

## OSI FTAM、ACSE、プレゼンテーション・プロトコルの実装

小花貞夫 加藤聰彦 鈴木健二

国際電信電話株式会社

筆者らは、OSI(開放型システム間相互接続)のFTAM(ファイル転送、アクセスと管理)、ACSE(アソシエーション制御サービス要素)およびプレゼンテーション層のプロトコルのソフトウェアを実装した。本ソフトウェアではOSI上位層、特に今後複雑化する応用層の実装に対して柔軟に対応できる汎用的なソフトウェア構成ならびにFTAMと実ファイル・システムとの実用的な対応付けを実現している。本稿では、まず実装の基本的な考え方を議論し、次いでこの結果に基づいて実装したソフトウェア構成および仕組みについて詳細に報告している。さらに実証実験等を通して、ソフトウェアの機能、効率等について考察している。

## Implementation of OSI FTAM, ACSE and Presentation Protocols

Sadao OBANA Toshihiko KATO Kenji SUZUKI

KDD R & D Labs. 2-1-23, Nakameguro, Meguro-ku, Tokyo, 153

Softwares for FTAM ( File Transfer, Access and Management ), ACSE ( Association Control Service Element ) and Presentation Protocols for OSI ( Open Systems Interconnection ) have been implemented. These softwares have an architectural flexibility to adapt wide variety of applications, and achieves sufficient mapping between OSI protocols and local system environment, such as mapping between FTAM and local file management system. In this paper, design strategies for the softwares are firstly discussed. Then detailed specifications of the softwares are discussed. Furthermore, facilities and throughput of these softwares are evaluated.

## 1. はじめに

現在ISO、CCITT等では、異機種のコピュータ・システムあるいは端末間で通信を可能とするためのOSI(開放型システム間相互接続)の標準化が精力的に進められている。既にトランスポート層、セッション層のサービス定義とプロトコル仕様がIS(国際標準)となり、さらに現在プレゼンテーション層ならびに应用層におけるACSE(アソシエーション制御サービス要素)や、FTAM(ファイル転送、アクセスと管理)<sup>[1]</sup>などの特定の应用サービス要素のサービス定義およびプロトコル仕様についても、ほぼISの状態となり確定してきている。このような背景の中で、筆者らは、通信網内において各種のプロトコル変換やメッセージ交換等高度な通信サービスを提供する通信処理ノードの構築を行っており、この通信処理ノードに各種の通信システムを收容するため、これまでにセッション層以下のプロトコルをVAX11/780上<sup>[2]</sup>に実装している<sup>[2]</sup>。今回さらにプレゼンテーション層、ACSEおよびFTAMのプロトコルを実装<sup>[3][4]</sup>したので、これについて報告する。

## 2.1 プレゼンテーション層

OSIでは应用プロセス間で通信する際に扱われる情報を、抽象構文(Abstract Syntax)と転送構文(Transfer Syntax)という2つの側面からとらえている。抽象構文はデータの意味・解釈を与えるための構文で、应用層で扱われる。一方転送構文は転送上の表現形式(符号化形式)を与えるもので、プレゼンテーション層で扱われる。これら抽象構文と転送構文を表現する手法として、ASN.1(抽象構文表記法1)とその基本符号化規則<sup>[5]</sup>が別標準として規定されている。プレゼンテーション層では、抽象構文と転送構文の関連付け(Pコンテキストと呼ばれる)を制御する機能と抽象構文/転送構文間の変換を行う機能を持っている。つまり应用層からの要求に応じて、抽象構文に対応する適切な転送構文を決定し、これに従ってデータを転送する。Pコンテキストは、プレゼンテーション・コネクション確立(P-CONNECT)時、あるいはコンテキスト変更(P-ALTER-CONEXT)時に、相手プレゼンテーション・エンティティとの間で折衝し設定される。

プレゼンテーション層は、この他应用層がセッション層のアクティビティ、トークン、同期などの会話制御機能を直接使用できるようなインタフェース(P-

## 2. プレゼンテーション層、ACSE、FTAMの概要

まずプレゼンテーション層、ACSE、FTAMの機能の概要についてのべる。(図1参照)

