

OSIトランザクション処理(TP)プロトコルソフトウェアの設計

杉山敬三 小花貞夫 鈴木健二

国際電信電話株式会社研究所

近年、異なった複数のシステムにまたがるデータ処理を効率よくしかも高信頼に実行するために、OSI(開放型システム間相互接続)のトランザクション処理(TP)プロトコルが注目されている。OSI TPの実装では、様々な応用から利用可能とする汎用的なソフトウェアを実現することが重要である。本稿では、まずOSI TPを用いて汎用的なトランザクション処理機能を提供可能とするための基本設計として、ソフトウェア構成およびローカル資源の管理方法について論じた。次いで、TPの特徴的な機能である、コミットメント機能の実現方法やアソシエーションプール及びPDUの連結の実現方法などについて検討した。最後に、これらの検討結果に基づき、OSI TPプロトコルソフトウェアの設計概要について報告した。

Design Strategy of OSI TP Protocol Software

Keizo SUGIYAMA Sadao OBANA Kenji SUZUKI
KDD R & D Labs. 2-1-15, Ohara, Kamifukuoka-shi, Saitama, 356

Software for TP (Transaction Processing) Protocol of OSI (Open Systems Interconnection) has been designed. Since OSI TP is expected to be used by various kinds of applications, its software should be widely applicable to various applications. In this paper, the software configuration based on the Application Layer Structure (ALS) and the management method of local resources are discussed. Then, the implementation methodology for TP specific functions, such as commitment function, the pool of association and the concatenation of PDUs, are discussed. Finally, specifications of the protocol software are described. The software is under development, based on this design strategy.

1.はじめに

近年OSI(開放型システム間相互接続)の応用層の標準化も進み、現実のシステムでFTAM(ファイル転送、アクセスと管理)などが稼働し始めている。ISO(国際標準化機構)では、さらに、OSIの応用層のプロトコルとして、トランザクション処理のための標準化を進めており、現在国際規格案¹⁾となっている。トランザクション処理は、商品の受発注や予約業務など、最もよく使われるオンライン処理の形態で、標準化に対する利用者からのニーズが高い。

OSI TP(以下TPと呼ぶ)は、異なった複数のシステムにまたがるデータ処理を効率よくしかも高信頼に実行するためのメカニズムを提供するものであり、種々の応用から利用される。例えば、RDA(遠隔データベースアクセス)では、応用コンテキストの一つとして、TPとCCR(コミットメント、同時性及び回復制御)を組み合わせる使用TPコンテキストを検討している。さらに、今後ネットワーク管理やEDI(電子データ交換)など、ますます各種の標準の中でトランザクション処理の概念が導入されていくと考えられる。このため、OSI TPのソフトウェア開発に際しては、様々な応用から汎用的に利用できるソフトウェアを実現することが重要となる。

本稿では、まず各種の通信応用で汎用的なトランザクション処理機能を提供できるTPプロトコルソフトウェアを実現するための、ソフトウェア構成及びローカル資源の管理方法について論じる。次いで、TPの特徴的な機能である、コミットメント機能の実現方法やアソシエーションプール及びPDU(プロトコルデータ単位)の連結の実現方法について検討し、最後にこれらの検討結果に基づき、OSI TPプロトコルソフトウェアの設計概要について報告する。

2. OSI TPの概要

(1) トランザクションの概念

トランザクションとは、以下の4つの特性(ACID属性)を持つ関連する操作の集合である。

① 原子性(Atomicity)

すべての操作を実行するか、何も実行しない

② 一貫性(Consistency)

操作の結果が正しく実行され、制御対象データが一貫した状態に遷移する

③ 独立性(Isolation)

処理中の結果に外部からアクセスできない

④ 耐久性(Durability)

完了した結果は障害があっても変更されない

このような特性を持つトランザクション処理のうち、特に複数のシステムにまたがるトランザクションを分散トランザクションと呼び、OSI TPは分散トランザクション処理のためのサービスとプロトコルを規定している。

