

ATM用確認型データ転送プロトコル SSCOP の

5C-7

実装に関する一考察

長谷川 輝之 長谷川 亨 加藤 聰彦 鈴木 健二

国際電信電話株式会社 研究所

1. はじめに

ATM LAN などの ATM ネットワークの普及に伴い、その上での高速なデータ転送が求められている。このためには、送達確認や誤り訂正などの機能を持つデータ転送プロトコルを、高いスループットで実現することが必要となる。現在 ITU-T では、ATM 用の確認型データ転送プロトコルとして、SSCOP(Service Specific Connection Oriented Protocol)^[1] の標準化を進めており、ATM 上のデータ通信において広く利用することを推奨している。そこで本稿では、SSCOP を既存のワークステーション上のソフトウェアとして実装するための検討課題と、それらに対する実装方法について述べる。

2. SSCOP 概要

SSCOP は AAL(ATM Adaptation Layer) タイプ 5 の上位にインタフェースする確認型データ転送プロトコルであり、その代表的機能は以下に示す通りである。

(1) ユーザデータの順序保存

送信側は、上位レイヤから渡されたユーザデータに順序番号を含む SSCOP トレイラを付加し、SD(Sequenced Data) PDU(Protocol Data Unit) を作成して、送出する。SD を受信した側は、順序番号の順にユーザデータを上位レイヤに渡す。

(2) SD の送達確認

SD の送達確認は、以下の方法により行なわれる。

- ・送信側から送られた POLL PDU に対して、受信側が送信する STAT(Solicited Status) PDU
- ・誤った順序で SD を受信した場合に、受信側が送信する USTAT(Unsolicited Status) PDU

(3) 選択再送による誤り訂正

受信側は、受信した SD の順序番号を検査し、その紛失を検出する。紛失した SD の順序番号は、STAT または USTAT によって送信側へ通知する。この際順序番号は、連続して紛失した SD のグループごとに指定する。送信側は、通知された紛失した SD のみを選択的に再送する。

(4) フロー制御

送信側は、STAT、USTAT 等で受信側から通知されるクレジットに基づいてフロー制御に用いるウィンドウ

を決定し、その範囲内で SD PDU を送出する。

3. 実装のための検討課題

SSCOP をワークステーション上のソフトウェアとして実装するためには、次のような検討課題が考えられる。

- (1) ライブラリ、デバイスドライバ、専用サーバなどから、最適な実装形態を決定する。
- (2) 明確なモジュール化を行ない、上位レイヤとの間のプログラムインタフェースを決定する。
- (3) 高いスループットを実現するために、上位レイヤやネットワークとの間での不要なデータコピーを避ける方式を検討する。
- (4) 選択再送機能などの SSCOP に固有の機能を効率的に実現するための、管理テーブルや処理の流れなどを検討する。

4. SSCOP ソフトウェアの実装方法

4.1 実装形態

SSCOP では、ATM レイヤでコネクションが識別されるため、受信したデータを対応するユーザプロセスに独自に分配する必要がないこと、並びにコンテキスト切替えなどのオーバーヘッドが少ないこと、さらには実装や移植が容易であることなどを考慮して SSCOP をライブラリとして実装することとした。図 1 に SSCOP を実装した通信プログラムの全体構成を示す。ここでは、SSCOP と上位レイヤプロトコルが、ライブラリとして実現され、メインルーチンを含む応用モジュールにリンクされる。また、これらを含むユーザプロセスは、ATM ボード用デバイスドライバを介して ATM ボードとの間でデータの送受信を行なう。

4.2 上位レイヤとのプログラムインタフェース

SSCOP ライブラリに対する送信要求は、上位レイヤプロトコルライブラリから行なわれるが、SSCOP PDU の受信は、応用モジュールにより検出されるため、以下のように送信と受信に対して異なるプログラムインタフェースを採用することとした。送信に対しては、勧告の規定する要求/応答プリミティブ(シグナル)に対応して、SSCOP PDU の作成・送出を行なうルーチンをそれぞれ用意する。受信に対しては、受信した SSCOP PDU の解析と対応する指示/確認プリミティブの通知を行なうルーチンを用意する。STAT や USTAT などのプリミティブを通知しない PDU の処理も、このルーチンで行なわれる。

“A Study on Implementation of SSCOP”

Teruyuki HASEGAWA, Toru HASEGAWA,
Toshihiko KATO and Kenji SUZUKI
KDD R & D Laboratories