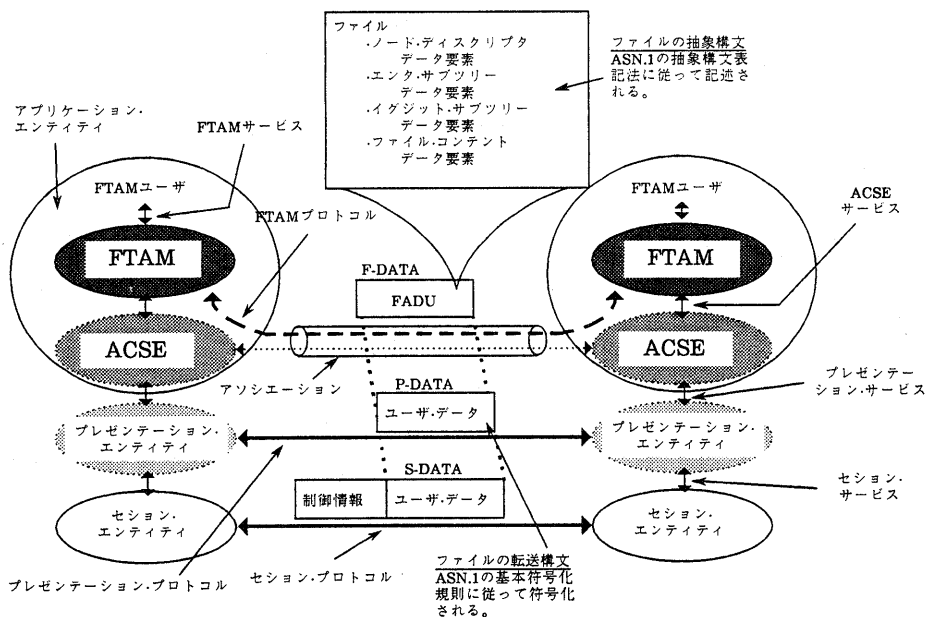


図1 プレゼンテーション、ACSEおよびFTAMの関係

ACTIVITY-START、P-TOKEN-GIVE、P-SYNC-MINORなど)機能も提供している。

## 2.2 ACSE

応用層は、抽象構文という形でデータを授受して特定の通信業務を遂行する。応用層の機能を遂行する実体であるAE(応用エンティティ)は、いくつかのASE(応用サービス要素)から構成される。ACSEは応用層のひとつのASEであり、各アプリケーションに共通な機能であるアプリケーション・アソシエーション(以下単にアソシエーションと呼ぶ)を制御する。アソシエーションは、双方の応用エンティティ間で情報を授受するための論理的な関連であり、ACSEにより確立(A-ASSOCIATE)、解放(A-RELEASE、A-ABORT、A-P-ABORT)される。アソシエーションを確立する際には、AEタイトルにより相手AEを指定し、またAコンテキストと呼ばれる応用層の通信環境(使用するASEやそれらの使用規則、オプションなどを折衝し設定する。

## 2.3 FTAM

FTAMも、応用層のひとつのASEで、OSI環境下で遠隔にある異種のファイル・システム(仮想ファイル・ストアと呼ぶ)間で共通の手順によりファイル全体あるいは一部を転送したり、新たなファイルを生成したり、削除したりするためのプロトコルを規定している。

FTAMでは、仮想ファイル・ストアを提供する側(レスポング)とそれを利用する側(イニシエータ)とが明確に区別されており、その動作は双方で不平衡となっている。

FTAMでは、異なる計算機のさまざまな形態のファイルを扱うために、次のような仮想ファイルを定義している。

(1)仮想ファイルは現実のファイル編成を、一般化された木構造により表現している。木の各ノードにはその名前とファイルの内容であるDU(データ単位)が付加され、ファイルの転送およびアクセスは木または部分木単位(このアクセス単位をFADUと呼ぶ)に行われる。さらに現実のファイル編成に対応させるため、木の構造と許されるDUへの操作等を制限した、コンストレイント・セットが定義されており、無構造(ルートノードのみ)、シーケンシャル・フラット(木の深さが1でノード名は無し)、オーダー・フラット(深さが1でノード名あり)等が定められている。

(2)仮想ファイルには、ファイル名、ファイルに許される操作、操作を許可するユーザ名、DUの抽象構文およびコンストレイント・セットを示すコンテ

ンツ・タイプ、生成ユーザ名、生成日時などの属性を持つ。

またFTAMの手順では、まずFTAM通信の開始を宣言し(F-INITIALIZE)、次にファイルを選択または生成する(F-SELECT/F-CREATE)。選択している間に、ファイルを開く(F-OPEN)、データを送受信(F-READ/F-WRITE、F-DATAなど)し、また属性の参照/変更を行うことができる(F-READ-ATTRIB/F-CHANGE-ATTRIB)。