(2) 応用層構造(ALS)²⁾との関係

OSIでは、多様化する応用を効率良くかつ体系的に実現するため、ALSと呼ばれるモデルが規定されている。ALSにおいては、TPSU(TPサービス利用者)は応用プロセスの一部であり、AEI(応用エンティティインボケーション)であるTPPM(TPプロトコルマシン)の提供するサービスを利用する。OSI TPでは、MACF(複数アソシエーション制御機能)はTPPMの構成要素であり、複数のアソシエーションにまたがるインタラクションを調停する。また、TPPMのうち単一アソシエーションに関するプロトコルを扱う部分はTP ASEと呼ばれ、TP ASEの提供するサービスはNNサービスと呼ばれる。

(3) ダイアログ

TPSUの特定のインスタンスはTPSUI(TPSUインボケーション)と呼ばれ、2つのTPSUI間で通信を行うために、ダイアログと呼ばれる関係を確立する。ダイアログ上では、データ転送、エラー通知、トランザクションの開始やコミットメント及びロールバック、ダイアログの正常終了や異常終了、処理の同期のための通信が行なわれる。TPでは、コミットメントサービスを提供するため、CCR ASE(応用サービス要素)を使用する。また、データ転送サービスはTP自身が提供するサービスではなく、TPを利用する応用のASE(U-ASEと呼ぶ)のサービスを使用する。

(4) ダイアログツリーとトランザクションツリー

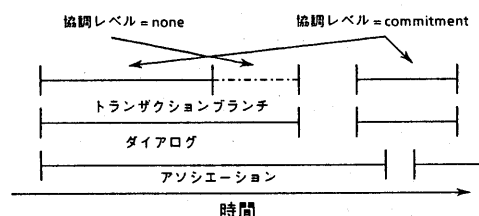


図1 トランザクションブランチ、ダイアログとアソシエーションの関係

分散トランザクション処理を行なうために協同動作するTPSUIは、ダイアログツリーと呼ばれる木構造を形成する。また、分散トランザクションのうち、ダイアログを共有する2つのTPSUIにより実行される部分をトランザクションブランチと呼ぶ。

トランザクションブランチのうち、協調レベル(ダイアログ上でコミットメントサービスが使われるかどうかを示す)がcommitmentであるものは、トランザクションツリーと呼ばれる木構造を形成する。この時、TPPMがACID属性の管理やリカバリの責任を持ち、トランザクションツリーのルートにあるTPSUIとTPPMがそのトランザクションに対するコミットメントの調停を行う。図1に、アソシエーションとダイアログ、トランザクションブランチの関係を示す。

(5)リカバリ

TPでは、障害が発生してもトランザクションツリー全体としてのコミットメントの状態の一貫性を保つトランザクションリカバリが行われる。この時、AEI間で障害以前のノードの状態を通知し合うために特別なアソシエーションが確立される。このアソシエーションは、TPチャンネルと呼ばれる。

3. OSI TPの実装におけるポイント

トランザクション処理は、複数のシステム間にもたがる処理の一貫性を保証する重要な概念であり、その応用は広範である。例えば、銀行のオンライン処理や座席予約などの各種予約業務処理に加えて、データベースアクセス、ファイル転送、ジョブ転送やオフィス文書処理などの応用が挙げられる。また、データベースアクセスとファイル転送といったように、これらの応用をまとめて一つのトランザクションとする場合も考えられる。このように、ユーザの観点からは、図2のように、TPは様々な応用と組み合わせて利用される共通的な要素と位置付けられる^[3]。

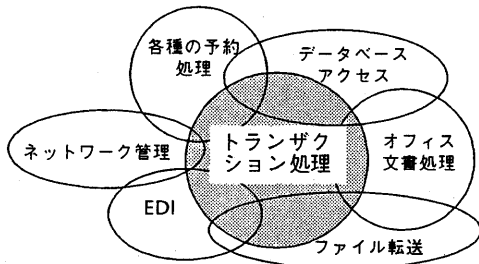


図2 トランザクション処理の位置付け

逆に、OSI TPを実装する観点からは、様々な応用と組み合わせて利用可能であり、様々な資源に対応できるように、特定の応用に特化しない汎用性を重視した設計にする必要がある。したがって、TPを実装する場合には、次の2つのことが重要である。

