

## ■5 群 (放送・通信) - 9 編 (ネットワーク管理)

# 2 章 ネットワーク管理アーキテクチャ

(執筆著者：山村哲哉) [2011 年 1 月 受領]

### ■概要■

ネットワーク管理すべき対象や利用する管理技術に対応して、様々なネットワーク管理概念が提案されてきたが、統一的な管理フレームワークをベースに適用技術の違いに応じて適宜カスタマイズし、ネットワーク管理を実現することは効率化の観点で重要である。

OSI システム管理において、管理対象となる構成要素は管理オブジェクトとして定義され、管理システムは管理を行うマネージャと管理されるエージェントから構成される。その間のインタフェースにはオブジェクト指向に基づく共通情報管理サービス/プロトコルが適用され、管理対象オブジェクトの生成・削除・属性取得設定・通知・アクションというシンプルに分類された管理操作によりネットワーク管理が実現される。管理操作を体系化する機能モデルとして、障害管理・構成管理・課金管理・性能管理・セキュリティ管理 (各頭文字を取って FCAPS と呼ぶ) が定義されている。この管理概念を適用すると、原理的に、適用されるネットワーク技術に依存せず同様の作法でネットワーク管理が実現可能である。また、TMN のマルチ機能レイヤ構造においても基本的に違いはなく、機能レイヤごとの管理対象と管理インタフェースが定義されれば、マネージャとエージェント間の管理操作と通知によりネットワーク管理は実現される。上記は集中型管理アーキテクチャベースであるが、ソフトウェア共通バスにより疎結合される分散型管理アーキテクチャも広く研究されており、両アーキテクチャは管理対象種別や実現すべきサービスの条件などから、選択的あるいは補充的に適用されるべきものである。

以上、管理フレームワークの概略を示したが、ネットワーク管理アーキテクチャを構成するためには、管理システム間インタフェース設計において、管理対象をデータで表現するための管理情報モデリングが必要である。管理情報モデリングはシステム構築後の機能追加やソフトウェア流用において重要な役割を果たす。管理システム間インタフェースを規定する方法論は種々あるが、標準的方法としては、適用する管理プロトコル技術に依存しないインタフェース仕様を作成し、これを適用技術に応じて詳細化していく手法であり、統一モデリング言語 (UML) が利用される。本章で述べる管理情報モデルは通信ネットワークのモデル化そのものであるが、ネットワーク機能を管理項目としてどうモデル化するか唯一の正解はなく、ITU-T や TM Forum で規定されている管理情報モデルを機能や仕様に応じて取捨選択・見直しを行うことがモデル化の近道であろう。ネットワークリソースのモデル化のみならず、ビジネス管理/サービス管理を目的とした情報モデルも規定されている。

ネットワーク管理に適用される様々な管理情報の定義は一般にプロバイダやベンダ区々になされており、管理システムやアプリケーション間のインテグレーションやシステム連携は容易ではない。この問題に対処するために、ビジネスプロセスのフレームワーク化やその情報モデル定義に加え、SONET や ATM といった技術に依存しないインタフェース規定、ネットワーク管理システムと上位アプリケーションを繋ぐ java API 規定、再利用可能なサービスをネットワークを介して連携させ大規模システムを構成する SOA アーキテクチャなど、インテグレーションを目的とした新たな取り組みとして検討が進められている。ごく一部である

が、本章の中で紹介させていただく。

**【本章の構成】**

本章では、ネットワーク管理アーキテクチャにおける、管理フレームワーク（2-1 節）、管理情報モデリング（2-2 節）、新たなインテグレーション技術（2-3 節）について解説する。

## ■5 群-9 編-2 章

### 2-1 管理フレームワーク

(執筆者：桐葉佳明) [2008年6月 受領]

ネットワーク管理技術<sup>1)</sup>は1章で述べたように変遷してきた。本節ではネットワーク管理システムを、管理対象や管理プロトコル及び管理機能の配備法の違いから類型化したものとして、OSI システム管理、インターネット管理とテレコミュニケーション管理、サービス管理、自律型分散管理の各々のアーキテクチャとそれを実現するソフトウェア技術について説明する。

