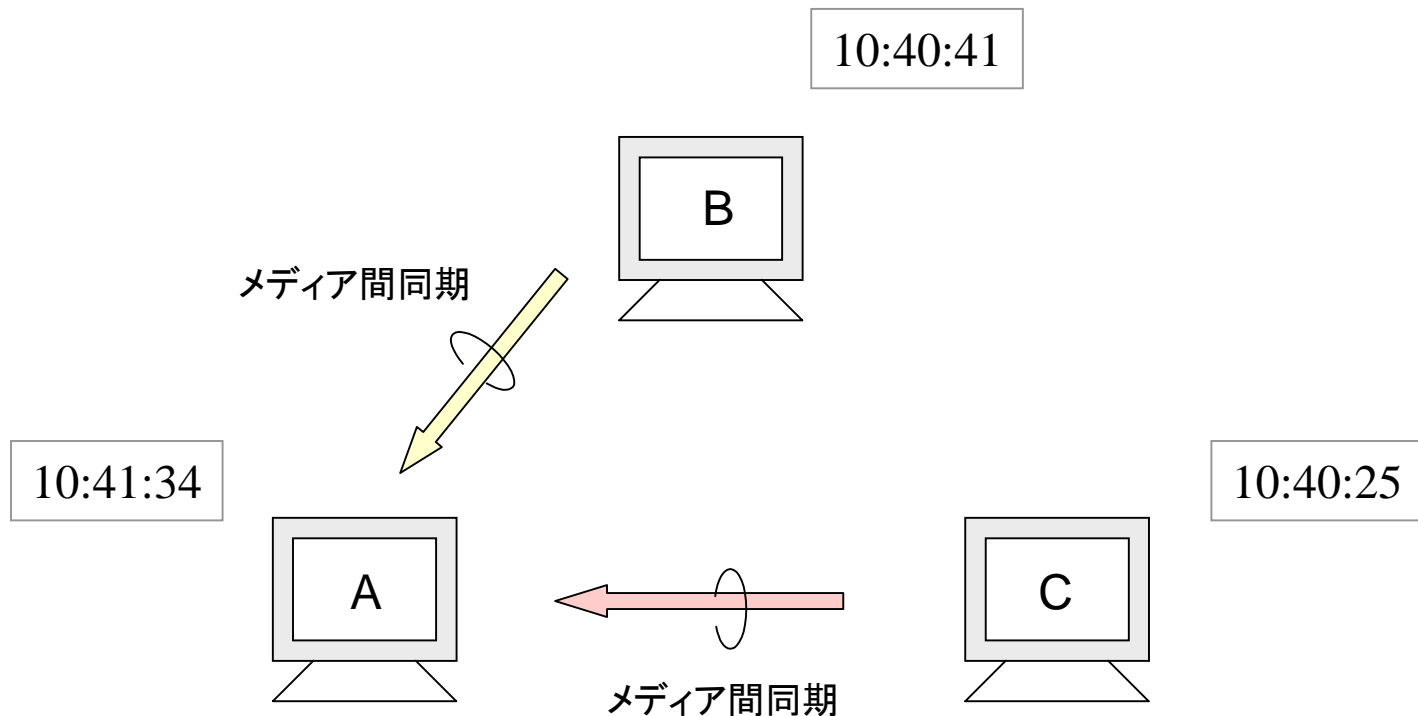


メディア内同期



システム間同期

複数人による実時間会議の問題点

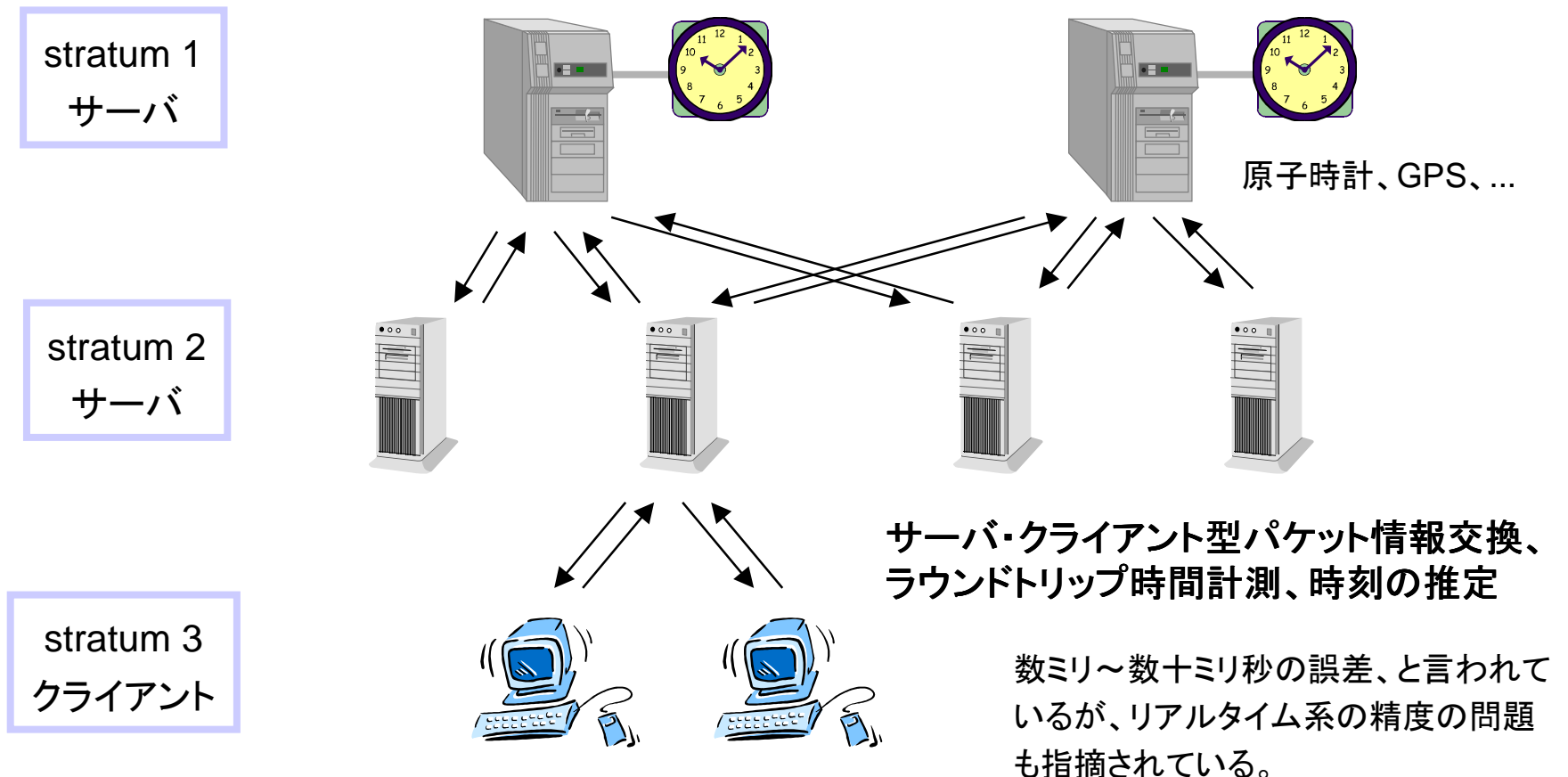


- Bから来るメディアとCから来るメディアの時間軸を整列することができない
- 複数人が同時に話しても、それを同時に再生できない
- 結局、すべての端末の時刻をそろえるしかない