(1)TPを様々なU-ASEと組み合わせることを可能とする。そのため、ALSに基づき、異なる応用コンテキストで動作させることが容易なソフトウェア構成とする必要がある。

(2)TPの現在の標準では、ローカル資源とのインタフェースやアクセス機能については規定されていないが、トランザクション処理では一般にデータベースやファイルの検索・更新を伴う。したがって、TPを実装する場合には、応用が扱う様々な資源に対応できるように、ローカル資源の管理主体やその実現方法を明確にする必要がある。

4. 基本検討

ここでは、汎用的なトランザクション機能を提供するのに重要となる、ソフトウェア構成とローカル資源の管理について考察する。

4.1 応用層構造(ALS)に基づくソフトウェア構成

ALSは論理的モデルであるが、筆者等は実装においてもALSに基づいたソフトウェア構成をとり、ASEの部品化や効率的なASEの実装が可能になることを示してきた^{[4][5]}。そこで、TPの実装においても、ALSに基づいたソフトウェア構成を取ることとする。

4.1.1 TPPMの実現方法

TPPMは、従来の標準と異なり、MACFが必要となるため、MACFを含めたTPPMの実現方法が重要である。TPPMはMACFとTP-ASEから構成されるが、NNサービスは他のASEからは使用できず、MACFだけがTP ASEのユーザとなる。したがって、TPPMの実現方法には、図3のように、①MACFとTP ASEを一つのモジュールとして実現する、②MACFとTP ASEを別々のモジュールとして実現する、の2種類が考えられる。

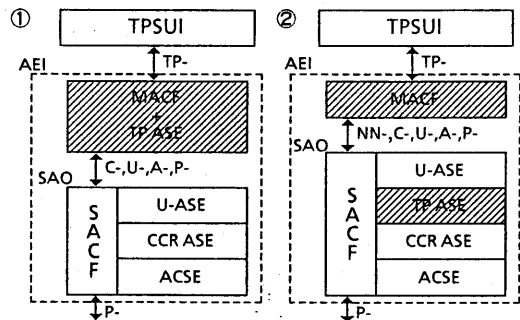


図3 TPPMの実現方法

①の場合、TPPMとして独立したモジュールを構成できるため、MACFとTP ASEにまたがる情報

の管理が容易である。しかしながら、TP ASEも含めたASE間のインタラクションやプレゼンテーションサービスの使用に関する調停などをSACF(単一アソシエーション制御機能)で一元的に管理できない。したがって、様々なU-ASEを使用して、異なる応用コンテキストのSAO(単一アソシエーションオブジェクト)が混在するような環境を構築することが難しい。

一方、②の場合には、TPPMで保持する情報がMACFとTP ASEに分散し、情報の管理が難しくなる可能性がある。しかし、プロトコルではMACFと単一アソシエーションに関する側面とに分けて記述されているためそれと親和性が高い。また、SACFで応用コンテキストが一元的に管理でき、様々なU-ASEを使用したり、従来のASEモジュールとの共存が容易になる。

したがって、②の方法を採用し、MACFとTP ASEに相当するモジュールをそれぞれ明示的に導入して、複数のアソシエーションに関する処理はMACFのモジュールに行わせ、TP APDU(応用プロトコルデータ単位)の作成・解析等の単一アソシエーションに関する処理はTP ASEのモジュールに行わせることとする。

4.1.2 MACFとSAOの実現方法

上記の検討に基づいたソフトウェア構成を図4に示す。

(1)MACF

ALSでは特に規定されていないが、MACFの機能は、複数のAEI管理やSAOの生成といったどの応用にも共通に存在する機能(ここでは共通MACFと呼ぶ)と、TPにおけるコミットメントの調停など応用に固有な機能(ここでは特定MACFと呼ぶ)とに分けることができる^[6]。このような機能モジュールに分けることにより、TP-MACFの部品化が可能になるとともに、TP-MACF以外の他の特定MACFの開発・追加も容易になると考えられる。そこで、MACFの機能を共通MACFとTP-MACFに分け、それぞれを異なるモジュールとして設けることとする。