管理フレームワークにおける基本原理の多くは OSI システム管理で規定されており、管理対象の違いよりインターネット管理、テレコミュニケーション管理、サービス管理にカスタマイズされ、管理基盤技術としてそれぞれ実用化されている。一方、大規模化・複雑化・多様化が進む情報通信の新しい基盤技術として研究レベルの自律型分散管理を紹介する。

#### 2-1-1 OSI システム管理<sup>3),4)</sup>

OSI (Open Systems Interconnection) システム管理は、開放型システムを運用管理する技術として 1980 年代後半より ISO で仕様策定検討が行われてきたものである。OSI システム管理においては、管理対象となるシステム構成要素を管理オブジェクトと定義し、管理システムは図 2・1 に示すとおり、システム管理を行うマネージャと管理されるシステムであるエージェントから構成される。マネージャがエージェントに対して管理オブジェクトへの管理操作を指示し、エージェントが管理オブジェクトへの管理操作を実行することにより、システム管理の目的を達成する。管理操作は基本的にマネージャから指示されるものであり、一つのマネージャが複数のエージェントへ指示する集中管理アーキテクチャとなっている。

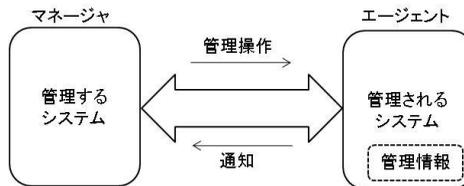


図 2・1 OSI システム管理のモデル

管理対象となる機器・機能を表現する様々な管理オブジェクト群は、総称して管理情報と呼ばれ、オブジェクト指向アプローチにおける継承関係や包含関係を導入し (1-2 節参照)、アクション、通知、振舞いなどの性質をもつアクティブデータとしてモデル化されることが特徴である。管理操作に関しては、管理情報に対するトランザクション型のサービスプロトコルである共通管理情報サービスプロトコル (CMIS/CMIP) が規定されている。マネージャ及びエージェント間で共通管理情報プロトコルを用いることにより、管理オブジェクトの生成、削除、管理オブジェクトのもつ属性の値の獲得・設定、エージェント側からの通知、管理オブジェクトにおいて定義されているアクションの起動、という管理操作を実現する。

また OSI システム管理では、管理操作を体系化するものとして、OSI システム管理機能領

域 (SMFA : Systems Management Functional Areas) と呼ばれる機能モデルを定めている。障害の検出、診断、復旧を行う障害管理 (Fault Management)、システム動作の監視やシステム構成・設定の変更を行う構成管理 (Configuration Management)、利用者ごとの資源利用量を監視し料金計算を行う課金管理 (Accounting Management)、システムの稼働状態を監視し性能監視を行う性能管理 (Performance Management)、システムを安心・安全に稼働させるためのセキュリティ管理 (Security Management) の五つの管理機能が定義されている。

## 2-1-2 インターネット管理とテレコミュニケーション管理<sup>4)</sup>

OSI システム管理の考え方を採用し、IETF においてはインターネット管理に関して、ITU-T においてはテレコミュニケーション管理に関して、技術標準化が行われた。いずれの管理システムも、マネージャ・エージェントから構成される集中管理アーキテクチャに基づき構成され、管理オブジェクトに対する管理操作の指示により運用管理を実現する。

インターネット管理においては、管理情報モデルを規定する仕様として SMI (Structure of Management Information) が策定され、管理情報を操作するためのマネージャ・エージェント間プロトコルとして SNMP (Simple Network Management Protocol) が開発された。SNMP はエージェントとして最小限のソフトウェアで実現可能であることを目的としており、管理操作と応答の種類が限定され、主として管理情報の取得 (Get)、設定 (Set)、及びイベント通知 (Trap) を利用して通信機器の状態管理を行うものである。管理情報を格納したデータベースは MIB (Management Information Base) と呼ばれ、TCP/IP ネットワーク管理の業界標準 MIB-II や RMON/RMON 2 と呼ばれるトラフィックの統計情報 MIB、が標準仕様として用いられている。

