

分散処理システムの処理モデルの一提案

森澤 好臣, 岩田 裕道, 外山 晴夫

日本ユニシス株式会社

〒135 江東区豊洲 1-1-1

{Yoshitomi.Morisawa, Hiromichi.Iwata, Haruo.Toyama}@unisys.co.jp

分散処理システムの処理形態の分類としてガートナグループの分散コンピューティング・モデルが有名である。我々は、分散処理システムを分散処理におけるデータ保管場所と処理形態の観点より7つのモデルに分類し、クライアント/サーバ・システム構築のための処理モデルとして利用している。7つの処理モデルとモデルを構成要素としたクライアント/サーバ・システム構築体系であるオープン・ソリューション・フレームワークを説明する。このフレームワークの目的は、多様な製品選択肢の洪水に悩まされているクライアント/サーバ・ユーザに、簡便なガイドを提供することである。このフレームワークを参照にすることにより、従来多くの時間と労力を費やし、かつ品質面でも問題の多かったC/Sシステムの構築をより迅速にかつ低コストで、高品質に実現できる。

A Proposal of Computing Models for Distributed Processing Systems

Yoshitomi Morisawa, Hiromichi Iwata, Haruo Toyama

Nihon Unisys, Ltd.

1-1-1 Toyosu, Koto-ku Tokyo 135, Japan

{Yoshitomi.Morisawa, Hiromichi.Iwata, Haruo.Toyama}@unisys.co.jp

The distributed computing model of Gartner Group is a well-known classification of processing style in the distributed processing systems. We have classified these systems into seven models from the viewpoints of the location of data storage and the processing style between client and server, and utilizing as computing models to implement client/server systems. This paper explains seven computing models and the Open Solution Framework as a framework for implementing client/server systems. The purpose of the framework is aimed to provide a simple guide for client/server users who are suffering by a flood of various choice of products. Utilizing the framework, we may implement client/server systems, which needed more time and cost and had problems on their quality, with more rapid, lower cost and higher quality.

1. はじめに

1990年代始めの米国経済の停滞期に始ったオープン化の本格的な波が数年遅れで日本でも始まり、オープン化、クライアント/サーバ(C/S)システム、ダウンサイジング等のキーワードが書籍、新聞、雑誌を賑わしている。また数多くのクライアント/サーバ用のソフトウェアが市場に氾濫している。新聞見開きサイズのC/S情報基盤の一覧表でさえC/Sシステム関連のツールが500種類程列挙されている⁽¹⁾。一説では全世界で7千とも1万とも云われている。ユーザがこれらの多くのソフトウェアを活用してアプリケーションを構築する時、使用するソフトウェア間での相互接続性や相互運用性、所謂ソフトウェア間の整合性検査に多くの時間と労力を費やすのが現状であり、弊社のC/Sシステムのシステム・インテグレーションの経験でも、整合性検査に多くの時間と労力を費やす事例が多い。これは、分散処理システム、特にC/Sシステムの構築が発展途上の技術であり、標準に基づく安定したソフトウェア・ツールや確立した方法論がまだ存在していないためである。

分散処理システムを構築するとき、そのシステムの企業情報システム全体における位置付けやそのシステムの分散処理形態、そして採用するハードウェア・プラットフォームやミドルソフトウェアを含んだ情報基盤および開発・実行・運用のためのソフトウェアの選択が重要になる。

企業情報システムの分類として、従来、金融業界に於ける勘定系、情報系、OA系などの分類が著名であるが、新しい分類が先進企業によって試みられている。取り扱う情報の性格によって「対顧客系」、 「対オフィス系」そして「対マーケット系」に分類してシステムの見直しを考えたり⁽²⁾、「基幹系業務」、「情報利用系業務」そして「身の回り業務」に体系化し、システムの方針を明確にしようとしている企業もある。生の情報をリアルタイムでオンライン・トランザクション処理する基幹系情報システム、基幹系情報システムを補完する情報系システムと経営的判断をする管理系システムを含む支援系情報システムに大別することを提案している企業もある⁽³⁾。またビジネスの将来像として競争相手、顧客、関連企業、供給業者をも結んだ企業間コンピューティングの重要性が指摘されている⁽⁴⁾。