(2)SAO

TP ASEは、種々の応用から利用できるようACSEやCCR ASEと同様に、比較的共通なASEとして位置付ける。また、SACFに相当するモジュールを明示的に導入し、SACFでSAOや応用コンテキストの管理を行うことにより、ASEの部品化を図る。一方U-ASEは、応用に固有なASEと位置づけ、SACFのモジュールとの独立性を高めることにより、異なるU-ASEへの移行を容易とする。

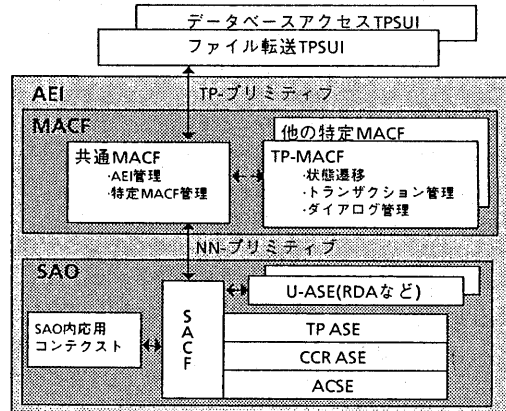


図4 ALSに基づいたソフトウェア構成

4.2 ローカル資源の管理

協調レベルがnoneの場合にはTPSUIがトランザクションを制御するため、以下では協調レベルがcommitmentの場合について検討する。

4.2.1 管理主体

ローカル資源には、TPを利用する応用固有のデータであり原子動作の制御下にある制御対象データ(Bound Data)と、トランザクションの状態情報やトランザクションの識別子などの制御情報である原子動作データ(Atomic Action Data)が存在する。

(1)制御対象データ

TPのモデルでは、制御対象データを管理する主体として、大きく分けてTPPMとTPSUIが挙げられ、どちらで管理することも許されるため、管理主体を定める必要がある。

制御対象データの管理を行うためには、管理者は応用のデータ構造を認識する必要がある。しかしながら、データベースアクセスで扱われるデータは例えば表形式であり、ファイル転送の場合には例えば無構造であるなど、応用のデータ構造は様々である。したがって、TPPMで制御対象データを管理する場合には、TPPMを特定の応用に特化させる必要があり、汎用性が失われることになる。

また、制御対象データを直接扱うU-ASEに管理させることも考えられる。その場合、U-ASEがトランザクションの単位を認識し、それに応じて管理の開始や終了を制御することが要求される。したがって、U-ASEが直接TPのプリミティブの意味を理解する必要があり、TPに依存した実装になるため望ましくない。

したがって、制御対象データの管理は、TPSUIのみで行うこととする。

(2)原子動作データ

TPのプロトコルには、トランザクションの状態や、トランザクション識別子、トランザクションブランチ識別子など、TPPMがログレコードとして使用する情報が規定されている。これらは原子動作データそのものであるため、原子動作データはTPPMで管理する必要がある。また、トランザクション識別子とトランザクションブランチ識別子は、それぞれCCRの原子動作識別子と分岐識別子に対応しているため、CCR ASEが原子動作データをログレコードとして保存することも考えられる。

しかし、現在のTPが対象としているCCRのプロトコルは3rdDIS版であり、3rdDIS版では障害回復後のCCR ASEの状態は障害発生直前の状態に依存しない。このため、CCR ASEでは原子動作データをログレコードとして保存する必要はない^[7]。また、ログレコードへの保存や検索などの効率を考えると、原子動作データは一元的に管理することが望ましい。したがって、原子動作データの管理は、TPPMのみで行うこととする。

4.2.2 管理方法

実際の管理方法は、応用や環境によって異なるが、ここでは基本的な方針について検討する。

(1)制御対象データの管理

TPSUIでは、TP-COMMIT-RESULTindを受信したらファイルの内容を確定しTP-DONEREQを発行するといったように、TPプリミティブとローカルな資源のコミットメント制御機能とのマッピングをとる必要がある。この時、データベースアクセスにおけるDBMS(データベース管理システム)のように、応用に固有の管理システムにおけるコミットメント機能とのマッピングをとる方法^[8]や、汎用的に使用できるジャーナル機能を利用する方法が考えられる。また、TPのトランザクションの単位とローカルな資源におけるトランザクションの単位を対応させる必要がある。例えば、データベースアクセスにおいて、一つの資源が複数のテーブルから構成されている場合、TPSUIでは複数のテーブルを一つのトランザクションの単位として扱う必要がある。