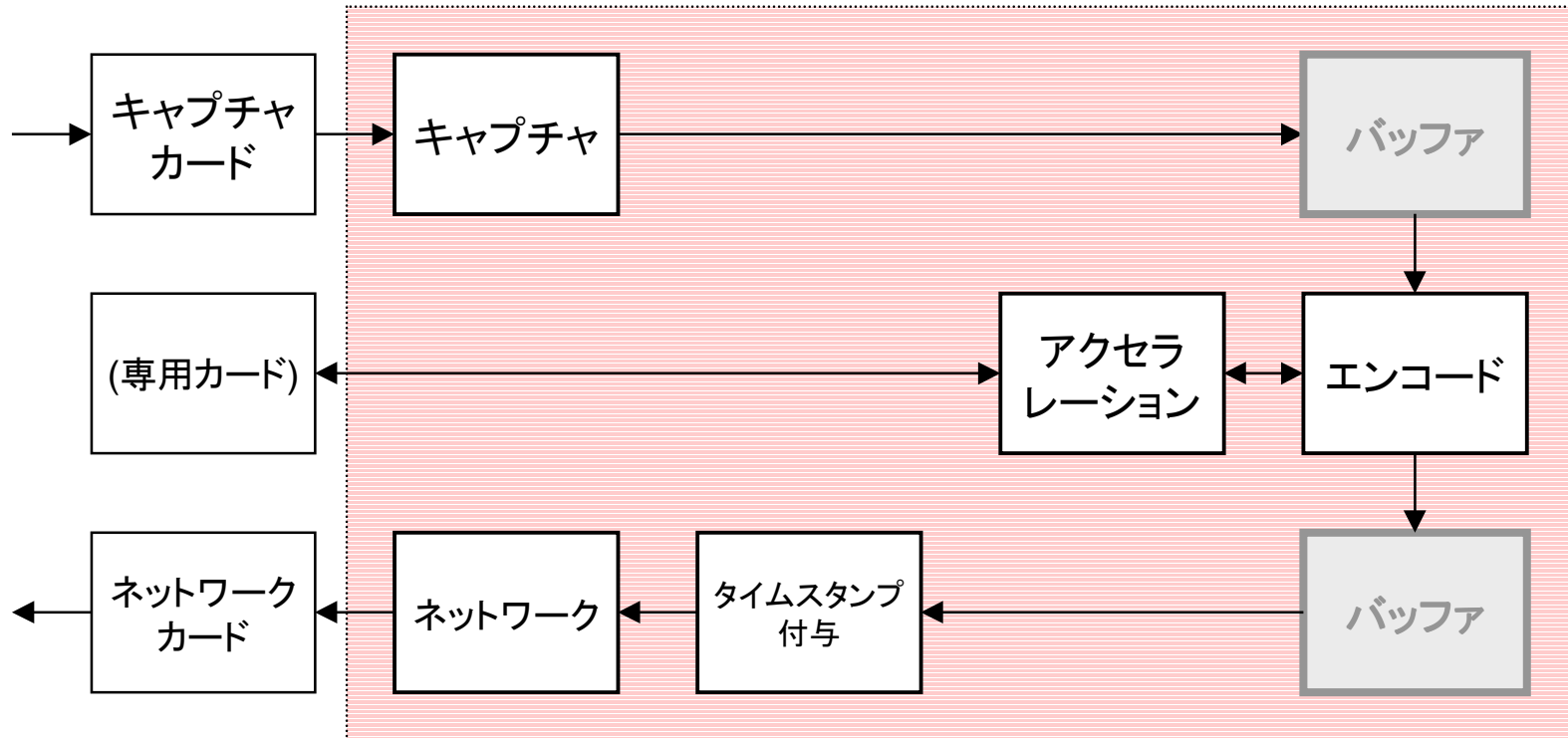
NTP (Network Time Protocol)

ネットワーク上の端末の時刻を合わせるためのプロトコル

(例) UNIX: xntpd、Windows: 桜時計、など...