情報基盤の情報技術のフレームワーク(体系)は、1987年のIBM社のSAAで始まったプロプラエタリなフレームワークで始まり、1990年に入りユニシス社より発表されたUnisys Architecture⁽⁵⁾や1994年に発表されたIBM社のOpen Blueprint⁽⁷⁾のような国際標準や業界標準ベースのオープン

なフレームワークへと発展している。Posix OSF⁽⁶⁾が国際標準へと進展中であるが、技術体系としてはまだ決定打がないのが現状である。

本論文では、分散処理システム、特にC/Sシステムの処理モデルを提案する。本論分の第2章で現存する分散処理モデルを紹介し、第3章では、C/Sシステムの7つのモデル、クライアント/サーバ・ソリューション(C/SS)モデル、を説明する。第4章でこのモデルを利用したオープン・ソリューション・フレームワークを概観し、7つのモデルをベースにしたC/Sシステムを構築・実行・運用して他システムとの接続を支援する時に使用するプロダクトの組み合わせパターンを示し、推奨するプロダクトの組み合わせを例示する。

2. 分散処理モデル

分散処理モデルを考察する時、大きく、プラットフォームの多階層環境と協調型クライアント/サーバ処理の二つに分けることができる。前者は、メインフレーム・ホスト、サーバ、ワークステーションのプラットフォームの三階層モデルがその例である。後者は、クライアントからの要求に対してサーバが回答するクライアント/サーバ処理方式がその例である。そして分散形態によって、分散表示、遠隔表示、分散ビジネス・ロジック、遠隔データ管理、そして分散データ管理の5つに細分化している⁽⁸⁾。後者の有名な例として、ガートナグループによるものがあり、これは図1で示すように、ネットワークの所在場所によって、分散表示、遠隔表示、分散機能、遠隔データアクセスと分散データ処理の5つに分類している。

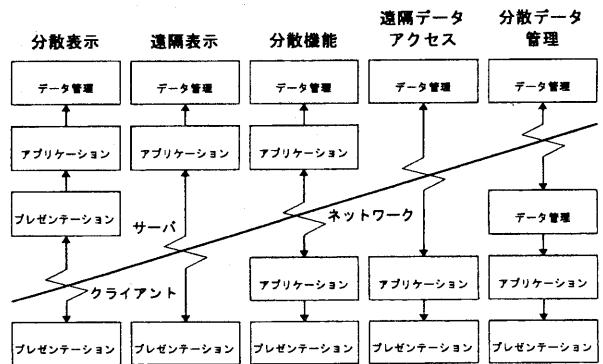


図1 分散コンピューティング・モデル

また代表的な顧客のアプリケーションをサーバ側し、分散システムを、遠隔表示型、フロントエンディング型、分散ロジック型、データステージング型、資源集中型、プロセス駆動型と複数アプリケーション型の7つの型に分類した例も

ある⁽¹⁰⁾。

今後の分散処理システムを考える時、クライアント/サーバ処理モデルに於ける三層モデル(3 tier model), いわゆるプレゼンテーション, アプリケーション・ロジック, データ管理の三層をベースにすることが妥当な選択であり, これらがどの様に分散させるかの観点よりモデル化すべきと考えている。

3. クライアント/サーバ・ソリューション・モデル

3.1 モデルの分類

クライアント/サーバ処理でのデータの保管場所の観点およびクライアントとサーバ間の処理形態の観点より表1に示すように七つの基本モデルに分類し, 基本モデルをクライアント/サーバ・ソリューション(C/SS)モデルとした。

表1 C/SS モデル

データ形態 処理形態	集中	分散	
		垂直	水平
トランザクション処理	C/SS モデル(1)	C/SS モデル(2)	
依頼応答処理	C/SS モデル(3)	C/SS モデル(4)	
遅延処理	C/SS モデル(5)	C/SS モデル(6)	C/SS モデル(7)

現実のC/Sシステムは, これらの基本モデルの一つ以上の組み合わせで構成されている。この分類は, 通常のクライアント/サーバ・モデルに対して時間の概念及び垂直・水平分散の概念を導入したことになる。原則として, 「プレゼンテーション(P)」は, クライアント側にあり, 「データ(D)」は, サーバ側にあることを仮定している。