(2)ヒューリスティック判断への対応

上位にコミット要求を発行した後に、通信障害等の理由により、上位からの最終結果通知(コミット/ロールバック)を待たず、自身で最終結果を決め制御対象データを解放することをヒューリスティック判断と呼ぶ。ヒューリスティック判断を行うかど

うかは、分散したシステム間でトランザクションの状態に不一致が生じないことが重要なのか、システム内の他のユーザに資源を早く解放することが重要なのかにより、ユーザが決定することとした。

5.設計の概要

以上の検討結果に基づいて、TPプロトコルソフトウェアの設計を行った。TPプロトコルとしてDIS版の標準に準拠し、非連鎖トランザクションを実現するために全機能単位を実装する。また、CCRプロトコルとして3rdDIS版の標準に準拠する。プレゼンテーション層以下やACSE、CCR ASE等のモジュールについては、既に開発済みのものを利用することとする^{[8],[9]}。

5.1 MACFの実現

MACFの詳細な機能やSACFとの関係については、ALSやTPの標準では具体的に示されていないため、以下のように実現した。図5に、MACFにおける情報の関連を示す。

(1)共通MACFの機能

共通MACFではAEI管理及び入出力管理を行う。

①AEI管理

MACFで持つ情報のうち、AEI識別子やAEタイトル、QOS(サービス品質)、対応するSAOのアソシエーション識別子など、各種応用で共通に使われる情報の管理を行う。これらの情報はアソシエーション毎に管理され、特定MACFから参照される。また、特定MACFを起動し、共通MACFと特定MACF内の情報の関連付けを行う。ここでは、同時に複数のTPSUIを制御可能とするため、複数のAEIをサポートする。

②入出力管理

アプリケーションプロセス及びSAOのモジュールからプリミティブを受信し、対応する特定MACFにプリミティブを渡すとともに、特定MACFから受信したプリミティブをアプリケーションプロセス及びSAOに振り分ける。この時、共通MACFでは複数のアプリケーションプロセスを一意に識別する必要がある。これは、アプリケーションプロセスの発行するプリミティブに、アプリケーションプロセス自身のプロセス番号を含めることで実現する。

(2)TP-MACFの機能

特定MACFの一つであるTP-MACFでは、ダイアログ/チャンネル管理、状態遷移管理及びトランザクション管理を行う。

① ダイアログ/チャンネル管理

ダイアログやチャンネルはアソシエーションの上位の概念であるため、これらの管理は、個々のアソシエーションを扱うSAOのモジュールではなく、MACFに持たせることとする。

ダイアログ確立時には、利用可能なアソシエーションを割当て(5.4参照)、プロトコルで使用するダイアログ/チャンネル識別子を作成する。また、TPSUタイトルやダイアログ/チャンネル識別子、TPの機能単位など、ダイアログ毎に情報の管理を行い、ダイアログとアソシエーションやチャンネルとアソシエーションの対応をとる。

② 状態遷移管理

SAOとTPSUIからのプリミティブの受信イベントに基づいて、TP-MACFとしての状態遷移を行う。この時、TPサービスとNNサービスをマッピングし、各SAOに対しトランザクションの状態(レディ、コミット等)に応じたCCRプリミティブの発行や受信を行うことで複数アソシエーションにわたるコミットメントの調停を行う。