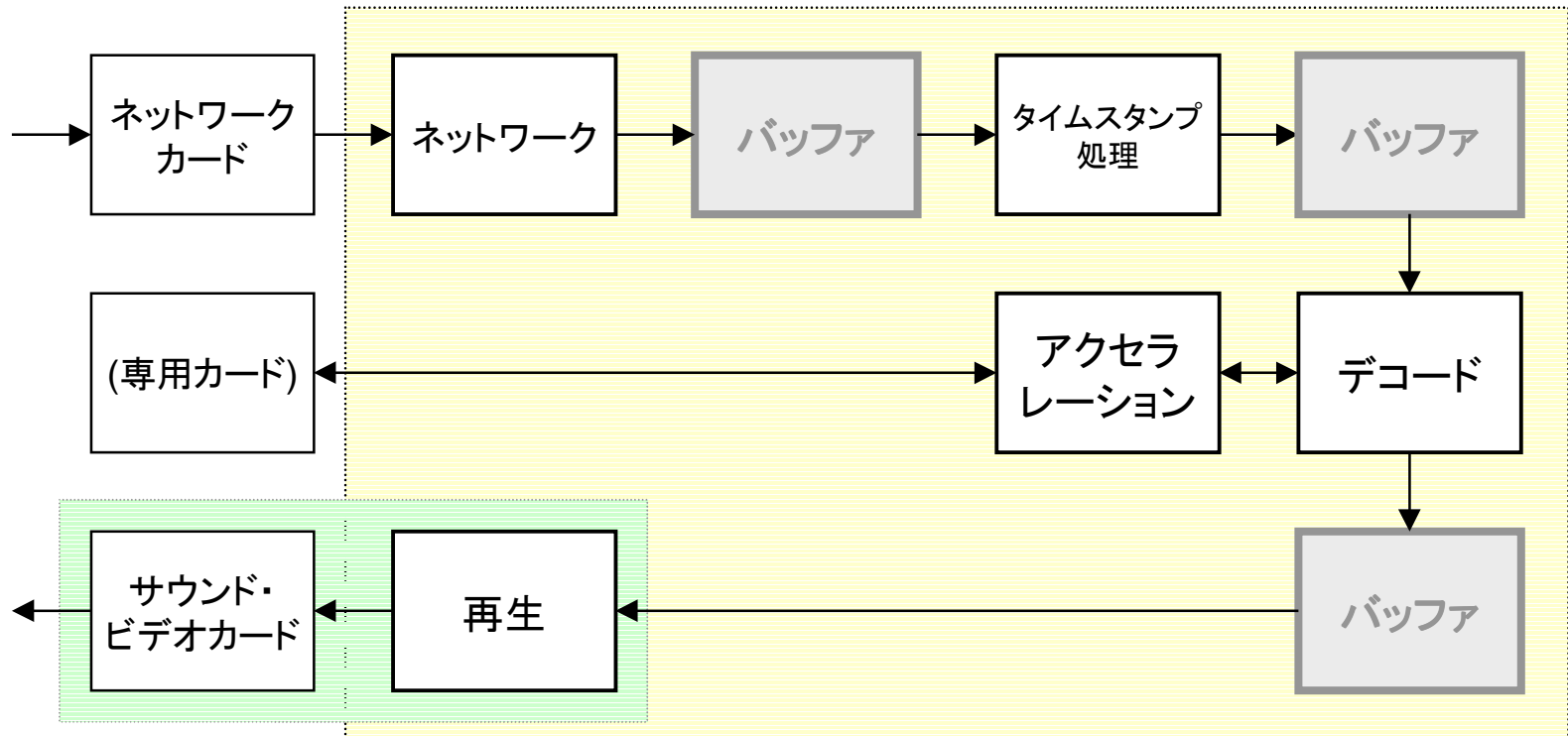
メディア同期の実際 (1. 送信側)



バッファの活用 (実体はメモリ)

バッファ: 冗長構成 (ダブルバッファ、数十ミリ~数百ミリ秒のバッファリング)

メディア同期の実際 (2. 受信側)