本章のモデル図で使用する記号を説明する。

- ・P: プレゼンテーション
- ・AP: アプリケーション・ロジック
- ・DM: データ管理
- ・D: データ

3.2 モデル(1): 集中トランザクション処理モデル

集中トランザクション処理モデルを図2に示す。

メッセージの流れと処理の流れを説明する。クライアント

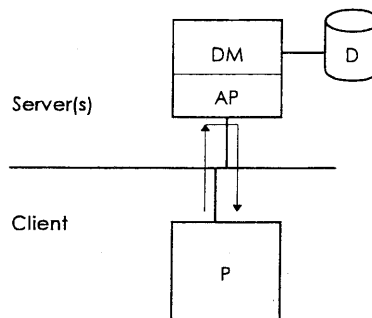


図2 集中トランザクション処理モデル

上のPよりトランザクションがAPに送られる。トランザクションの処理結果でDの中のデータがDMにより検索・更新・追加・削除される。そして処理結果がクライアントに返される。APとDMが別のサーバのこともあり, この時このモデルは, 3層モデル(3 tier model)と呼ばれている。

このモデルは, トランザクション管理プログラムを利用する単一データベースによる追加更新主体のトランザクション処理に適する。トランザクション負荷が大きい場合は, アプリケーション・サーバとデータベース・サーバに分割されることがある。アプリケーションの例としては, 受発注管理, 商品在庫管理, 株式注文処理, 生産管理, 小売POSシステム等がある。

3.3 モデル(2): 分散トランザクション処理モデル

分散トランザクション処理モデルを図3に示す。

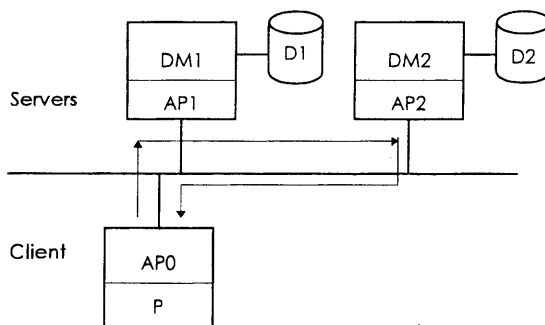


図3 分散トランザクション処理モデル

メッセージの流れと処理の流れを説明する。クライアント上のAP0より, トランザクションがAP1に送られる。そのトランザクションの一部の処理結果でD1の中のデータがDM1により検索・更新・追加・削除される。そして処理されなかった又は処理された結果のトランザクションがAP2に送られ, 処理結果でD2の中のデータがDM2によ

り検索・更新・追加・削除される。そして処理結果がクライアントに返される。必要に応じて、D1 と D2 の整合性を保つためのメカニズムが使用される。APx と DMx が別のサーバである場合がある。このときこのモデルは3層モデル(3 tier model)と呼ばれている。

このモデルは、複数のデータベースへの更新や問い合わせが複雑にからむ基幹系トランザクション処理に適する。トランザクション負荷が大きい場合はアプリケーション・サーバとデータベース・サーバに分割されることがある。トランザクション管理プログラムのメッセージ・キューイングなどを利用する非同期処理を伴う場合は、モデル(5)(6)(7)のいずれかに分類される。アプリケーションの例としては、銀行の勘定系システム、座席予約システム、工場生産管理システム等がある。

3.4 モデル(3) : リモート・データ・アクセス処理モデル

リモート・データ・アクセス処理モデルを図4に示す。

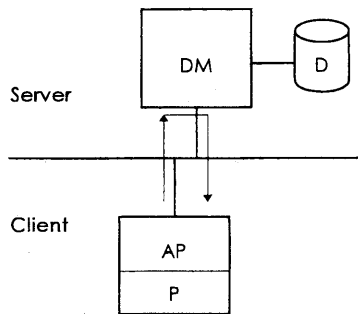


図4 リモート・データ・アクセス処理モデル

メッセージの流れと処理の流れを説明する。クライアント上の AP よりメッセージ (SQL 等) が DM に送られる。メッセージ形式で表された要求により D の中のデータが DM により検索・更新・追加・削除される。そして処理結果がクライアントに返される。