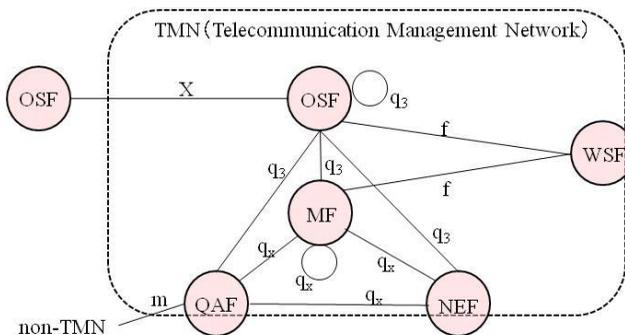


図 2・2 TMN 機能モデル

テレコミュニケーション管理においては、インテリジェント化する通信システムを運用管理するために、通信網オペレーションシステムと通信システムを相互接続する TMN (Telecommunication Management Network) が開発された。TMN では図 2・2 に示す機能モデルを策定し、機能ごとの相互接続関係と対応する接続点 (参照点) を定義している。

図 2・2 において、OSF は通信網オペレーションシステムの機能、NEF はネットワーク機器 (エレメント) の機能、MF は TMN 内のインタフェースを変換する機能、QAF は TMN 外のインタフェースを TMN の標準インタフェースに変換する機能、WSF は TMN を構成する各

機能と操作者とのインタフェース機能を示している。参照点として、TMN 間の相互接続を行う X インタフェースと TMN 内での機能間相互接続を行う Q インタフェースが代表的であり、これを実現する手段として OSI システム管理で標準化された管理情報モデルと共通管理情報プロトコル (CMIP) が採用されている。これを、OSI 管理モデルに沿って表現すると、OSF がマネージャ、NEF がエージェント、Q インタフェースが共通管理プロトコルとなる。

OSI システム管理、インターネット管理、テレコミュニケーション管理の各々の管理フレームワークで共通することは、①マネージャ・エージェントから構成される集中型管理アーキテクチャであること、②管理対象は管理オブジェクトとして抽象化され、これを操作するための管理プロトコルが規定されていること、である。集中型管理フレームワークの実用化・相互接続検証・機能拡張にあたっては、デスクトップや LAN 管理に重点をおく DMTF (Desktop Management Task Force) や主要通信事業者及びベンダから構成されテレコム・情報通信管理領域を対象とする TMF (TeleManagement Forum) が大きな貢献を行ってきた。

### 2-1-3 サービス管理<sup>2), 6), 7)</sup>

1990 年代後半より、通信事業の世界的規制緩和、モバイル・インターネットサービスプロバイダの急増が起り、管理対象となるネットワーク構成や提供サービスが多様化し始めた。このためネットワークシステム管理への機能要求は、事業者・プロバイダに閉じたものから、複数事業者・プロバイダにまたがり管理システムの連携が必須なものとなってきた。更にカスタマ要求やビジネスモデルの変化に迅速に対応するため、運用管理の最適プロセスが常に変化することを前提とした管理システムが要請される時代となった。これを実現するために、既存標準が定めていた管理モデルに加え、管理システムの内部アーキテクチャや管理アプリケーションのコンポーネントにまで踏み込んだ標準化が TMF を中心に進められた。

その一つであるサービス管理は、事業者・プロバイダが利用者へのサービス品質を一定水準以上に保ち、かつ保証するものであり、これを実現するビジネスプロセスや運用管理システムの機能コンポーネントが新しい管理対象となることが特徴である。管理システムにおいても COTS (Commercial Off The Shelf) と呼ばれる市販コンポーネントを導入・活用し、カスタマイズを施すという開発手法が一般的になった。このように、ビジネスプロセスのコンポーネントと COTS に代表されるソフトウェアのコンポーネントを一対一に対応させ、最大限、標準的なソフトウェアコンポーネントの組合せにより管理システムを構築するために、TMF が定めたフレームワークが NGOSS (New Generation Operations Systems and Software) である。図 2-3 に示すコンセプトアーキテクチャにおいては、ビジネスアプリケーションをコンポーネント化したモジュールを共通の通信バスで接続し、その業務プロセスをフローとして管理するためのワークフローエンジンでシステム制御を行うことが特徴である。