ここで、ALSのモデル上、MACFとSAO間では、ACSE, CCR ASE, TP ASE, U-ASE, プレゼンテーションのプリミティブが存在する。しかしながら、TP-COMMITreqが発行された場合にはC-PREPAREreqのユーザはMACFであるが、TP-PREPAREreqの場合にはTP ASEであるなど、ASEのユーザが分散するとサービスの状態管理が複雑になると考えられる。また、TP ASEは単一アソシエーションに関するTPプロトコルの状態を知る必要があるため、上記のプリミティブの授受を把握できることが望ましい。そこで、TP-MACFがSAOに発行するプリミティブはすべてNNプリミティブの形に統一し、TP-ASEを通過させることで、TP ASEでアソシエーションの状態を管理できるようにし

た。ローカルに使用するNNプリミティブの一覧を表1に示す。

表1 ローカルNNプリミティブの一覧

ローカルNNプリミティブ	対応するプリミティブ
NN-RELEASE	A-RELEASE
NN-DATA	U-ASEのプリミティブ
NN-BEGIN	C-BEGIN
NN-PREPARE*	C-PREPARE
NN-READY	C-READY
NN-COMMIT	C-COMMIT
NN-ROLLBACK	C-ROLLBACK
NN-RECOVER	C-RECOVER
NN-TOKEN-GIVE	P-TOKEN-GIVE
NN-TOKEN-PLEASE	P-TOKEN-PLEASE

*: NN-PREPAREは既に存在しているため、特別なパラメタを設け既存のサービスと区別する。

また、標準では、TP APDUがCCRのどのサービスプリミティブで運ばれるかをTP ASEとMACFで認識するために、ローカルパラメタ(local parameter)が使用されている。このローカルパラメタはMACFとSAO間で共有されるべき情報であるため、ローカルパラメタを必要とするNN-REJECT/NN-ABORT/NN-P-ERRORプリミティブのパラメタに、この情報を追加することとする。

③ トランザクション管理

協調レベルがcommitmentのダイアログが存在する場合に、トランザクション識別子やトランザクションブランチ識別子の割当てを行い、複数のトランザクションブランチの管理や一つのTPPMにおけるトランザクション全体の状態の管理を行う。また、システムクラッシュ時に複数のSAOに対するリカバリを起動するなど、リカバリの調停を行う。

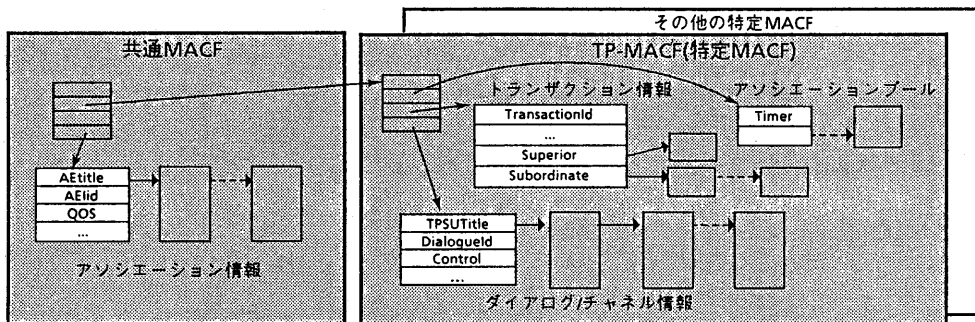


図5 MACFにおける情報の関連

これらの情報は、協調レベルの変更に伴い、動的に生成されたり消滅する可能性がある。また、トランザクションツリー内のノードの位置や連鎖モードであるのか等によって、状態遷移や下位サービスへのマッピングが複雑に変化する。したがって、動的な協調レベルの変更に柔軟に対処し、原子動作データを一元的に管理するため、協調レベルに関わらず管理する必要があるダイアログに関する情報はダイアログ/チャンネル管理部で管理し、協調レベルがcommitmentの時だけ必要となるトランザクションに関する情報はトランザクション管理部で別々に管理することとする。