## 3. 実装の方針

今回のプレゼンテーション、ACSE、FTAMプロトコルの実装においては、以下のような方針をたてた。

(1)既にVAX11/780 VMS上に実装したセッションに加えて、新たにプレゼンテーション、ACSEおよびFTAMを実装し、異機種システムとの相互接続が可能なようにOSIに完全に準拠した形で実現する。

(2)今後のFTAM以外のASE(例えばVT、JTM、RDASP、ディレクトリ・システムなど)の効率的な実装や複数ASEの柔軟な組み合わせの実現を考慮したソフトウェア構成とする。

(3)基本的にプレゼンテーション、ACSEおよびFTAMのすべてのプロトコル要素を実現するものとする。ただしFTAMの仮想ファイルについては、VMSが提供するファイル管理機能と対応をとることによって実現する。

(4)FTAMのイニシエータ(ファイル・ユーザ側)とレスポング(ファイル・ストア側)の両機能を実現する。

(5)VMSのローカル・ユーザが、遠隔からFTAMを使ってアクセスできるようにする。

(6)FTAMのイニシエータ側のユーザ・インタフェースとしては、VMSのコマンド・レベルと応用プログラムからの関数呼び出しレベルをサポートする。

(7)ソフトウェアの移植性を考慮し、C言語によりプログラミングする。

## 4. 基本検討

プレゼンテーション、ACSEおよびFTAMを実装するにあたり、次の幾つかの項目についてまず検討する必要がある。

### 4.1 OSI上位層プロトコルの実装の考え方

#### (1)ソフトウェア構成

OSIの上位層プロトコルのためのソフトウェアの構成方法としては、①プレゼンテーション層以上の機能すべてを一つのプロセスとして構成する、②プレゼンテーション層、ACSE、FTAMを独立のプロ

セスとして構成する、が考えられる。本来プレゼンテーション層と応用層機能は極めて密接な関係にあるため①の方法で実現するのが自然である。しかしながら今後FTAM以外のASEを追加実装し、プレゼンテーション層やACSEの上でそれらを切り替え可能としようと考えた場合、②の方法が有利である。従って今回の実装では、プレゼンテーション層、ACSE、FTAMを独立のプロセスとして構成することとした。

## (2) 抽象構文 / 転送構文変換

2.1節で述べたように抽象構文および転送構文は、それぞれ応用層およびプレゼンテーション層で扱われる。抽象構文は、実プロセスが扱う実システム固有のデータ構造をOSI環境で共通に理解できるデータ構造としてモデル化したものである。従ってプレゼンテーション層における抽象構文と転送構文の変換においては、直接実システムのデータ構造(例えばファイル構造や操作コマンド)に対応付けすることが考えられる。つまりプレゼンテーション層プロセスは応用層(ACSEおよびFTAM)プロセスと実システムのデータ構造で授受することになる。例えばFTAMの場合、ファイル・データを直接実ファイル・システムの書き込みコマンドの引数に対応付ける。しかしながらこのような方法を採用すると、効率的な構文解析が行えるものの、①ACSEやFTAMでプロトコル・エラーが生じた場合に不必要な構文変換を行ってしまう、②プログラムの他システムへの移植性が低下する、③応用プロセス(あるいはASE)によるFTAMの使用は画一的ではない(ある応用では直接ローカル・ファイル・システムに書き込み、またある応用では応用プログラムが直接利用するなど)、④異なるプロセス間で構造化された実システムのデータ構造を授受させるのは難しい、などの不都合があるため、今回の実装ではソフトウェアの汎用性を重視し、プロセス間のデータ授受では、抽象構文に対する表現としてASN.1の基本符号化規則による転送構文を採用することとした。

## 4.2 FTAMと実ファイルシステムとの対応

FTAMの仮想ファイル・ストアを実現するにあたり、①新たに仮想ファイル・ストアに相当するファイル管理システムを作成する、②既に実システムが提供するファイル管理機能を利用する、の方法が考えられる。今回の実装では、ローカル環境とOSI環境におけるファイルの共通利用の観点から②の方法を採用することとした。ここでは、以下の仮想ファイル・ストアと実ファイル・システムとの対応付けが重要となる。