このモデルは、意思決定支援システムに代表されるいわゆる EUC (エンドユーザ・コンピューティング) , および問い合わせ応答型の単純な即時処理に適する。AP 負荷が大きい場合の AP 機能や複数のクライアントに共通の AP 機能などはサーバに置かれる。EUC のアプリケーションの例には、予算計画、財務分析、市場調査・分析、売上分析、所要量計画、需要予測などの各種統計・分析・報告がある。問い合わせ応答処理のアプリケーションの例には、顧客サービス、営業支援、各種照会処理、情報提供サービスなどがある。

3.5 モデル(4) : 分散データ・アクセス処理モデル

分散データ・アクセス処理モデルを図5に示す。

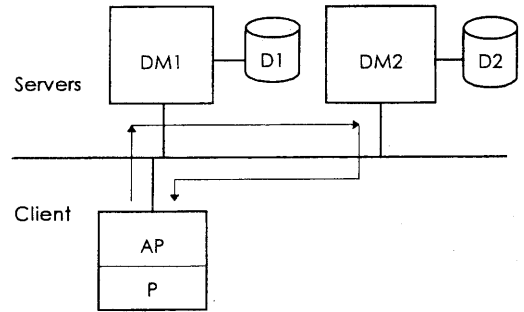


図5 分散データ・アクセス処理モデル

メッセージの流れと処理の流れを説明する。クライアント上の AP より、メッセージ (SQL 等) が DM1 に送られる。そのメッセージの(一部の)要求により D1 の中のデータが DM1 により検索・更新・追加・削除される。そして処理されなかったメッセージが、または処理された結果のメッセージが DM2 に送られ、D2 の中のデータが DM2 により検索・更新・追加・削除される。そして処理結果がクライアントに返される。必要に応じて、D1 と D2 の一貫性を保つためのメカニズムが使用される。

このモデルは、複数のデータベースやファイルを同時にアクセスする EUC および問い合わせ中心の即時処理に適する。すでに存在するデータベースの共有による情報の有効活用ができる。モデル(3)のアプリケーションはほとんど含まれるが、他に、全社売上統計、全社生産性データ分析などがある。

3.6 モデル(5) : 集中メッセージ処理モデル

集中メッセージ処理モデルを図6に示す。

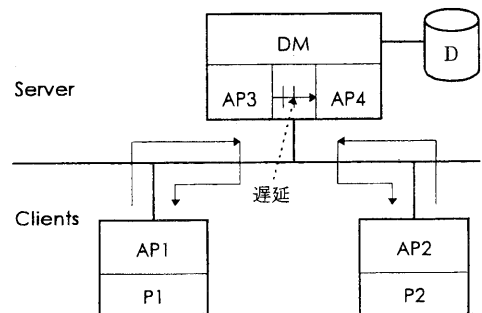


図6 集中メッセージ処理モデル

メッセージの流れと処理の流れを説明する。クライアント上の AP1 より、メッセージ (SQL, 文章等) が AP3 に送られる。そのメッセージの処理結果で D 中のデータが DM により検索・更新・追加・削除される。D の変更に対応して、適当なタイミングで別のクライアント上の AP2 よりメッセージが AP4 に送られ、変更された D を使用してメッセージが処理される。

このモデルは、グループ内あるいは組織内での単純なワークフローの自動化に適する。アプリケーションの例としては、電子メールによる伝票・社内文書の配送と回付およびイベント通知、内部文書のファイリング、業務フローの実行・監視・報告などがある。