5.3 SAOの実現

(1)SACF

SACFでは、SAO/アソシエーション管理、応用コンテキスト管理、応用ディレクトリ機能、入出力管理を行わせる¹⁵⁾。

SAO/アソシエーション管理では、アソシエーションに関する情報のうち各ASEに共通な情報管理するとともに、対応するASEへプリミティブを振り分ける処理を行う。この時、NN-DATAプリミティブをTP ASEにマッピングすると、U-ASEが変わる毎にTP ASEにおける下位サービスへのマッピングが異なり、TP ASEが汎用的でなくなる。そこで、NN-DATAについては、直接U-ASEにマッピングすることとする。応用コンテキスト管理では、ファイルとして作成された応用コンテキストの知識を読み出して、SACFの処理を制御する。また、応用ディレクトリでは、AEタイトルからプレゼンテーションアドレスへのマッピングを行い、入出力管理ではプレゼンテーションのモジュールなど他のモジュールとの通信を管理する。

(2)TP ASE

TP ASEでは、MACFからのNNプリミティブや下位からのC-,P-,A-プリミティブを受信して、TP APDUの作成・解析を行い、対応するサービスへマッピングする。また、ダイアログの終了したアソシエーション上でstray PDUのパーキングを行う。

5.4 アソシエーション管理

TPでは、アソシエーションをプールに保存しておき、必要に応じてフリーのアソシエーションをダイアログに割当てるアソシエーションプールの概念が導入されている。アソシエーションプールの管理方法はローカルマターとなっているため、以下のように実現することとした。

(1)ダイアログの確立と終了

①確立時

ダイアログ確立要求を受信した時には、ダイアログ確立に要する時間を短縮するため、まずプールを検索することとする。利用可能なアソシエーションが存在した時はダイアログに割当て、存在しなければ確立要求を発行し、アソシエーション確立後にダイアログに割当てる。また、常に同一の相手と通信する場合など、ダイアログを固定的に利用する場合をサポートするため、通信相手のAEタイトル等の情報をファイルとして保存し、プログラムの起動時にそれを読み出してアソシエーションを自動的に確立し、プールする機能も設ける。

②終了時

プールへの保存を利用者が制御できるようにするため、ダイアログ終了要求プリミティブにローカルなパラメタを設け、プールに保存したい場合にはそのパラメタを指定し、指定がない場合には対応するアソシエーションを解放することとする。また、アソシエーションがプールに保存されたまま長時間使用されないことを防ぐために、無通信監視タイマを設ける。このタイマは、アソシエーションがプールに保存された時に設定され、そのアソシエーション上で通信が行われたときに解除する。また、タイマが満了した場合には、自動的にアソシエーションを解放する。

(2)TPチャンネルの確立と終了

アソシエーション管理の観点では、TPチャンネルもダイアログと同様である。ただし、アソシエーションをTPチャンネルとして使用した後で、そのままTPチャンネルを保持しておくことはせず、TPチャンネルを終了させ、対応するアソシエーションを解放することとする。

5.5 PDUの連結と埋め込み

TPでは、パフォーマンスや効率を向上させるために、PDUの連結(concatenation)や埋め込み(embedding)を行う場合があるため、その実現方法を考慮する必要がある。

(1)連結

従来、FTAMのグループ化制御のように、単一のASEのAPDUに対する連結を行う場合には、ASEの機能として連結をサポートしていた¹⁶⁾。しかしながら、TPでは、TP APDUとUASE-APDUのように異なるASEのAPDUに対する連結を行う。この時、TP ASEのような特定のASEにその機能を持たせる

ことは、ASEの部品化の観点から好ましくないため、連結機能はSACFに持たせることとする。

連結を行う場合には、各ASEで作成されたAPDUを受信したSACFがAPDUのキューイングを行い、連結に必要なAPDUを全て受信したらキューイングを終了する。この場合、各ASEからAPDUを受信する順番が変化する可能性があるため、連結のパターンと比較して異なる場合には並び換えを行う必要がある。さらに、マッピングの優先順位を比較して、マッピングすべき下位サービスを決定しプリアミティブを発行する。

(2)埋め込み

TP-PREPARE-RIをC-PREPAREreqにマッピングするというように、PDUを埋め込むためのASE間の依存関係は、各ASEが把握している。また、埋めこまれたAPDUをSACFで意識して処理を行う必要はない。したがって、埋め込みについては、埋め込む側のASEのモジュールで行うこととする。

