

## リアルタイム通信用ソケットインタフェースの提案

4N-7

伊藤 嘉浩 前島 治 石倉 雅巳 浅見 徹

国際電信電話株式会社 研究所

### 1. はじめに

近年、インターネット電話やインターネットTV会議など、音声や画像などのデータをリアルタイムにTCP/IPネットワーク上で転送したいという要求が増加している。音声や画像などのデータをリアルタイムに伝送するためには、ネットワークに対して一定の帯域や遅延時間の保証などのサービス品質 (Quality of Service:QoS) が要求される。しかし、インターネット標準プロトコルであるTCP/IPネットワークにおいてサポートされるQoSはベストエフォート型のデータ転送のみである。そこで、TCP/IPネットワークでリアルタイムにデータを伝送するためのプロトコルとしてRTP(Realtime Transport Protocol) [1]およびRSVP (ReSerVation Protocol)[2]が提案され、IETFにおいて標準化が進められている。RTPはタイムスタンプ、シーケンス番号の付与、また、RTCP(RTP Control Protocol) による制御情報の交換などのサービスを提供しており、アプリケーションによる動的な制御が可能である。一方、RSVPは網資源予約型のプロトコルであり、データを転送する経路上のノード間で、必要な網資源を予約して通信を行うプロトコルである。これらのプロトコルには、TCP/UDPのsocketに相当するようなユーザインタフェースは標準化されていない。本報告では、リアルタイム通信のトランスポートプロトコルとしてRTPを想定し、リアルタイム通信として、socketインタフェースを拡張したSOCK-REALTIME型のsocketを提案する。

### 2. TCP/IPにおけるリアルタイム通信の利用形態

リアルタイム通信プロトコルの利用形態を以下に記す。

- RTP:ネットワークがQoSの保証を行わないため、ネットワークの状態に応じてアプリケーション側での制御が必要となる。RTPについてはsocketのような標準的なインタフェースが存在しないので、アプリケーションが独自にインタフェース部を実装する必要がある。
- RSVP:アプリケーションがRSVPを用いて網資源の予約を行い、QoSの保証を行う形態。RSVPに対するAPIとしてはRAPIがインターネットドラフトとして提案されている[3]。

- RTP/RSVP:RSVPにより、網資源の確保されたネットワーク上でRTPを用いて通信を行う形態。RSVPによりQoSの制御を、RTPによりメディア間同期などのアプリケーション側の制御を行う。

表 1: 利用形態

プロトコル	形態	インタフェース
RTP	ネットワーク指向	RTP独自
RSVP	アプリケーション指向	RAPI
RTP/RSVP	アプリケーション指向	RTP独自およびRAPI

### 3. 動作概要

#### 3.1 プロトコルスタック

一般的にRTPは多重化を行わないため、RTP/RTCPおよび複数のセッションの多重化については下位レイヤのサービスに依存する。従って、RTPはUDP上に実装される場合が多い。本提案ではRTPはUDPの上に実装されるものとする。全体の動作概念を図1に示す。アプリケーションはsocketインタフェースを通して

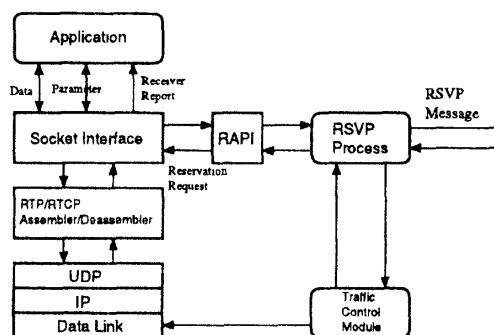


図 1: 全体概要

データ、各パラメータ、受信状態などの情報をやりとりする。リアルタイム型のsocketを用いる際、TCP/UDPのsocket生成時に必要なパラメータに加えて、最小限、表2に示すパラメータを指定する必要がある。Windows

表 2: 必要なパラメータ

送信者	受信者	網資源予約時
CNAME (セッションの 識別名)	RTCPの 送出間隔	flowspec (QoS パラメータ)

Sockets 2 API[4]においても、RFC1363に基づいたQoSのパラメータを指定することが可能であるが、CNAMEやRTCP送出間隔等のTCPやUDPが持たないRTP独自のパラメータを指定することはできない。そこで、本提案では、RTPおよびRSVPで統一のインタフェー

"A proposal of socket interface for real time communication" by Yoshihiro ITO, Osamu MAESHIMA, Masami ISHIKURA and Tohru ASAMI  
KDD R & D Laboratories