NGOSS アーキテクチャの具体的なシステム構築技術として、ソフトウェアのコンポーネント化には EJB (Enterprise JAVA Beans) が、共通の通信バスとして CORBA (Common Object Request Broker Architecture) や EAI (Enterprise Application Integration) が、データモデル・インタフェース定義の COTS として XML (eXtensible Markup Language) などが念頭におかれている。これらの技術を用いて、事業者間にまたがるサービス管理のための情報共有や業務プロセス連携を実現することが可能となる。更には、カスタマ要求やビジネスモデルの変化に対しても、既存コンポーネントの機能拡張や新規コンポーネントの追加、及びワークフロー

の再構成や新規定義，を行うことで柔軟な対応が可能となる。

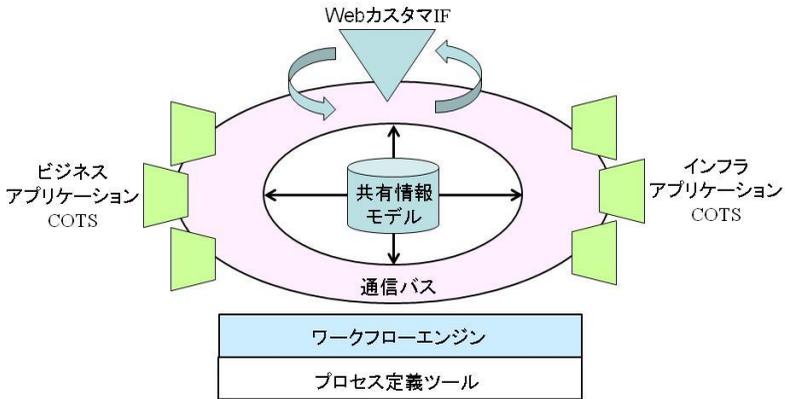


図 2・3 TMF の NGOSS コンセプトアーキテクチャ

サービス管理を実現するための管理フレームワーク及びアーキテクチャは，連携型分散管理アーキテクチャとして，OSI システム管理やインターネット・テレコムネットワーク管理のフレームワークでは下記の点が異なる点が特徴である。一つは，マネージャ・エージェントの階層構造をとることなく機能コンポーネントがソフトウェア共通バスで粗結合されるシステム構成であることである。もう一つは，管理対象が通信機器ハードウェアやソフトウェアから管理アプリケーションサービスに変わり，管理オブジェクトと呼ばれるアクティブデータから管理処理機能を有するコンポーネントプログラムとして抽象化されるようになったことである。集中管理アーキテクチャと連携型分散管理アーキテクチャは二者択一の関係ではなく，管理対象の種別・規模や提供管理サービスの種別・ビジネス形態に応じて選択的・補完的に活用すべきものである。

#### 2-1-4 自律型分散管理アーキテクチャ<sup>5)</sup>

分散オブジェクト指向技術のデファクト標準である CORBA, Java, Web サービスなどの普及にともない，OSI システム管理やテレコミュニケーション管理で規定される負荷分散型の集中管理アーキテクチャから，CORBA, Java, Web サービスで実装されるソフトウェア共通バスを用い負荷分散かつ機能分散を行う分散管理アーキテクチャの研究が盛んに行われた。2-1-3 項で述べたサービス管理への適用は一つの形態であり，分散環境下で動作する様々なサービス管理業務のプロセス群やシステム・ネットワーク管理システム群を統合し，一つのフロースルーオペレーションを実現するために，ワークフロートランザクション処理や異種管理情報データベースの統合が行われた。

一方で，複雑・大規模の一途をたどるインターネットやテレコミュニケーションネットワーク，更にはインターネットアプリケーションを効率的に運用管理するためには，集中型の管理アーキテクチャでは限界があるとして，管理処理機能をネットワーク機器やアプリケーションに組み込んで制御を行う技術が発展してきた。一つはポリシー管理と呼ばれるも

