

OSI応用層ソフトウェア実装における ASEソフトウェアモジュール再利用化の方式

6V-3

小花 貞夫 杉山 敬三 鈴木 健二

国際電信電話(株) 研究所

1. はじめに

OSI(開放型システム間相互接続)通信を利用する応用業務が高度化、多様化するに従い、ひとつの応用業務で特定の単一応用層プロトコル(FTAM、MHS等)を利用したり、また、複数の異なる応用層プロトコルを同時に利用するといった様々な利用形態に対応する応用層ソフトウェアの実装が重要となる¹⁾。このような応用層ソフトウェアを効率的に実装するため、本稿では、応用層における基本的な機能要素であるASE(応用サービス要素)を再利用可能なソフトウェアモジュールとして実現し、それらを組み合わせて様々な形態で利用される応用層ソフトウェアを体系的に開発可能とする2つの方式(スタック方式とSACF方式)を提案する。また、両方式を比較・評価し、それぞれの適用領域を明確にする。

2. 応用層ソフトウェア実装におけるソフトウェアモジュール再利用化の必要性

応用層構造(ALS)²⁾に従うと、応用層の構成要素のうち、特にSAO(単一アソシエーションオブジェクト)は独立のプロトコル機械(PM)が規定されるASEの組合せにより実現され、この組合せとそれらの依存関係は要求される応用層プロトコルの種類により異なる。このため、各種応用業務で必要となる応用層プロトコルをサポートするソフトウェアを効率的に実装するには、一般に各層単位にソフトウェアをモジュール化する下位層の場合とは異なり、さらに応用層内部のASEも再利用可能なソフトウェアモジュールとして実現する必要がある。

3. スタック方式とSACF方式の提案

ASEソフトウェアモジュールの再利用化の方式として、図1に示す特徴的相違を持つ2つの方式(スタック方式とSACF方式)を提案する。

(1)スタック方式

ALSにおけるSAO内のSACF(単一アソシエーション制御機能)を実現するソフトウェアモジュールを明示的に導入せず、ASEのソフトウェアモジュールを、応用層プロトコルの種類毎に異なる組合せで階層的(スタック)に積み上げる。

(2)SACF方式

SACFのソフトウェアモジュールを明示的に導入

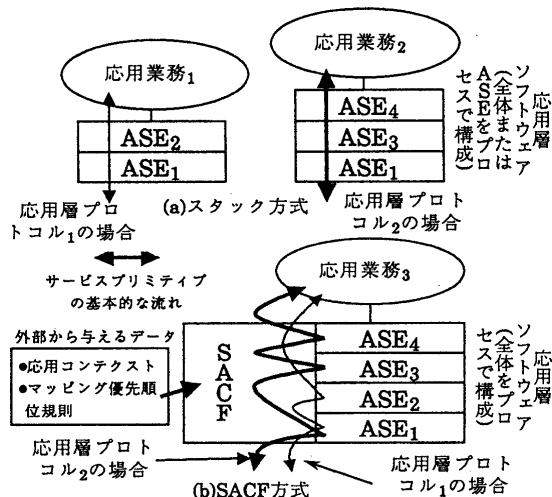


図1 スタック方式とSACF方式

し、応用業務プログラムから要求される応用層プロトコルの種類に応じて、ASEのソフトウェアモジュールの組合せと依存関係を動的に制御する³⁾⁴⁾。これらの方式は、表1に示すように、a)ASEの組合せとその依存関係の制御方法、b)SAO内のASEによるアソシエーションの共有方法、c)各モジュールの機能分担およびd)モジュール間インタフェースの点で実現方法が異なる。

4. スタック方式/SACF方式の比較と適用領域

スタック方式とSACF方式を、a)応用層プロトコル種類の変更に対する柔軟性、b)複数応用層プロトコルの同時サポート、c)ASEモジュール作成における自由度およびd)ソフトウェア規模の観点から比較評価する。(表2)。比較結果より、以下に両方式の適用領域を示す。

(1)スタック方式が有効な適用領域

- ひとつの応用層ソフトウェアで単一応用層プロトコルをサポートすることを主目的とする場合
- 各ASEモジュール内部の作りに自由度を持たせ開発者が独自の方法により作成可能とする場合

(2)SACF方式が有効な適用領域

- ひとつの応用層ソフトウェアで複数応用層プロトコルを同時サポートすることを主目的とする場合

Implementation methodologies of OSI Application Layer Software composed of Reusable ASE Software Modules

Sadao OBANA, Keizo SUGIYAMA and Kenji SUZUKI

KDD R & D Laboratories

- ASEモジュールに全く変更を加えずに再利用を図る場合

5. おわりに

本稿では、様々な応用層プロトコルの利用形態に対応する応用層ソフトウェアを体系的に開発可能とするためのASEモジュール再利用化の2つの方式を提案し、両方式の適用領域を明確にした。特に、パソコン等一般に主記憶サイズや処理能力に制約のある計算機環境で、特定の単一応用層プロトコルをサポートする端末応用を実現するにはスタック方式が、また、高機能ワークステーションや汎用機等で複数の応用層プロトコルを同時に扱うような高度