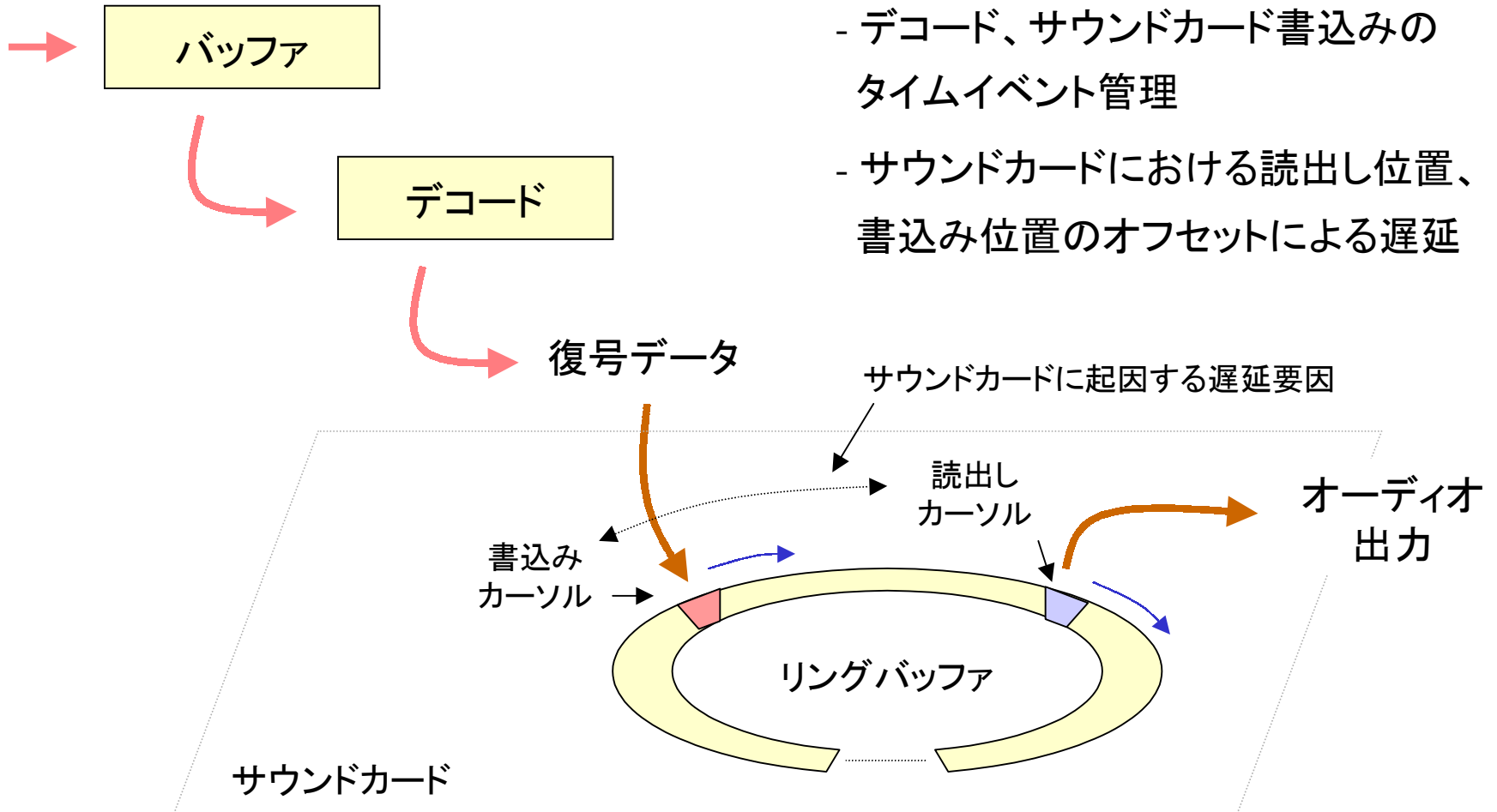


バッファの活用

バッファ: 冗長構成 (ダブルバッファ、数十ミリ～数秒のバッファリング)