ので、システム・ネットワークの状態に応じてどんな制御をすべきかを定義したポリシーをネットワーク機器やアプリケーションに予め配布しておき、ポリシー制御モジュールが状態監視と制御を自律的に行うアーキテクチャである。特に、IP ネットワークの QoS 管理への適用が実用化されており、IF-THEN 形式で記述されたポリシーに基づき各ルータでのトラフィック制御が自律分散的に行われるようになった。OSI 管理やテレコミュニケーション管理において共通管理情報プロトコルにより管理操作が規定されていた集中管理アーキテクチャに比べると、管理操作がルールもしくはプログラムの形態で管理対象へ配布され、自動的・自律的に管理操作が実行される点が分散管理アーキテクチャの特徴である。

システム・ネットワークの管理処理を行うプログラムが、ネットワーク機器やアプリケーションを移動しながら、管理操作を実行するシステムとして、図 2・4 に示す移動管理エージェントを用いた管理システムや、管理プログラムがマネージャから権限委譲されてエージェントに移動し自律的管理操作を行う管理システム (Management by Delegation) も、自律型分散管理アーキテクチャに分類される方式である。例えば、図 2・4 では、障害発生時に管理コンソールから障害情報収集・分析・診断を行う管理プログラムが起動され、障害原因の原因候補となるネットワーク機器を巡回しながら根本原因の究明を行う自律管理システムを示している。また、OSGi (Open Service Gateway Initiative) が仕様策定を行ったオープンサービスプラットフォームにおいては、バンドルと呼ばれるソフトウェアコンポーネントの配布機能が具備されており、多種多様な機器・通信プロトコルサービスが混在するホームネットワークを柔軟かつ自律的に管理するためのフレームワークが規定され、活用されている。

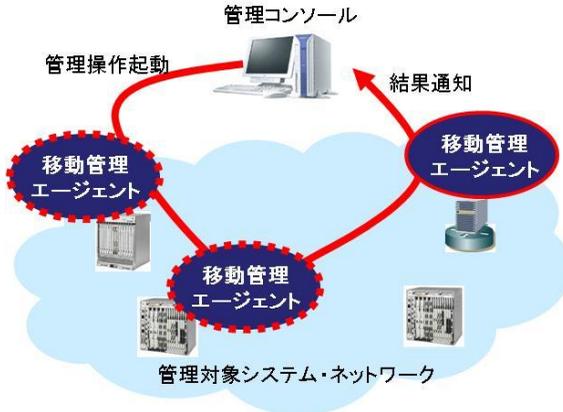


図 2・4 移動エージェントの管理システムへの適用

このように自律型分散管理アーキテクチャは、①複雑・大規模化する分散システム・ネットワーク管理の負荷分散・機能分散を実現すること、②多種多様化するサービス仕様や利用者要求に柔軟に対応可能な拡張・カスタマイズ容易性を実現すること、③複数事業者にまたがって提供されるサービスにおいてもそれらの違いを意識することなく統一的な管理システム連携を実現することが期待されている。

## 2-1-5 ネットワーク管理フレームワークとシステム構築技術<sup>5)</sup>

集中管理アーキテクチャとして OSI システム管理, インターネット管理とテレコミュニケーション管理を, 次いで集中型と分散型が混在するアーキテクチャであるサービス管理を, 最後に自律型分散管理アーキテクチャを紹介してきた. 2-1-3 項で述べたように, 管理対象の種別・規模や提供管理サービスの種別・ビジネス形態に応じて, それぞれの管理アーキテクチャは選択的・補完的に利活用すべきものである.

図 2・5 は集中型及び分散型の管理アーキテクチャの大きな違いを示しており, それぞれの利点を整理する. 集中管理アーキテクチャの利点として, ①管理システムを要求仕様から可変要素なく設計可能であるため特定目的に最適な管理システムが構築できる, ②管理システムの振舞が正確に把握できる, 点があげられる. 一方で分散管理アーキテクチャの利点として, ①負荷分散・機能分散を駆使することにより管理システムの最適化を様々な状況に合わせて実現できる, ②機器管理・ネットワーク管理・サービス管理などの機能要件が異なるシステムの連携を用意に実現できる, 点があげられる.