①VMSのファイル操作機能は、RMS(Record Management Service)としてまとめられており、ファイルのオープン、レコードの読み書き、ファイル属性の参照/設定等はRMSのシステム・サービスを発行することにより達成される。そこでFTAMのサービス・プリミティブとRMSのシステム・サービスとの対応をとる。

②VMSのローカルなファイルの編成と仮想ファイルの木構造の対応付けは、他の異機種システムとの通信で共通に解釈できるものを対象とする。

③ファイル属性の対応付けは、基本的にはVMSで管理している属性についてのみ行う。

④ユーザとそのパスワードや、ファイルに対するアクセス権等に関しては、FTAM独自の管理は行わず、VMSの管理する情報を使用する。

## 5. 実装結果

### 5.1 ソフトウェア構成

(1)ソフトウェアは図2に示すようなソフトウェア・モジュール群から構成され、プレゼンテーション、ACSE、FTAMは独立のプロセスとして構成され、各プロセスは上り/下りのメール・ボックスの対によるキュー・インタフェースで結合される。これらのキュー・インタフェース上のコマンド/レスポンスは各層あるいはASE間のサービス・プリミティブをフォーマット化したものを採用した。

(2)FTAMのイニシエータ側とレスポンド側とでは、一方がファイル提供者側で、もう一方がその利用者側というFTAM本来の使用形態から、以下のような不平衡な作りとした。

①イニシエータ側としては、今後の応用層構造の拡充(他のASEがFTAMを制御するなど)を考慮し、FTAMプロセスとは独立なユーザ・プロセスとして、ユーザ・プログラムとユーザ代行プログラムを用意した。ユーザ・プログラムとユーザ代行プログラムはひとつのプロセスとして構成され、FTAMプロセスの上で複数のユーザ・プロセス(異なる種類のものでも良い)が動作できる。FTAM/ユーザ代行プログラム間においてもメール・ボックス・インタフェース経由で、FTAMのサービス・プリミティブをフォーマット化したものでインタフェースさせた。

ユーザ代行プログラムは、ユーザ・プログラム(ユーザが作成する応用プログラム)を作成し易いように、マクロ的なファイル操作(例えばファイル全体をコピーするなど)を構造化データを用いた関数

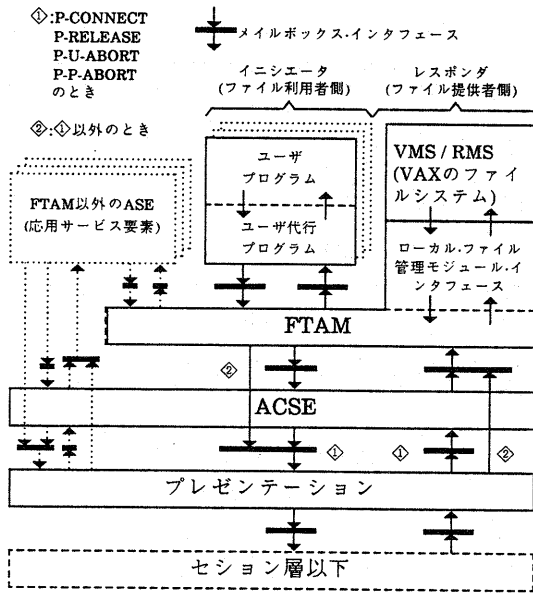


図2 FTAM実装におけるソフトウェア構成

呼び出しの形<sup>[6]</sup>で、ユーザプログラムに対して提供できるようにした。

②またレスポнда側としては、VMSの提供するRMS、FTAMおよびこれらの対応をとるローカル・ファイル管理モジュール・インタフェース・プログラムをひとつのプロセスとして構成した。

(3)プレゼンテーション・プロセスとACSE/FTAMプロセスとのインタフェースでは、ユーザデータの抽象構文に対する内部表現としてASN.1の基本符号化規則による転送構文を採用した。

