

高速転送プロトコル (H T P) の検討

3H-5

本村 公太 坂口 勝章

N T T 通信網総合研究所

1. はじめに

ローカルエリアネットワーク(LAN)は光伝送技術の進展に伴って高速化し、GbpsオーダのLANが試作され始めている。それに対して上位でのOSIプロトコルの処理速度は2Mbps¹⁾以下であり、高速データ転送実現への妨げとなっている。この問題に対処するため、高速データ転送に適したプロトコルを新しく構築する試みがなされている²⁾³⁾。

本稿では、プロトコルの簡略化によって高速化をねらった新しいプロトコルとして、高速転送プロトコル(HTP: High-speed Transfer Protocol)を提案する。HTPはコネクション型のプロトコルであり、OSIの論理リンク(LLC)サブレイヤからトランスポートレイヤまでを一つのレイヤに縮退させることによって重複機能を排除するとともに、選択再送方式や最大SDU長に基づくバッファ予約等により高速化を図っている。

2. HTPの検討のステップ

HTPの検討のステップを表1に示す。

まず最初に、プロトコルの簡略化のために従来のプロトコルに対してレイヤの縮退を行なう。

次に、高速データ転送に適合するように個々の手順要素(フロー制御、送達確認等)の見直しを行なう。従来規定されていた手順要素はそれらのプロトコルが作成されたときの環境(高ビット誤り率、低伝送速度)を前提に考えられており、必ずしも高速データ転送に適したものではないからである。

さらに、プロトコル処理のハード(LSI)化を意識して、パラメータ長や並びの最適化を行なう。

表1 高速転送プロトコル(HTP)検討のステップ

ステップ	アプローチ	具体的手段
第1ステップ (簡略化)	レイヤ縮退 (OSIのLLC~トランスポートの1レイヤ化)	重複機能の排除(ヘッダ項目数、ヘッダ長削減) 不要機能の削除(同上) レイヤ間インタフェースの削減
第2ステップ (適合理化)	手順要素の見直し	選択再送方式(←全再送) 最大SDU長に基づくバッファ予約 転送方向別サービス品質
第3ステップ (最適化)	ヘッダフォーマットの見直し	LSI化時のバス幅に合せたパラメータ長 処理順序に適したパラメータの並び方

A Study for High-speed Transfer Protocol(HTP)

Kota MOTOMURA, Katsuaki SAKAGUCHI

NTT Telecommunication Networks Laboratories

3. プロトコルの簡略化の検討(第1ステップ)

(1) 縮退対象レイヤ

レイヤ縮退の対象は、LLCからトランスポートまでとした。その主な理由は次の通りである。

- ①重複機能がLLCからトランスポートまでに多い。
- ②エンドエンドでデータを転送する機能はトランスポートまでで満たされている。
- ③プリミティブの少ないレイヤ(使い易いレイヤ)で切るのが望ましい。

したがって、高速転送プロトコルのレイヤモデルは図1に示すようになる。

(2) 提供するサービス品質(QOS)

光ファイバを利用する高速LANではビット誤り率が低いため、画像系データなどに対してはビット誤りの検出手順を使用しなくとも要求品質を満足できる場合もある。したがって、ビット誤りの検出手順を使用するか否かをQOSとして指定可能とする。

また、高速LANを実時間制御に適用する場合を考えると、利用者がデータを受け取る前に次のデータがきた時、フローを止めるよりもデータを上書きした方が望ましい場合もある。したがって、データを上書きするか否かをQOSとして指定可能とする。

さらに、MACサービスに優先クラスがある場合に備えて、HTPにもQOSとして優先クラスを設け、MACでのクラスにマッピングする。

(3) 提供する高速転送サービスプリミティブ

提供するサービスプリミティブは、OSIトランスポート相当として、OSIとの整合性を考慮する。プリミティブの種類は以下の通りである。

H_CONNECT要求/指示/応答/確認

H_DATA要求/指示

H_DISCONNECT要求/指示

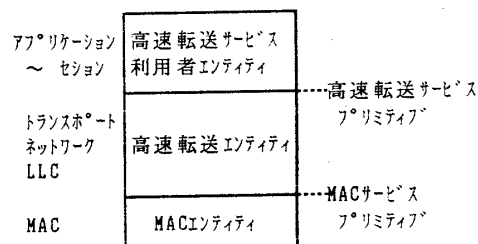


図1 高速転送プロトコルのレイヤモデル

(4) サポートする機能

OSIトランスポートをベースとして、サポートする機能について検討した。OSIとの比較を表2に示す。MACサービスであるユニットデータの送信要求と受信通知を直接使用するので、下位コネクションを前提とした機能は不要である。また、連結と分離の手順は処理を複雑化するので省略した。タイマによる再送手順は、タイマのセット/リセットによるオーバーヘッドが大きくなるので、受信側から再送要求を送る方式とした。

4. 高速転送に適した手順の検討 (第2ステップ)