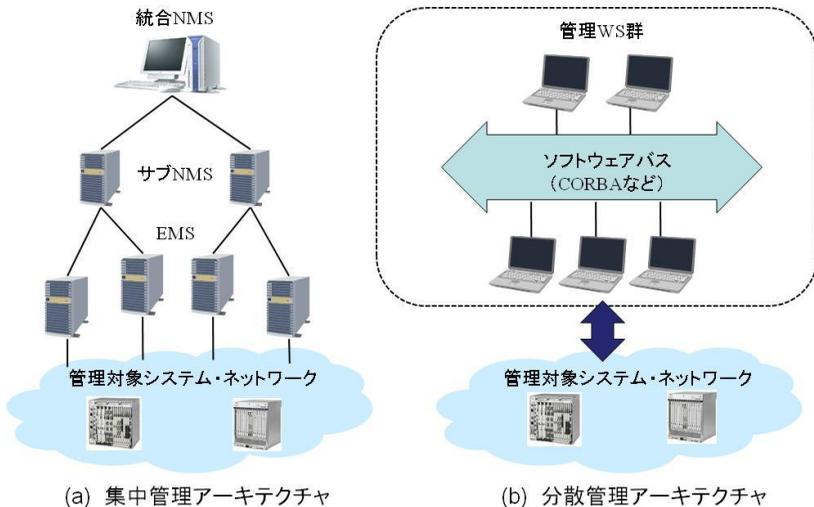


図 2・5 集中管理アーキテクチャと分散管理アーキテクチャ

最後に, これまで説明してきたアーキテクチャに基づく管理システムを構築するための共通機能を管理プラットフォームとして説明する. ここで管理プラットフォームとは, 共通機能をソフトウェアコンポーネントとして実現したものを指し, 各種の管理アプリケーションコンポーネントと組み合わせることで, 管理システムを短期間・低コストで構築することを可能とするものである.

図 2・6 は, 障害発生などのイベント発生を検知・表示し, 関連情報収集を行い, 障害対策の操作を行うなどの一連のプロセスを想定した各種機能コンポーネント群を示しており, これらについて以下に特徴を述べる.

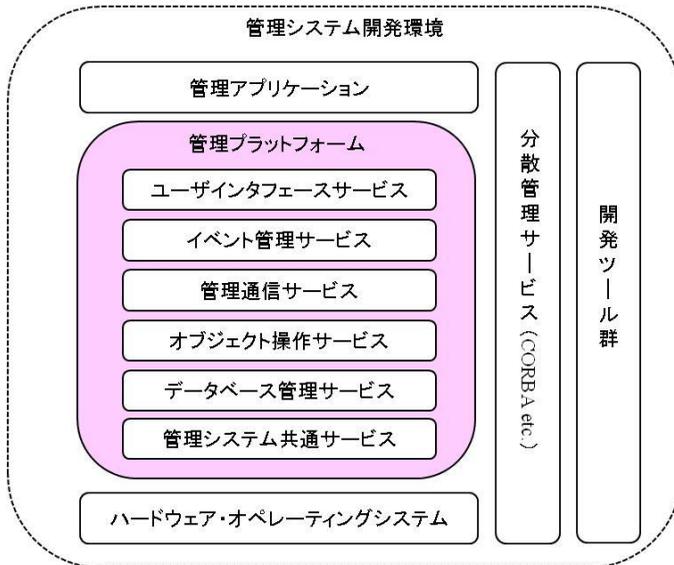


図 2・6 管理プラットフォーム構成

#### (1) ユーザインタフェースサービス

管理操作者に対し管理情報の表示や管理アプリケーションの処理結果を操作端末上に表示する機能を有する。本サービスはグラフィカルインタフェースやコマンドラインインタフェースが含まれ、管理対象の状態や管理操作の処理状況などをアイコンやリスト形式で表示することを可能とする。

#### (2) イベント管理サービス

管理対象となるシステム・ネットワークにおける状態変化や障害検知などのイベントを通知・処理する機能である。管理対象から発生する場合と管理アプリケーションから発生する場合が想定される。共通的なイベント処理機能としては、収集、ロギング、フィルタリング、監視などがあげられる。

#### (3) 管理通信サービス

標準化された管理情報プロトコルである SNMP, CMIP などを実現する機能であり、サービス管理や分散管理においては CORBA や HTTP/SOAP などが利用される場合もある。