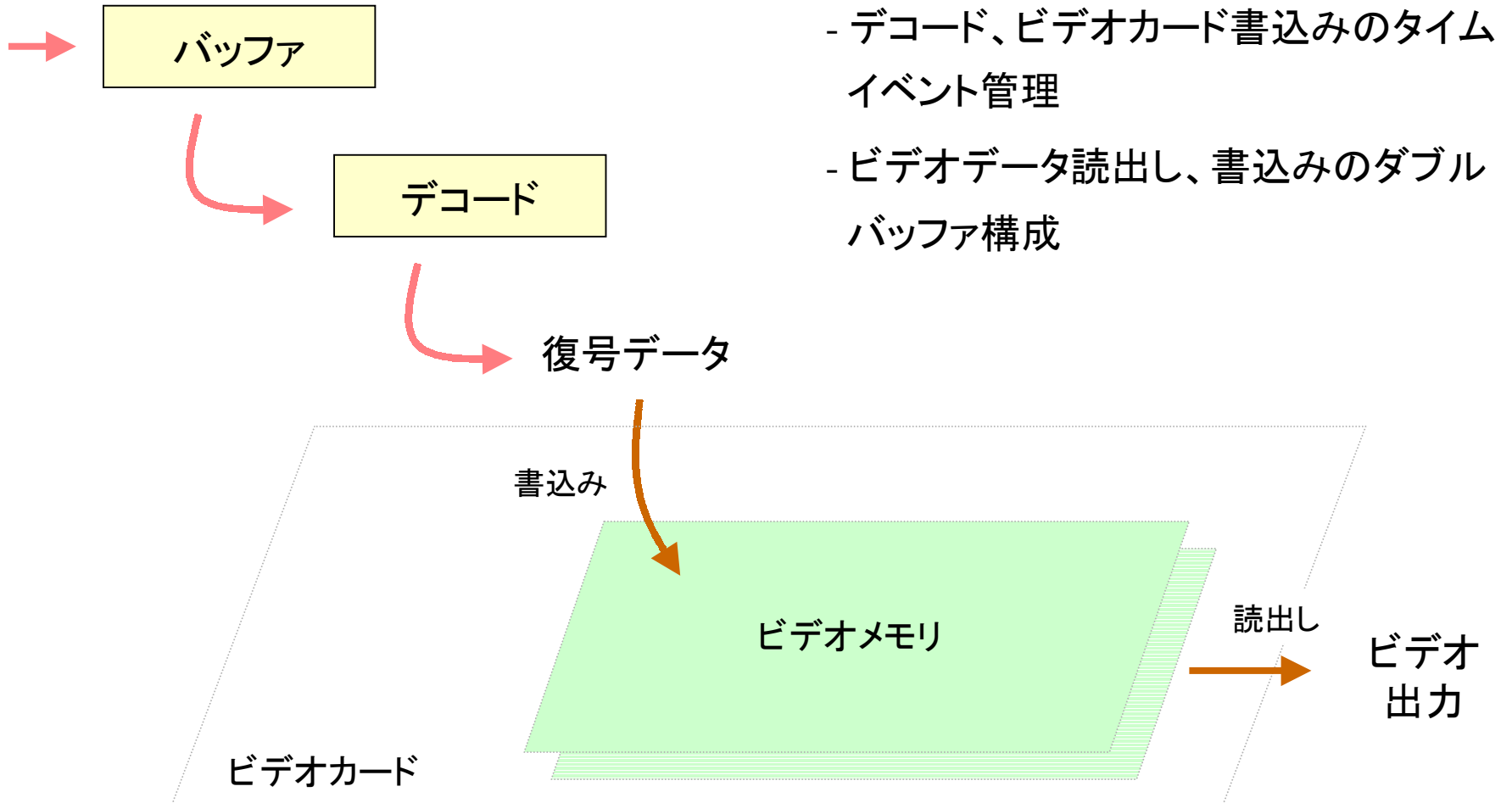
メディア同期の実際 (3)

• オーディオのストリーミング再生



メディア同期の実際 (4)