(4)FTAMプロセスは複数の異なる種類のユーザ(異なるユーザ代行プログラムやFTAM以外のASE)をサポートするために、上位に提供するメール・ボックス名とFCEP(FTAM Connection End Point)番号を用いて個々のFTAMコネクションを管理している。FCEP番号は、ユーザ代行プログラムあるいは他のASEプロセスの中で、ユニークに番号付けされ管理される。またアソシエーションの確立/解放を伴う場合以外(例えばF-DATAが直接P-DATAにマッピングされるなど)は、ACSEを介さずにFTAMとプレゼンテーションが直接データ授受を行えるようにする必要があり、このため[FTAM/ACSE間の上りメールボックス名+AEP(Association End Point)番号]およびPCEP(Presentation Connection End Point)番号がコネクション確立時に、要求/確認あるいは指示/応答を使ってFTAMとプレゼンテーション間で通知し合うようにした。ここでAEP番号はACSEが

アソシエーションを管理、識別するためのもので、PCEP番号はプレゼンテーション・プロセスがプレゼンテーション・コネクションを管理、識別するためのものである。FCEP番号とAEP番号の概念はOSI標準にはなく、今回の実装のために新たに導入したものである。

## 5.2 FTAMとローカル・ファイル・システムとの対応

レスポнда側のローカル・ファイル管理モジュール・インタフェースでは以下のようにFTAMとVMSのファイル・システムであるRMSとの間で機能の対応付けを行っている。

### (1)ファイル操作の対応付け

RMSでは、次のようなファイル操作のシステム・サービスが用意されている。ファイルの入出力のためには、まずSYS\$OPENで、ファイルに対する操作や専有/共有の区別を指定してファイルをオープンし、次にSYS\$CONNECTでレコード入出力の指定を行い、実際の入出力はSYS\$PUTまたはSYS\$GETで行う。またファイルの属性はSYS\$DISPLAYで参照する。ファイルの書き込み手順における、FTAMとRMSのシステムサービスの対応付けを図3に示す。

なお後述する無構造ファイルの場合には、RMSのSYS\$OPEN、SYS\$CONNECT、SYS\$PUT(あるいはSYS\$GET)を使用せず、代わりにC言語のfopen、fputc(あるいはfgetc)を用いる。

また、F-OPEN要求のパラメータはSYS\$CONNECTには対応せず、特にF-OPEN要求のコンカレンシ制御はF-SELECT要求よりも制限の強い指定が可能であるのに対してSYS\$CONNECTではコンカレンシ制御を指定できないため、その場合はファイルを一旦クローズし、新たなコンカレンシ制御に従って再度オープンするようにした。

### (2)ファイル編成の対応付け

VMSのファイルに対して、順/相対/索引の3つのファイル編成と、固定長/可変長/ストリーム等のレコード・フォーマット、CR(キャリッジ・リターン)制御/印字書式制御等のレコード属性の組み合わせにより、表1に示すような対応付けを検討した。

ここで順編成でストリームのファイルは無構造としたのは、この形式のファイルがレコードを区切るための特別な構造を持たないためである。現在までの実装では、無構造とシーケンシャル・フラットに対応するVMSの順編成のファイルのみを対象としている。

表1 VMSのファイル編成と仮想ファイルの対応

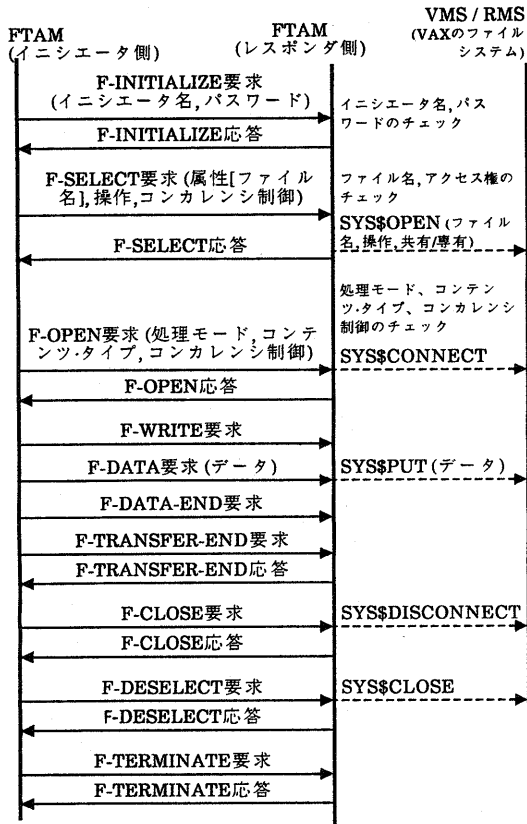


図3 FTAMとRMSのファイル操作の対応

ただしシーケンシャル・フラットはいくつかの異なるVMSファイルに対応するため、ファイル名のファイル・タイプ(例えばtest.OBJの"OBJ")によってどの順編成ファイルに対応付けるかを決定している。