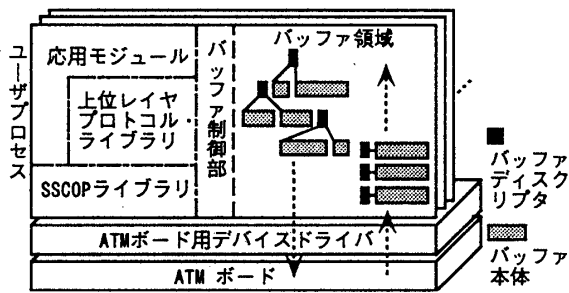


図 1: 全体構成

4.3 不要なデータコピーの回避

図1に示すように、SSCOP ライブラリは、応用モジュールおよび上位レイヤプロトコル・ライブラリとバッファ制御部を共有し、レイヤ間でユーザデータをコピーしないバッファ制御方式^[2]を使用する。このため、送信時は上位レイヤプロトコル・ライブラリから渡されたものをそのまま使用する。この場合は、図1に示すように、PDUは複数のバッファ本体に分割して格納される場合がある。また受信においては、SSCOP ライブラリが、確立された ATM コネクション毎に、クレジット分の SD を受信できるバッファを確保する。

さらに、ATM ボード用デバイスドライバにおけるコピーを避けるために、送信時は、分割された PDU データの配列をデバイスドライバに渡し、ドライバ内で連続領域にコピーすることなしに直接 ATM ボードに書き込み、また受信時には、前もって SSCOP ライブラリが確保したバッファ領域に ATM ボードから書き込む方式を用いる。

4.4 SSCOP 固有の機能の実現

4.4.1 送受信管理テーブル

選択再送やフロー制御を効率的に実現するために、以下に示す送信管理テーブルおよび受信管理テーブルをコネクション毎に用意する。

送信管理テーブルは、図2に示すように送達確認されていない SD のうち最小の順序番号 (VT(A))、次に送出する SD の順序番号 (VT(S))、ウィンドウの上限 (VT(MS))、SD 管理テーブル、送達確認用 SD グループリストなどを保持する。SD 管理テーブルは、未送信または送達確認がとれていない SD のバッファへのポインタや順序番号等を保持する。送達確認用 SD グループリストは、選択再送を効率良く行うために、送達確認されていない連続した SD のグループを管理する。図2では、順序番号が i から $i+1$ 、 $i+3$ から $i+4$ である SD のグループが本リストに登録されている。

受信管理テーブルは、次に上位レイヤに通知すべき SDU (Service Data Unit) に対応する順序番号 (VR(R))、次に受信が期待される SD の順序番号 (VR(H))、上位レイヤへの通知を待つ SDU を管理する SDU 管理テーブルなどを保持する。

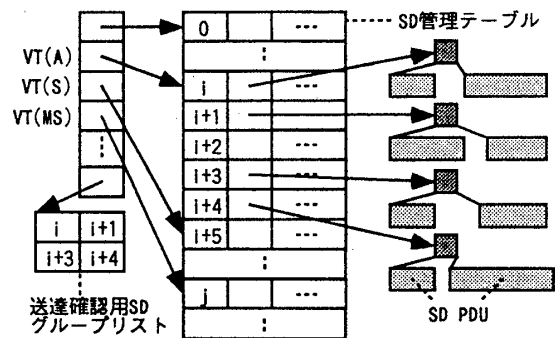


図 2: 送信管理テーブルの構成

4.4.2 送信処理

データ送信を要求されると SSCOP ライブラリは、受け取ったバッファ上の SDU にトレイラを付加して SD PDU を作成し、SD 管理テーブルに登録する。次に、作成した SD のシーケンス番号がウィンドウの上限を越えない場合は、SD を ATM ボード用デバイスドライバを介して ATM ボードに送出する。ウィンドウを更新する STAT または USTAT を受信すると、送信可能となった SD を送信する。

4.4.3 選択再送処理

STAT または USTAT を受信すると、再送を要求された SD のグループごとに、送達確認用 SD グループリストと SD 管理テーブルを参照し、再送済みでない SD のみを再送する。また STAT または USTAT が送達確認する SD に対しては、SD 管理テーブルに保持されたバッファを解放するとともに、送達確認用 SD グループリストを更新する。

4.4.4 データ受信

VR(R) に等しい順序番号をもつ SD を受信すると、まず、その SD に連続した SDU が既に受信済みであるかを確認する。受信済みでない場合はその SDU を単独で通知し、受信済みの場合は SDU をまとめて通知する。次に、上位レイヤに通知した SD 分の受信バッファを新たに確保する。一方、VR(R) と異なる順序番号をもつ SD を受信すると、上位に通知できないため受信管理テーブルに保持しておく。

5. おわりに

本稿では、ATM 用のデータ転送プロトコルである SSCOP を高速に処理するソフトウェアを実装する方法について、検討した結果を示した。最後に日頃御指導頂く KDD 研究所 浦野所長、眞家次長に感謝します。

参考文献

[1] ITU-T, "Service Specific Connection Oriented Protocol," Final Draft Recommendation Q.2110, December 1993.
 [2] 加藤, 井戸上, 鈴木, "OSI プロトコル実装のためのユーザデータをコピーしないバッファ制御方式," 情報マルチメディア通信と分散処理研究会, 62-13, September 1993.