#### (4) オブジェクト操作サービス

管理オブジェクトとして抽象化された管理対象を操作する機能であり、システム・ネットワーク機器からアプリケーションまでが対象となる。管理アプリケーションの開発容易性の観点から、管理オブジェクト操作時における位置透過性 (Location Transparency) を提供する機能が重要となる。

#### (5) データベース管理サービス

管理情報を蓄積・検索する機能を提供する。管理情報としては、構成・障害・性能に関連する動的な情報も含まれ、標準化されたデータベースアクセス仕様である SQL (Structured

Query Language) や、コンカレンシ制御などの排他制御機構、データバックアップ機構が具備されることが望ましい。

#### (6) 管理システム共通サービス

管理プラットフォームソフトウェアが動作するハードウェアやオペレーティングシステムに関係するバックグラウンドプロセスの管理や管理アプリケーションの動作状況を管理する機能を提供する。

#### (7) 分散管理サービスと開発ツール

各種サービスモジュールが管理システム内で相互接続するための分散管理サービスとして CORBA, Java, Web サービスなどが、管理アプリケーション開発を容易化するためのツール群として管理情報定義からプログラムテンプレートを自動する GDMO コンパイラや仕様記述からプログラムコードを生成する UML/MDA などがあげられる。

### ■参考文献

- 1) 田中良明 他, “快適コミュニケーションを支える～進化するネットワーク管理技術～,” 電子情報通信学会誌, vol.87, no.12, pp.1003-1042, 2004.
- 2) 高野 誠, 藤本喜作, 江尻正義, “これからのテレコム管理システム開発の課題,” 電子情報通信学会誌, vol.85, no.6, pp.408-413, 2002.
- 3) 江尻正義, 伊勢田衡平, “テレコミュニケーション管理 [I]～テレコミュニケーション管理のコンセプトと情報モデル～,” 電子情報通信学会誌, vol.83, no.8, pp.637-642, 2000.
- 4) 藤井伸朗, “テレコミュニケーション管理 [II]～TMN 統合管理と OSI システム管理～,” 電子情報通信学会誌, vol.83, no.9, pp.707-712, 2000.
- 5) 桐葉佳明, 東 充宏, 長谷川聡, “テレコミュニケーション管理 [III]～管理システム開発とソフトウェア技術～,” 電子情報通信学会誌, vol.83, no.10, pp.776-782, 2000.
- 6) 多田 壽, 吉田健一, 谷口尚己, 林 博之, “テレコミュニケーション管理 [IV]～新しい管理パラダイム～,” 電子情報通信学会誌, vol.83, no.12, pp.927-933, 2000.
- 7) 山中康史, 前川貴夫, “テレコミュニケーション管理 [V・完]～世界の動向～,” 電子情報通信学会誌, vol.84, no.2, pp.135-139, 2001.

## ■5 群-9 編-2 章

### 2-2 管理情報モデリング

(執筆者：吉田龍彦) [2008年7月 受領]

管理システム間のインタフェース設計においては、まずシステム化を要望している人とその要件を確定し、システム開発者がこれを分析した後に設計を行うのが通常の流れである。1 回だけの管理システム開発ならば管理情報モデリングは不要であるが、通常は管理機能の追加が必要になったり、類似の管理システム開発に既存管理システムのソフトウェアを流用する場合があるため、システム間管理インタフェースの開発を効率良く行う管理情報モデリングは重要である。本節では、2-2-1 項で管理インタフェース規定の方法論を述べるが、このなかの分析フェーズで規定される管理情報モデルについて 2-2-2 項で述べる。また、2-2-3 項で管理インタフェース規定方法論の最新動向についても述べることにする。