3.7 モデル(6) : データ・ステーシング処理モデル

データ・ステーシング処理モデルを図7に示す。

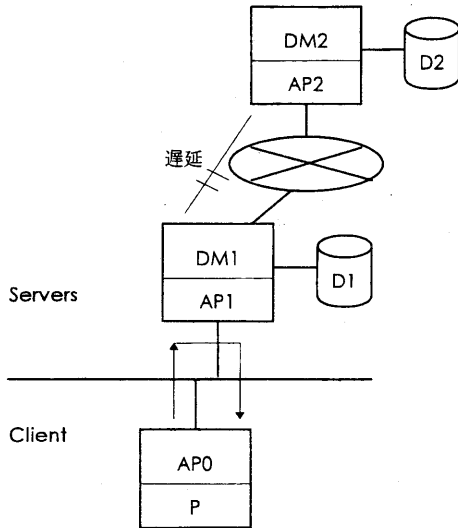


図7 データ・ステーシング処理モデル

メッセージの流れと処理の流れを説明する。クライアント上の AP0 より、メッセージ (SQL 等) が AP1 に送られる。そのメッセージの要求により D1 中のデータが DM1 により検索・更新・追加・削除される。そして AP1 の処理で作成された新しいメッセージが AP2 に送られる。そして処理結果が、AP0 のクライアントに返される。AP1 からのキック等、適当なタイミングで、別のサーバ上の AP2 より、既に送られている AP1 よりメッセージに従って、D2 中のデータが DM2 により検索・更新・追加・削除される。

典型的な利用方法としては、適当なタイミングで、上位のサーバ上の D2 中のデータの一部 (全部) を使用して、下位のサーバ上の D1 中のデータを更新 (ダウンロード) す

る。又は、その逆、D1 中のデータを使用して D2 中のデータを更新 (アップロード) する、アプリケーションである。

このモデルは、データの非同期共有による情報の管理と活用を促進するアプリケーションに適する。アプリケーションの例を次に示す。

- 1) 複製データ (レプリカ) の非同期更新を行うアプリケーション。例えば、複数支店にまたがる人事情報システムや経理システム。
- 2) 業務系データベースの一部のダウンロードによる情報活用。例えば、大福帳や多次元データベースなどによる DSS, EIS。
- 3) 支店や部門別に収集されたトランザクション・データのアップロード。例えば、部門サーバによって収集された受注データのアップロードと企業サーバでの集中受注処理

3.8 モデル(7) : 連鎖型メッセージ処理モデル

連鎖型メッセージ処理モデルを図8に示す。

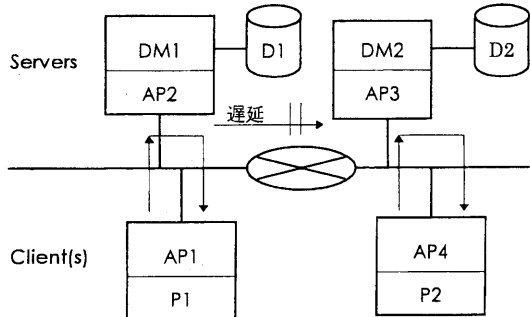


図8 連鎖型メッセージ処理モデル

メッセージの流れと処理の流れを説明する。クライアント上の AP1 より、メッセージ (SQL 等) が AP2 に送られる。そのメッセージの要求により D1 中のデータが DM1 により検索・更新・追加・削除される。そして AP2 の処理で作成された新しいメッセージが AP3 に送られる。そして処理結果が、AP1 のクライアントに返される。AP3 からのキック等、適当なタイミングで、別のクライアント上の AP4 より、メッセージが AP3 に送られ、既に送られている AP2 よりメッセージと併せて、D2 中のデータが DM2 により検索・更新・追加・削除される。そして処理結果が別のクライアント上の AP4 に返される。

このモデルは、独立した複数のアプリケーションやシステム同士の連携による疎な統合を行うのに適する。アプリケーションの例を次に示す。

- 1) グループ間組織間のワークフロー
 - ・電子メールによる関連部門への伝票・社内文書の配布・回付およびイベント通知。
 - ・社内稟議の回付と承認
- 2) 企業内システム間のアプリケーション連携による統合
 - ・メッセージ配送による、生産計画・在庫管理・工程管理などの統合
 - ・受注・仕入・在庫・出荷・請求の各システムの統合。
- 3) EDI などによる企業間のシステム連携
 - ・顧客システムと連携した受発注システム
 - ・仕入れ先システムと連携した部品調達管理
 - ・関連企業のシステムと連携したバック旅行予約

4. モデルの利用

C/SSモデルを設定した目的は、C/Sシステムの処理形態を分類し、各形態にあった構築用の実証済みのソフトウェア・ツールを設定し、トータルなシステム・インテグレーション工数を削減し、安定したシステムを短期間で安く構築することである。このため、図9で示すようなC/Sシステム構築体系としてオープン・ソリューション・フレームワークを作成し、その中核のモデルとした。