6. 考察

(1)ALSに基づいたモジュール構成とし、応用コンテキストの管理を一元化することで、様々なU-ASEへの対応を可能とした。また、MACFを共通MACFと特定MACFに分け、どの応用にも共通に存在する機能を共通MACFに持たせることで、MACFを使用する他の応用の開発・追加を容易とした。

(2)制御対象データの管理機能をTPSUIに持たせることにより、TPPMは特定の応用が扱う資源に依存しない実装とした。原子動作データはTPPMに持つので、システムクラッシュなどにより、TPSUIとTPPMで状態が異なる可能性がある。そのため、TPSUIに制御対象データを管理させる場合には、リカバリの時に、障害発生以前のTPPMのトランザクションの状態をTPSUIが知る必要がある。

(3)協調レベルに関わらず必要となる情報と、協調レベルがcommitmentの時だけ必要となる情報を分けて管理することにより、動的な協調レベルの変更やリカバリへの効率的な対応を容易とした。

(4)今回の実装では、アソシエーションプールは、TPに特有な機能としてTP-MACFに持たせたが、アソシエーションプールの考え方が今後他の応用にも適用される場合には、共通MACFの機能とすることが望ましいと考えられる。

(5)PDUの連結を開始する方法として、利用者の指示による方法と、SACFが固定的に行う方法が考えられる。UASE APDUの発行される数が不明な場合や応用でのレスポンスを即座に受信したい場合には、

SACFで固定的に行わず、利用者の指示で行えるようにすることが望ましい。

(6)TPのサービス定義では、TPを利用する応用のデータはTP-DATAプリミティブという一つの形式で表現される。しかし、応用には様々なものが考えられるため、MACFがTPSUIに提供するインタフェースでは、異なったデータ形式にも対応できるインタフェースを提供することが重要である。

(7)コミットメントは、そのトランザクションに関与するすべてのダイアログに対して行われる。したがって、TPSUIとTPPMのインタフェースでは、データ転送などのサービスに対してはダイアログを識別する情報が必要となるが、コミットメントサービスに対してはトランザクションを識別する情報が必要となる。

(8)トランザクション処理機能を持つ応用標準を今後開発する際には、TPとの機能の重複を避け、多様化するOSIの応用標準を効率よく開発するために、TPと独立してトランザクション処理機能を規定しないことが重要であると考えられる。

7. おわりに

本稿では、RDAやEDIなど様々な応用から汎用的に利用できるTPのソフトウェアを実現するためには、様々なU-ASEとの組み合わせを可能とする応用層構造(ALS)のソフトウェア構成とすることと、応用が扱う様々な資源に対応できることの2点が重要であることを示した。現在、VAX8700(OS:VMS)上に実装を行っている。今後は、U-ASEとしてのRDAやファイル転送^[10]の実装や、TPの拡張機能の検討を進める予定である。最後に日頃御指導頂くKDD研究所小野所長、浦野次長に感謝します。

参考文献

- [1] ISO/IEC JTC1 DIS10026-1~3, OSI TP
- [2] 小花, 杉山, 鈴木, 「OSIにおけるトランザクション処理の位置付けに関する一考察」第36回情処全大
- [3] ISO 9545, Application Layer structure
- [4] 小花, 西山, 杉山, 鈴木, 「OSIにおける応用層構造(ALS)の実現方法の提案」第37回情処全大
- [5] 杉山, 西山, 小花, 「OSIの応用層構造(ALS)におけるSACFの実装」第39回情処全大
- [6] 杉山, 小花, 鈴木, 「汎用OSI TPプロトコルソフトウェアの基本設計」第41回情処全大
- [7] 杉山, 小花, 鈴木, 「OSI CCRにおける障害回復機能の実現手法」第40回情処全大
- [8] 小花, 杉山, 鈴木, 「OSI CCRの実装」情処学会DPS研究会43-3, 1989
- [9] 小花, 加藤, 鈴木「OSIプレゼンテーション, ACSE, FTAMプロトコルの実装と評価」情処学会論文誌30-7, 1989
- [10] 杉山, 小花, 鈴木, 「OSI TPを利用したファイル転送プロトコルの提案」第40回情処全大