な応用層ソフトウェアを実現する場合には、SACF方式が有効である。最後に日頃御指導いただくKDD研究所小野所長、浦野次長に感謝します。

参考文献

- (1) 小花, 杉山, 鈴木: ソフトウェアモジュールの再利用化を考慮したOSI応用層ソフトウェアの実装方法, 情処学会, DPS-55-1 (1992.5).
- (2) ISO/IEC 9545, Application Layer Structure (1989).
- (3) 小花, 杉山, 鈴木: OSI応用層プロトコルソフトウェア実装のためのソフトウェア部品化に関する検討, 44回情処全大, 4L-5 (1992.3).
- (4) 杉山, 小花, 鈴木: OSI応用層プロトコルソフトウェア部品化における応用コンテキスト制御方式の提案, 44回情処全大, 4L-6 (1992.3).

表1 スタック方式とSACF方式の実現方法

実現項目	スタック方式	SACF方式
ASEの組合せと依存関係の制御	スタック状に積み上げたASEモジュールの固定的な組合せとスタックにおける上下関係により実現。依存関係にあるASEモジュール間に設けたキューや関数等の通信パスを通してサービスプリミティブをルーティング。	SACFモジュールが、外部から与える1)結合されるASEモジュールと呼出し関数との対応、2)応用層プロトコルの種類毎のASEモジュールの組合せと呼出し順序(依存関係)、3)ASEモジュールと抽象構文名との対応に関する情報(応用コンテキストデータ)を用いて、サービスプリミティブを汎用的にルーティング。
ASEによるアソシエーションの共有	すべてのASEモジュールにA-ASSOCIATE等のアソシエーション制御サービスプリミティブを経由。各ASEモジュールでアソシエーションの状態を管理。	SACFモジュールがアソシエーション状態を一元管理し、ASEモジュールでは管理せず。すべてのASEモジュールから状態の参照、変更可能。
各モジュールの機能分担	各ASEモジュールは、サービスプリミティブ/APDUの解析・生成、状態遷移、アソシエーション管理、同一ASEに関するPDUの連結/分離ならびに依存関係にあるASEモジュールへのサービスプリミティブのルーティング等を実現。ただしROSE/CCR ASE等には、標準で規定される機能に加え、アソシエーション制御サービスプリミティブ処理を追加。	SACFモジュールは、応用コンテキストデータに従うサービスプリミティブルーティング、アソシエーション管理ならびに外部から与えるマッピング優先順位規則データに従うASEにまたがるAPDUの連結/分離を実現。各ASEモジュールは、サービスプリミティブ/APDUの解析・生成、状態遷移、同一ASEに関するPDUの連結/分離等を実現。
モジュール間のインタフェース	各ASEモジュール間では、基本的に、各ASE標準で規定されるサービスプリミティブを、他ASEへのキューまたはASEを呼び出す関数を通して交換。	SACF/ASEモジュール間では、a)SACFモジュールからASEモジュールの呼出し、b)ASEモジュールがSACFモジュールにアソシエーション情報を生成・削除・参照・変更、c)ASEモジュールがサービスプリミティブを送信等を行う約20種類の関数がある。

表2 スタック方式とSACF方式の比較・評価

比較項目	スタック方式	SACF方式
応用層プロトコル種類の変更に対する柔軟性	既存ASEモジュールへ変更必要な場合有り。 1) モジュール間通信パスの再定義 2) 複数ASEに関するAPDU連結処理の変更等	既存ASEモジュールへ変更無し。 1) 応用コンテキストデータ/マッピング優先順位規則データの変更 2) SACFモジュール/ASEモジュールのオブジェクト再リンク
複数応用層プロトコルの同時サポート	1) 実行時にASEモジュールの共有を図るスタック構成 2) 共有されるASEモジュールは、応用層プロトコルの種類毎に他ASEモジュールとの依存関係を制御	1) 必要なASEモジュールをSACFモジュールにリンク 2) 複数の応用コンテキストデータ/マッピング優先順位規則データを定義
ASEモジュール作成における自由度	ASE標準で規定されるサービスプリミティブに従えば、ASEモジュール内部の作りに自由度大。	SACFモジュールとの協調動作のための関数(約20種類)の制約に従う必要が有り。
ソフトウェア規模	使用ASEモジュールが少ない場合、SACF方式より応用層ソフトウェア全体の規模小。SACF方式よりASEモジュールの規模大。	使用ASEモジュールが多い場合、スタック方式より応用層ソフトウェア全体の規模小。スタック方式よりASEモジュールの規模小。