4.1 オープン・ソリューション・フレームワーク

C/Sシステムの構築には、図9に示すように大きく3つの工程がある。

- 1) ビジネス戦略計画：情報戦略立案，解決策定義，BPR，プロジェクト定義
- 2) 情報ソリューション計画：要求定義，アーキテクチャ定義，システム基本設計
- 3) ソリューション構築：詳細設計，開発，プラットフォーム/ソフト導入，テスト

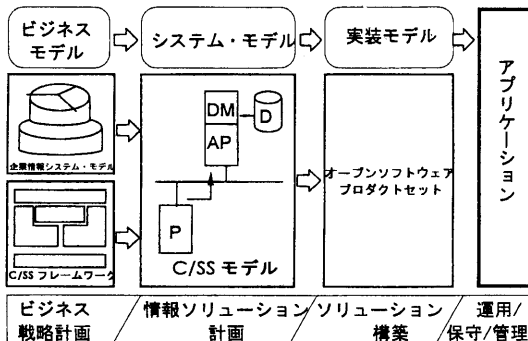


図9 C/Sシステム構築の流れ

- オープン・ソリューション・フレームワークは、上記2) 3) の設計・構築段階のガイドとなるものである。これを参

照することにより、従来多くの時間と労力を費やし、かつ品質面でも問題の多かった C/S システムの構築をより迅速にかつ低コストで、高品質に実現できる。

オープン・ソリューション・フレームワークは、Unix や PC のみならず、エンタープライズ・サーバを含むプラットフォームを使用してクライアント/サーバ・ソリューションを実現するためのガイドライン（構築体系）であり、(1)企業情報システム・モデル、(2)C/SS フレームワーク、(3)C/SS モデルと、それらに基づくプロダクトやサービスなどから構成される。オープン・プロダクトとしては、(4)オープンソフトウェア・プロダクトセットを定義している。これら4つのガイドラインは密接に関連している。企業情報システム・モデルとクライアント/サーバ・ソリューション・フレームワークについては、概要のみを説明する。詳細は別途報告する。

4.2 企業情報システム・モデル

オープン・ソリューション・フレームワークでは、企業の情報システムを、目的、役割、及び取り扱う情報の視点から業務系、情報系、オフィス支援系に大きく分類する⁽¹¹⁾。そしてこれらの系内、系間及び外部間で情報共有と連携を支援するための連携基盤を明確化した。企業の情報システムが連携する外部としては、関連企業、供給業者、調査機関、金融機関、顧客、競争相手が存在している。そして、外部との連携は増大し、ますます重要なものになると考えている。関連を図10に示す。

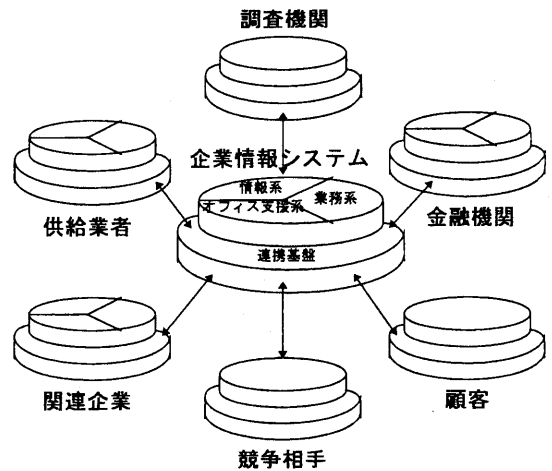


図10 情報システム関連図

企業情報システム・モデルの各系を実現するためには、システムの目的、業種、規模等によって各種の C/SS モデルが組み合わされて使用される。各系で典型的に使用されるモデルを表2に示す。

表2 企業情報システム・モデルと C/SS モデル

企業情報システム モデル	C/SS モデル
業務系	(1)集中トランザクション処理モデル (2)分散トランザクション処理モデル
情報系	(3)リモート・データ・アクセス処理 モデル (4)分散データ・アクセス処理モデル (6)データ・ステージング処理モデル
オフィス支援系	(5)集中メッセージ処理モデル (7)連鎖型メッセージ処理モデル
連携基盤	(6)データ・ステージング処理モデル (7)連鎖型メッセージ処理モデル