(3) ファイルの属性の対応付け

仮想ファイルの必須の属性であるファイル名とコンテンツ・タイプについては次の様に対応付けた。VMSでは階層的なディレクトリ構造をとっており、またバージョン管理も行っているため、ファイル名の一部にディレクトリやバージョン番号の指定も許した。またコンテンツ・タイプの中のDUの対象構文(テキストか、バイナリか)については、VMSでは対応する情報を管理していないため、ファイル名のファイル・タイプからテキスト・ファイルかバイナリ・ファイルかを決定することとした。その他のオプションの属性については、最後にファ

VMSのファイル	仮想ファイル
編成：順編成 レコードフォーマット：ストリーム レコード属性：CR制御 (例 ユーザがC言語の標準入出力で作成したファイル)	無構造
編成：順編成 レコードフォーマット：可変長 レコード属性：CR制御 (例 エディタで作成したファイル)	シーケンシャル・フラット
編成：順編成 レコードフォーマット：可変長 レコード属性：なし (例 オブジェクト・ファイル)	シーケンシャル・フラット
編成：順編成 レコードフォーマット：固定長 レコード属性：なし (例 EXECファイル)	シーケンシャル・フラット
編成：相対編成 レコードフォーマット：固定長/可変長	ユニーク名付オーダード・フラット
編成：索引編成(主キーのみ/重複可)	オーダード・フラット
編成：索引編成(主キーのみ/重複不可)	ユニーク名付オーダード・フラット

イルを読んだユーザ名やその日時などVMSのファイル属性にないものは対応付けなかった。

(4) アクセス制御の実現

F-INITIALIZEでのイニシエータ名とパスワードによるイニシエータの正当性の確認に対しては、VMSでパスワードの検査機能がユーザに提供されていないため、VMSとは別にパスワードを管理する必要がある。また仮想ファイルに許される操作はVMSのファイルのアクセス権に対応させるが、VMSではユーザの識別及びファイルのアクセス制御にUICという識別子を用いている。このためイニシエータ名/パスワード/UICの対応情報を別途管理することとした。

5.3 FTAM関連プロセス仕様

- (1)今回実装の対象としたFTAM機能を表2に示す。
- (2)FTAMの各サービスは表3に示すように下位のACSE、プレゼンテーション、セッションのサービスにマッピングしている<sup>[7]</sup>。
- (3)ファイル構造としては、無構造ファイルとシーケンシャル・フラット・ファイルをサポートした。無構造ファイルの場合には、一定長毎に分割してF-DATAとして送出すし、またシーケンシャル・フラット・ファイルの場合には、各レコードをF-DATAとして送出する。この際、複数のF-DATAをひとつのP-DATAにマッピング(コンカチネーション)することができる。
- (4)F-DATAなどのFTAMのプロトコルをACSEやプレゼンテーション・サービスのユーザ・データとして

表2. 実装機能

ファイル・ストア	ドキュメント・タイプ 無構造テキスト、無構造バイナリ、 シーケンシャル・テキスト、シーケンシャル・バイナリ
	アクセス・コンテキスト HA(Hierarchical All Data Units)、 FA(Flat All Data Units)、 UA(Unstructured All Data Units)、 US(Unstructured Single Data Unit)
サービス	Constraint Sets 無構造、シーケンシャル・フラット
	サービスレベル User Correctable(UCFS)、Reliable サービスクラス ファイル転送、ファイル・アクセス、 ファイル管理、転送+管理、制限無 機能単位 Kernel、Read、Write、File Access、 Limited File Management、Enhanced File Management、 Grouping、Recovery、Restart Data Transfer
プロトコル	プロトコル・タイプ Basic File Protocol、 Basic Bulk Data Transfer Protocol、 Error Recovery Protocol