(1) 選択再送方式

OSIトランスポートでは、実効上、再送要求されたDT以降を連続的に再送する全再送方式になっている。このため、一旦再送が発生すると転送効率が急に低下する。そこで、HTTPでは、制御が若干複雑になるが効率のよい選択再送方式を採用する。選択再送を実現した場合に受信側でのバッファ管理を効率化するため、分割と組立ての手順で使用するパラメータとして終端識別子(EOT)に加えてSDU識別子、セグメントオフセットを使用する。

(2) 最大SDU長に基づくバッファ予約

高速通信のためには、データ転送中、特に一つのSDU(Service Data Unit)分の転送中にはフローの停止が起こらないことが望ましい。そこでHTTPでは、コネクション確立時のプリミティブおよびPDUに最大SDU長のパラメータを設け、コネクション確立時にそれぞれの転送方向で使用する最大のSDU長を利用者からの要求に基づいてHTTPレベルで折衝し、バッファの確保を行なう。これによってSDU転送中のフローの停止を回避できる。

(3) 転送方向別QOS

静止画DBアクセスの場合には、コマンドを送る端末→センタ方向と、静止画データを送るセンタ→端末方向では要求されるQOSが異なる。このような要求条件に応えるため、転送方向別のQOSを指定する機能をコネクション確立時の機能に加える。

5. 高速化の見通し

HTTPの採用によって、OSIプロトコルに比べてどの程度処理が高速化されるかについての目安を得るために、ヘッダ項目数とヘッダ長を比較してみる。OSIプロトコルとしては、LLCはタイプ1、ネットワークはコネクションレス型、トランスポートはクラス4であるとする。スルーputを比較するために、データの転送時のみ対象とする。

表3.1および表3.2にOSIプロトコルおよびHTTPのデータ転送時ヘッダ項目を示す。プロトコル処理に要する時間がヘッダ項目数にある程度比例すると考えると、ヘッダ項目数が7/23になっていることから3倍程度の高速化は期待できる。また、ヘッダ長は1/10になり、転送効率が改善されている。さらに、OSIプロトコルのインプリメント法に依存する問題であるが、レイヤ間インタフェースのオーバーヘッドの減少による高速化の効果がこれに加わるものと期待できる。

6. おわりに

筆者らが検討を進めているHTTPに関する検討結果について述べた。今後は定量的な評価を行なう予定である。

表2 HTTPとOSIトランスポートクラス4の比較

プロトコルの機能	OSI 7514	HTTP
下位コネクションへの割当て	◎	×
PDUの転送	◎	◎
分割と組立て	◎	◎
連結と分離	◎	×
コネクションの確立	◎	◎
コネクションの確立拒否	◎	◎
明示的正常解放	◎	◎
異常解放	×	◎
自己コネクションとPDUの関連付け	◎	◎
DTのPDUの番号付け	◎	◎
優先データ転送	◎	×
PDU到達確認までの保持	◎	◎
再同期	×	◎
多重化と逆多重化	◎	×
フロー制御	◎	◎
チェックサム	○	○
レファレンスの凍結	◎	◎
タイムアウト時の再送	◎	×
再順序付け	◎	◎
下位コネクションの異常解放監視制御	◎	×
プロトコル誤りの扱い	◎	◎
分流と合流	◎	×

◎:必ず含まれる手順
○:実装することは必須だが、使用については折衝(指定)可能な手順
×:適用されない手順

表3.1 OSIプロトコルのデータ転送時ヘッダ項目

	項目名	長さ(オクテット)
LLC層 タイプ1	DSAPアドレス	1
	SSAPアドレス	1
	制御部	1
ネットワー ク層 コネクシ ョンレス 型	ネットワーク層7*トコル識別子	1
	ヘッダ長	1
	ネットワーク層7*拡張7*トコル識別子	1
	ライフタイム	1
	タイプコード	1
	セグメント長	2
	チェックサム	2
	宛先アドレス長	1
	宛先アドレス	15
	発信元アドレス長	1
	発信元アドレス	15
トランス ポート層 クラス4	データ単位識別子	2
	セグメントオフセット	2
	全長	2
	ヘッダ長	1
	TPDUコード	1
宛先レファレンス	2	
EOTと		
TPDU-NR		
チェックサム		
	合せて4	2

表3.2 HTTPのデータ転送時ヘッダ項目

項目名	長さ(オクテット)
HPDUコード	1
宛先レファレンス	1
EOTと	
シーケンス番号	合せて2
SDU識別子	2
セグメントオフセット	2
チェックサム	2

参考文献

- 1) "OSI throughput performance: Breakthrough or bottleneck", Data Communications May 1987
- 2) 谷他:「マルチメディアLANにおける高速通信プロトコル」, 信学技報IN86-129
- 3) G.Chesson: "THE PROTOCOL ENGINE PROJECT", UNIX REVIEW Sep.1987