4.3 クライアント/サーバ・ソリューション・フレームワーク

クライアント/サーバ・ソリューション (C/SS) フレームワークの目的は、C/S システムの実行時の情報基盤に技術体系を与えることである。C/SS フレームワークでは、ソフトウェアを5つのサービス・セット、開発サービス、クライアント実行サービス、サーバ共用サービス、分散基盤サービスと管理サービス、に分類している。サービスとは、ソフトウェア中に実現される代表的な技術のビルディング・ブロックである。

4.4 オープンソフトウェア・プロダクトセット

C/S システムの実行形態をモデル化した C/SS モデルに従って、C/S システムの開発・実行・管理そして他システムとの接続を支援するためのプロダクトセットを記述する。

プロダクトセットは、7つの C/SS モデルに対応した C/S システムを開発・実行するための7つの基本パターン (図11) と運用管理のための3つの基本パターン (小規模管理, 中規模管理, 大規模管理) および接続のための3つの基本パターン (インターネット接続, オープン接続, メインフレー

ム接続) からなる。現実の C/S システムは、これら基本パターンの1つ以上の組み合わせで開発・実行・管理されそして他システムと接続されている。

それぞれに推奨できるソフトウェアの組み合わせがあり、現在数十種類のセットがある。ここではその全てを紹介できないが、どのようなものを理解して頂くため簡単なセットを紹介する。これらのプロダクトセットの選定にあたっては、一定の選定基準を設け、それに従い十分な評価をして設定している。

4.4.1 トランザクション指向タイプ

C/SS モデルの「モデル(1): 集中トランザクション処理モデル」及び「モデル(2): 分散トランザクション処理モデル」の構築であるトランザクション指向パターンは、アプリケーションの実行がサーバ中心に行われるものとクライアントとサーバ全体として処理を行っていくものとに分類する。

- 1) サーバ中心処理: トランザクション件数が比較的少なくサーバ処理が重いタイプ
- 2) 会話中心処理: トランザクション件数が多く、短い応答が要求されるタイプ (例: Visual Basic+OLTP+ORACLE)

4.4.2 アドホック検索タイプ

必要な検索条件を指定して、サーバのデータベースからデータを表計算のシートに取り込み、データ分析、レポート作成等を行う。アプリケーションの開発を伴わないアドホックな検索タイプである。一般に非定型処理である。
(例: EXCEL+SEQUELINK+ORACLE)

4.4.3 データ・サーバ・タイプ

データベースをアクセスする業務アプリケーションを開発するためのツールとミドルソフトウェアの組み合わせ。以下の3つのタイプに分けてソフトウェアを選択する。一般に定型処理である。

- 1) 規模: 小, 機能: 検索中心 (例: ACCESS+ODBC+ORACLE)
- 2) 規模: 中, 機能: 検索・簡易更新 (例: Visual BASIC+ODBC+ORACLE)
- 3) 規模: 大, 機能: 検索・更新 (例: PowerBuilder+SQL*NET+ORACLE)

4.4.4 グループウェア・タイプ

グループウェア・タイプは、アプリケーションの開発が伴わない電子メール型とアプリケーションの開発が伴う情報共有型とワークフロ

モデル(1) 集中トランザ クション処理	モデル(2) 分散トランザ クション処理	モデル(3) リモート データ アクセス処理	モデル(4) 分散 データ アクセス処理	モデル(5) 集中 メッセージ処理	モデル(7) 連鎖型 メッセージ処理	モデル(6) データ ステージ ング 処理
トランザクション指向		アドホック検索		グループウェア		複製型 データ ステージ ング
		データ・サーバ		メッセージ・キューイング		転送型 データ ステージ ング

図11 構築の基本パターン

一型に分類する。

- 1) 電子メール型 (例: MS-Mail)
- 2) 情報共有型 (例: Lotus Notes)
- 3) ワークフロー型 (例: Staffware)