• ビデオのストリーミング再生

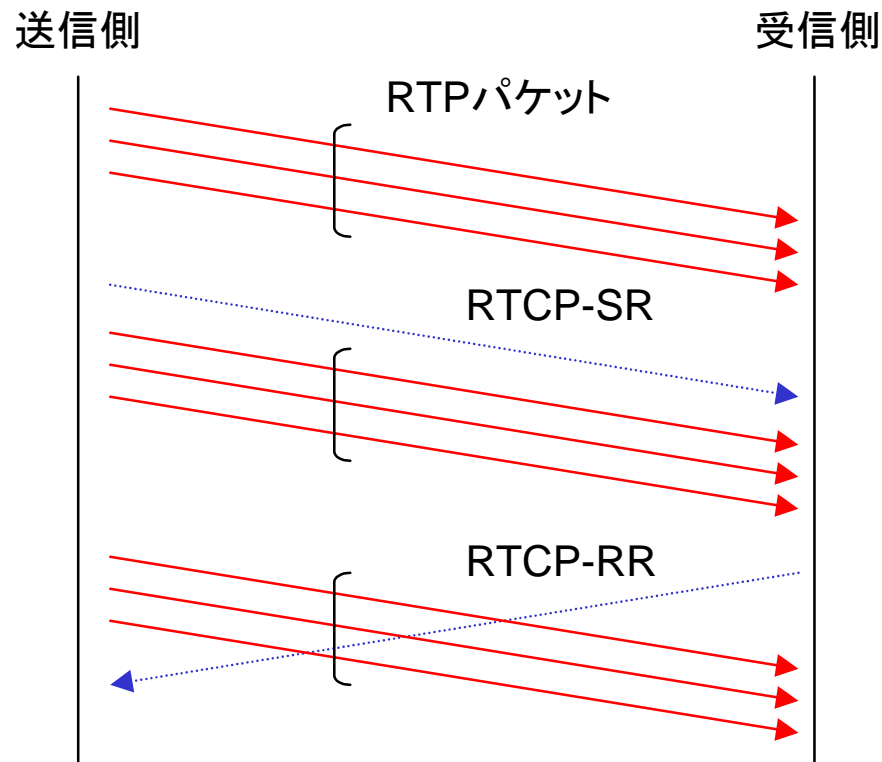
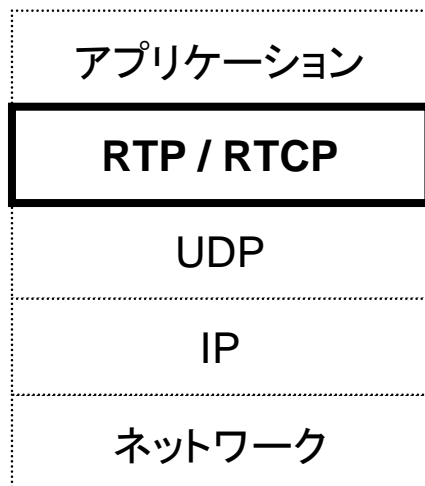


RTP/RTCP

RTP: Real-time Transport Protocol

RTCP: Real-time Transport Control Protocol

RTPとRTCP (1)



RTP: メディア情報の転送

RTCP: RTPの制御

RTPとRTCP (2)

RTP: メディア情報の転送

メディア内同期: RTP タイムスタンプ

パケット廃棄対策: シーケンス番号と

RTP ペイロードフォーマット

RTCP: RTPの制御

メディア間同期: NTP タイムスタンプ

フロー制御: 各種統計情報の交換

RTPヘッダ

v=2	P	X	CSRC カウント	M	パケットタイプ	シーケンスナンバ
タイムスタンプ						
SSRC 識別子						
CSRC 識別子 (list)						
(拡張フィールド)						
データ						

パケットタイプ:

転送メディアの符号化アルゴリズム

シーケンスナンバ:

パケットの順序逆転、廃棄の検出

タイムスタンプ:

同期再生 (メディア内同期)

Mビット:

フレーム境界の通知

SSRC:

送信者の識別

RTCP (Sender Report)

v=2	P	RC	PT=SR=200	パケット長
送信元 SSRC 識別子				

sender report

NTP タイムスタンプ (MSB)
NTP タイムスタンプ (LSB)
RTP タイムスタンプ
送出パケット数
送出バイト数

report block * n

SSRC 識別子 #n
廃棄パケット数
シーケンスナンバーの最大値
ジッタ遅延
SSRC #n の最新のSR受信時の NTP タイムスタンプ (LSR)
LSR から現在までの遅延 (DLSR)

RTCP (Receiver Report)

v=2	P	RC	PT=SR=201	パケット長
送信元 SSRC 識別子				

report block * n

SSRC 識別子 #n
累積廃棄パケット数
シーケンスナンバーの最大値
ジッタ遅延
SSRC #n の最新のSR受信時の NTP タイムスタンプ (LSR)
LSR から現在までの遅延 (DLSR)