表3 FTAM、ACSE、プレゼンテーション、セッションサービスの対応

FTAM	ACSE	プレゼンテーション	セッション
F-INITIALIZE	A-ASSOCIATE	P-CONNECT	S-CONNECT
F-TERMINATE	A-RELEASE	P-RELEASE	S-RELEASE
F-P-ABORT 注1)	A-ABORT or A-P-ABORT 注1)	P-U-ABORT or P-P-ABORT	S-U-ABORT or S-P-ABORT
F-U-ABORT 注1)	A-ABORT 注1)	P-U-ABORT	S-U-ABORT
F-SELECT	→	P-DATA	S-DATA
F-DESELECT	→	P-DATA	S-DATA
F-CREATE	→	P-DATA	S-DATA
F-DELETE	→	P-DATA	S-DATA
F-READ-ATTRIB	→	P-DATA	S-DATA
F-CHANGE-ATTRIB	→	P-DATA	S-DATA
F-OPEN	→	P-DATA	S-DATA
F-CLOSE	→	P-DATA	S-DATA
F-BEGIN-GROUP	→	P-DATA	S-DATA
F-END-GROUP	→	P-DATA	S-DATA
F-RECOVER	→	P-DATA	S-DATA
F-LOCATE	→	P-DATA	S-DATA
F-ERASE	→	P-DATA	S-DATA
F-READ	→	P-DATA	S-DATA
F-WRITE	→	P-DATA	S-DATA
F-DATA 注2)	→	P-DATA	S-DATA
F-DATA-END	→	P-DATA	S-DATA
F-TRANSFER-END	→	P-DATA	S-DATA
F-CANCEL	→	P-RESYNCH (abandon) or P-DATA 注3)	S-RESYNCH (abandon) or S-DATA
F-CHECK 注2)	→	P-SYNC-MINOR	S-SYNC-MINOR
F-RESTART	→	P-RESYNCH (restart)	S-RESYNCH (restart)

注1) 今回F-PABORT、F-U-ABORT、A-ABORT、A-P-ABORTはPDUを実装せず、プレゼンテーションの異常終了を直接対応付けた。

注2) PDUはない。

注3) FTAMのRecovery FUとプレゼンテーションのResynchronize FUを選択した場合にはP-RESYNCH(abandon)に、それ以外はP-DATAにマッピングされる。

下位のプロセスに渡す際には、Pコンテキスト識別子を含むASN.1のEXTERNAL型のデータとして符号化している。

(5)今回作成したユーザ・プロセス(ユーザ・プログラムとユーザ代行プログラム)では、自システムのファイルと遠隔システムとの間でファイルの複写を行うための機能をVMSのコマンド・レベル(>COPY ファイル名-1 TO ファイル名-2)で呼び出せるようにしている。また仮想ファイルのファイル属性の参照(>READATTRIB ファイル名)、ファイルの削除(>DELETE ファイル名)などの機能も提供している。さらに相手ファイル・システムがVMSの場合を想定し、仮想ファイル・ストアのディレクトリ参照(>DIR ディレクトリ名)も可能とした。ここではディレクトリ・ファイルもひとつのシーケンシャル・フラット・ファイルとして扱うことで実現している。

(6)相手AEを指定するためにAEタイトルを使用しているが、AEタイトルはASN.1のオブジェクト識別子型のデータとして表現した。今回AEタイトルとPSAPの対応、PSAP/SSAPの対応はそれぞれ別ファイルとして対応表を準備し、それを使用してアドレス対応をとった。

#### 5.4 ACSE、プレゼンテーション・プロセス仕様

(1)ACSEプロセスは、標準で規定されるアソシエーション制御機能をサポートしている。またプレゼンテーション・プロセスは、コネクション制御、プレゼンテーション・コンテキスト制御(現在P-ALTER-CONTEXTは実装していない)およびセッション・パススルー機能などの諸機能をサポートしている。

(2)4.1節(2)で述べたように、転送構文をそのまま抽象構文としても使用しているため、プレゼンテーションでは抽象構文/転送構文の変換は行わずに、転送構文の構文チェックのみを行っている。

(3)現行標準におけるセッションの異常終了サービス(S-U-ABORT)のユーザ・データ長が9オクテットと限定されており、上位層の異常終了(P-U-ABORT、A-ABORT、F-U-ABORTなど)のプロトコル・データ単位(PDU)を収容することができないという問題があるため、今回異常終了についてはプレゼンテーションのみPDUを実装し、ACSEとFTAMでは、プレゼンテーションの異常終了を直接対応させた。