4.4.5 メッセージ・キューイング・タイプ

分散システムにおいて非同期に処理を連携するとき使用する。

4.4.6 複製型データ・ステーキング・タイプ

複製 (レプリケーション) 機能を利用してデータ・ステーキングを行う。

4.4.7 転送型データ・ステーキング・タイプ

ファイル転送機能を利用してデータ・ステーキングを行う。

4.4.8 管理の基本タイプ

運用管理では、サーバ (S) 数とクライアント (C) 数を目安にシステム規模を大きく3つに分類してプロダクトセットを設定している。

- 1) 小規模管理型: S=1~2, C=30以下, クライアント管理, 印書管理を行う。
- 2) 中規模管理型: S=3~10, C=31~300, 集中管理を行う。
- 3) 大規模管理型: S=11以上, C=301以上の拠点での管理とそれを統合管理する階層型管理を行う。

4.4.9 接続の基本タイプ

接続用のプロダクトセットは、接続の形態により次の三つに分類する。

- 1) インターネット接続型: インターネットを利用して情報を発信, 収集, 交換する。便利なフリーソフトがあるのでこれを利用する。
- 2) オープン接続型: マルチベンダのホスト・コンピュータを接続する TCP/IP 接続やファイル転送と端末エミュレーション機能が一般的である。
- 3) メインフレーム接続型: メインフレームとの接続を高いレベルのプロトコルまでサポートする。

5. おわりに

今回提案している C/SS モデルを含めオープン・ソリューション・フレームワークは、明確なことであるが、全てのアプリケーション領域をカバーしていない。また従来の情報システムで取り扱われ今後も重要であるバッチ処理や大量プリント処理は今回のフレームワークの中では明示的には扱われていない。また、発展途上のオブジェクト指向やエージェント指向、さらにモバイル・コンピューティングについても同様である。もっともクライアントのクライアントとも呼ばれているモバイル・コンピューティングや PDA は、クライアントの一部が移動体 (子クライアント) として動き、適当なタイミングで親のクライアントに統合される。これは

「モデル(6): データ・ステーキング処理モデル」の一種と見なすことが出来るが、エージェント指向等と合わせて考察する必要があるので今後の検討課題とした。

今後、オープンソフトウェア・プロダクトセットの充実に加え、上記の課題に開発方法論、CASE ツール等の取り扱いを含めた形でフレームワークの改定を実施する予定である。

謝辞 オープン・ソリューション・フレームワークは、日本ユニシス株式会社オープン技術部内外の多くの方々との議論の結実である。フレームワークの早い段階より議論に参加して頂いた多くの方々に感謝致します。

参考文献

- 1) SEER Technologies: Client/Server Infrastructure Road Map, Client/Server Journal, ComputerWorld, 1994
- 2) 野村証券が基幹システムを刷新へパソコン1人1台で業務改革に挑戦, 日経コンピュータ, 1995.5.13号, Page53-55
- 3) 伊藤利郎, 坂和磨編著: リエンジニアリングで会社を集合天才に変える本, オーム社, 1995.3
- 4) ドン・タブスコット, アートキャストン著 (野村総合研究所訳): 情報技術革命とリエンジニアリング, 野村総合研究所, 1995.4
- 5) Integrated Information Environment: A Conceptual Overview of the Unisys Architecture (7833 1667-100). Unisys Corporation.
- 6) 手島歩三, 小池俊弘, 遠藤清三: ソフトウェアのダウンサイジング, 日本能率協会マネジメントセミナー, 1994.11
- 7) Open Blueprint Introduction, IBM Corporation, 1994.4
- 8) ISO/IEC JTC1/SC22: Information Technology - Programming Languages, their environments and system software interfaces - Guide to the POSIX Open System Environment(PDTR 14252), 1994
- 9) Alex Berson: Client/Server Architecture, McGraw-Hill, 1992
- 10) J.J.Shedletsky & J.J.Rofrano: Application reference designed for distributed system, IBM Syst.J., Vol.32, No.4, 1993
- 11) 根来龍之, 國領二郎, 永峯弘毅, 繁野高仁, 森澤好臣, 吉岡正: パネルディスカッション-情報技術の戦略的意義, ユニシス・ニュース, 1995年6月1日第410号 (「手島歩三, 小池俊弘, 大熊新仁: やわから情報戦略ブック, オーム社, 1996.3」に全文引用あり)