スを提供するために、UNIX/socket インタフェースの標準関数である、setsockopt コールを拡張して、QoS パラメータ、RTP パラメータを含めた必要なパラメータを指定するものとする。socket インタフェースはこれらの情報を RTP/RTCP パケットとして構成し、UDP にカプセル化してネットワークに送出する。アプリケーションから網資源の予約要求があった場合は、socket インタフェースは RAPI を通して RSVP Process に網資源の予約要求を送る (図 1)。網資源の確保状況は、RSVP Process から socket インタフェースに送られ、socket インタフェースは setsockopt コールの返り値によってアプリケーションに伝える。

3.2 送信側

RTP データ送信側のアプリケーションの動作とそれに伴う socket インタフェースの動作について記述する。この概念図を図 2 に示す。送信者は最初に、socket コ

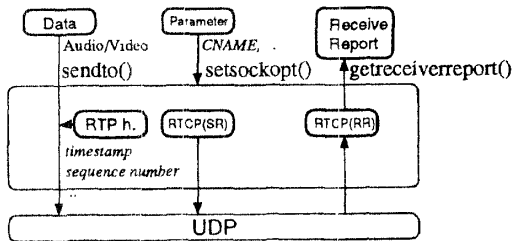
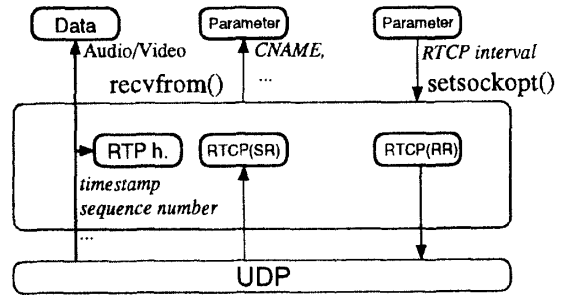


図 2: 送信側の動作概念

ールにより新規に socket を生成する。送信者は必要なパラメータを setsockopt コールを用いて設定する。アプリケーションから渡される音声や画像などのデータは、sendto コールを用いて送信先に送られる。データを受け取った socket インタフェースはデータに RTP パケットヘッダを付加し、更に UDP パケットにカプセル化してネットワークへ送出する。RTP ヘッダ内の timestamp, sequence number 等の各フィールドの値は socket インタフェースが自動的に値を入れる。送信者が受信した RTCP(Receiver Report:RR) パケット内に含まれるパケット損失数、ジッタ等の情報は、新規に定義する getreceiverreport コールを用いて、これらの値を持つ構造体型の変数に入れられる。

3.3 受信側

RTP データ受信側のアプリケーションの動作とそれに伴う socket インタフェースの動作について記述する。この動作概念を図 3 に示す。送信者と同様に、受信者は socket コールを用いて新規に socket を生成する。受信者からの機能拡張した recvfrom コールにより、socket インタフェースは受信した RTP パケットから RTP ヘッダ部を取り外し、受信者に渡す。送信者情報 s\_info および受信報告 rr\_info は構造体型の変数として同時に取り



```
recvfrom(int socket_fd, char *buffer, int length, int flag,
struct sockaddr *so_address, int sizeof(so_address),
struct senderinfo *s_info, struct receiverreport *rr_info)
struct senderinfo {
    int sequencenumber;
    int timestamp;
    ...
}
struct receiverreport {
    int ssrc;
    int sourcenumber;
    struct sourceinfo;
    ...
}
```

図 3: 受信側の動作概念

だされる。socket インタフェースは計算した受信状態を RTCP(RR) パケットとして定期的を送信する。マルチキャストの場合、送信者の数は 1 とは限らないが、送信者の送る RTCP(Sender Report:SR) パケットと RTP ヘッダ内に送信者の情報が記述されており、これを取り出すことにより、送信者数が動的に変化した場合でも対応可能である。また、必要なパラメータは setsockopt コールによって設定する。

3.4 RSVP への対応

ネットワークが RSVP に対応している場合、データ通信用のフローに対して、網資源の予約が必要な場合がある。RSVP は受信者指向のプロトコルであるので、網資源予約要求メッセージは受信者側から送出される。新規受信者がセッションに参加する場合、socket コールにより socket を生成した後、拡張した connect コールによって網資源の予約を行う。セッションの途中で必要な QoS が動的に変化する場合は ioctlsocket コールにより、QoS の変更を行う。

4. おわりに

本報告では、RTP 及び RSVP を用いたリアルタイム通信用アプリケーションインタフェースとして、SOCK-REALTIME 型の socket インタフェースを提案した。最後に日頃御指導頂く KDD 研究所 村上所長に感謝します。

参考文献

- [1] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC1889, January 1996.
- [2] Braden, R., Zhang, L., Berson, S., Herzog, S., Jamin, S., "Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification", Internet Draft, October 1996.
- [3] Braden, R., Hoffman, D., "RSVP Application Program Interface (RAPI)", Internet Draft, August 1996.
- [4] "Windows Sockets 2 Application Programming Interface". ftp://ftp.microsoft.com/bussys/winsock/winsock2/WSAPI22.DOC, May, 1996.