(4)Pコンテキストについては、FTAMに関するものはFTAMプロセスで、またACSEに関するものはACSEプロセスで管理している。

## 6. 実証実験の結果および考察

### 6.1 実証実験の結果

実装したプレゼンテーション、ACSE、FTAMのソフトウェアを実証するために、X.25公衆パケット交換網(VENUS-P)を経由した接続試験を行った。ここでは、pVAXII VMS上にもこれらのソフトウェアを移植し、VAX-11/780とpVAXIIの間でファイル転送を行った。この際の通信環境として、ネットワーク以下の層ではX.25、トランスポート層ではプロトコル・クラス0を、またセッション層ではBCSを使用した。

この結果、転送効率として物理回線速度が9.6Kbps(スループット・クラスが4.8Kbps)のX.25回線の場合で約4Kbpsの実効スループットを確認できた。

### 6.2 考察

今回のプレゼンテーション層、ACSEおよびFTAMの実装を通して、次のことが判明した。

- (1)極めて汎用性のあるソフトウェア構成とすることができた。これによりFTAM以外のASEプログラムの追加や上位層の構造、特に複雑な応用層構造に対する対応を容易とした。(現在ISOでは応用層構造で、複雑な応用機能も実現可能なSACF(単一アソシエーション制御機能)やMAASE(複数アソシエーション応用サービス要素)等の検討を行っており、応用層の構造がより複雑化しつつある。)
- (2)実証実験によりソフトウェアの機能や効率等についての実証を行い、実用できる見通しを得た。
- (3)ディレクトリのためのシーケンシャル・フラット・ファイルがあるかのように扱うことにより、遠隔の仮想ファイル・ストアのディレクトリ参照も可能とした。これは、現在のFTAMでは規定されていないが遠隔の仮想ファイル・ストアを利用しようとする場合に極めて有効となる機能である。
- (4)FTAMとVMSのファイル・システムとの手順の対応付けはほぼ可能である。但し必ずしも一対一には対応しない場合もあった。
- (5)今回解釈が比較的明確でどの計算機システムでも対応のとり易い無構造ファイルとシーケンシャル・フラット・ファイルをサポートしたが、これ以外のファイル構造については、現在のFTAMで規定されている一般木構造と実ファイル構造との対応に不明確な点が多い(例えば、索引順編成ファイルの場合の主キー、副キーをFTAMの木構造でどのように対応させるか等)。

(6)ファイルの内容のテキスト/バイナリの区別はVMSでは知りえないため、ファイル名のファイル・タイプから判断した。

(7)現行のセッション標準では、S-U-ABORTのユーザ・データ長に制限があるため、上位層のABORT PDUを全て実装することができず、異常終了の詳細な理由をユーザが知るができない。

(8)現行のFTAMでは、各DU(テキストやバイナリ・データ)を転送する際に任意に分割してF-DATAとして転送できるという暗黙の了解がなされているが、今後DUとして構造を持ったものが扱われる場合には、F-DATAとして不定長(無限長)のデータを扱うこととなり、この場合F-DATAの途中でも同期をとれるようなメカニズムを検討する必要がある。

(9)VMSの場合に見られるように、実際のローカル・ファイル・システムでは、ファイル編成のみでなく、レコード・フォーマットやレコード属性なども含めてファイル構造を管理しているが、現行のFTAMの仮想ファイルではこれらが十分に対応付けられる機能を持っていない。

## 7. おわりに

本稿では、筆者らの実装したOSI FTAM、ACSE、プレゼンテーション・プロトコル・ソフトウェアについて報告した。今後は、扱えるファイル形式の拡充等FTAM機能の充実および他機種システムとの相互接続試験を行っていく予定である。最後に日頃御指導頂くKDD研究所村谷所長、小野次長、浦野情報処理研究室長に感謝します。

## 参考文献

- [1] ISO DIS 8822、8823、8649、8650、8571/1-4
- [2] 鈴木、加藤、「OSIセッションレイヤ標準のインプリメント」情処学会分散処理システム研究会、Vol.24-4、1984.
- [3] 小花、加藤、鈴木、「OSI FTAM実装の基本設計」情処学会第34回全国大会、1987.
- [4] 加藤、小花、鈴木、「OSI FTAMとローカルファイルシステムとの対応関係」情処学会第34回全国大会、1986.
- [5] ISO DIS 8824、8825
- [6] 加藤、鈴木、「OSI FTAM実装における抽象構文に関する一考察」情処学会第33回全国大会、1986.
- [7] 小花、鈴木、加藤、「OSI ファイル転送プロトコルとプレゼンテーション・レイヤ・サービスについての考察」情処学会第29回全国大会、1985.