#### 2-2-1 管理インタフェース規定方法論

管理システム間インタフェースを設計するための方法論としては、当初はその要件とそれに対応するインタフェース仕様を列挙する簡易な方法論しかなかった。その後、要件からインタフェース仕様を導くまでの分析フェーズを重視し、開放型分散処理 (ODP: Open Distributed Processing) ベースの方法論が一時期研究されたが<sup>1)</sup>、現在はオブジェクト分析・設計をベースにした要件/分析/設計の3フェーズ方法論が主流になっている。標準的方法論としては、3GPP のそれも参考にした(2)であるが、これの特徴は、設計フェーズでのインタフェース仕様として SNMP, CMIP, CORBA, XML など、様々な技術があるため、分析フェーズで技術非依存のインタフェース仕様をまず作成し、これを各技術対応のインタフェース仕様にマッピングする手法をとっていることである。表記法としては、主に統一モデリング言語 (UML: Unified Modeling Language) が利用されているが、これは以下のよう

##### (1) 要件フェーズ

システム要求者が決めるビジネス要件とシステム開発者が決める仕様要件に分類され、UML の Use Case やシナリオで記述される。

##### (2) 分析フェーズ

要件フェーズで規定された要件をもとに管理実体及びその関係などを特定するが、これは UML のクラス定義、データ定義、クラス間関係定義、インタアクションダイアグラム (インタアクションダイアグラムまたはコラボレーションダイアグラム)、状態遷移図、アクティビティダイアグラムなどで記述される。なお、(3) 項の仕様記述言語 (SDL: Specification and Description Language) も UML を補完するものとして推奨されている。

##### (3) 設計フェーズ

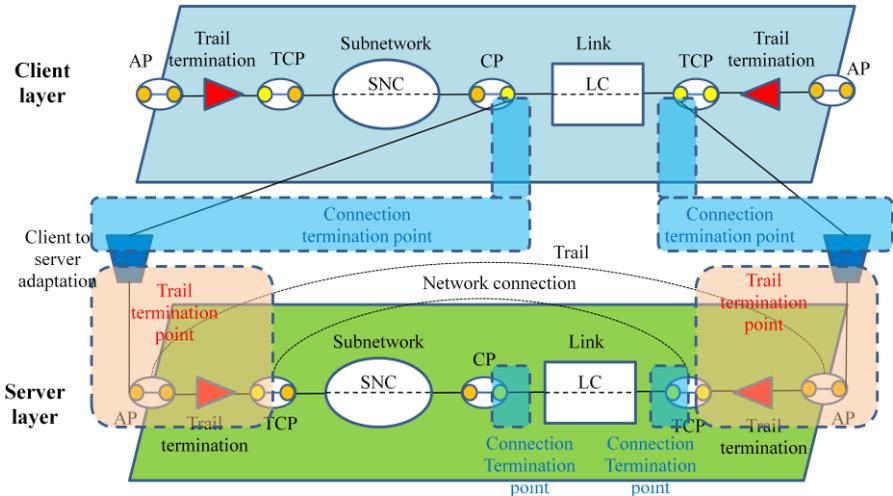
ここでは、3 種類のデータを規定する。すなわち、各インタフェースで交換されるメッセージ、各技術 (SNMP, CMIP, CORBA, XML) 対応のデータ及びエンコーディング方法である。また、そのビヘイビア規定では、要件・分析フェーズでの UML 規定を参照する。具体的には、SNMP の MIB, CMIP での GDMO と ASN.1, CORBA での IDL, SOAP 上の XML など、対応に規定がされる。

## 2-2-2 管理情報モデル

分析フェーズで規定される技術非依存の管理実体及びその関係を一般に管理情報モデルと呼ぶ。下記の(1)，(2)，(3)項では，ITU-T 及び TM Forum で標準化されている管理情報モデルについて述べる。また，(4)項ではこれらの管理情報モデルが他の標準化機関で検討されている管理情報モデルとどのように関係しているかも述べる。

### (1) ネットワークリソース管理情報モデル

ネットワークは一般にコネクションオリエンティッドなネットワークとコネクションレスなネットワークに分類できるが，前者のネットワークモデルは文献 4) で，後者のそれは文献 5) で勧告化されている。これをベースにこれらを管理するための管理情報モデルを規定しているのが文献 6) ある。図 2・7 は，コネクションオリエンティッドなネットワークモデルと管理情報モデルの関係を示している。図に書かれているようなネットワークモデルによると，ネットワークの機能は，各ネットワークのレイヤの Trail と呼ぶパスまたは回線を終端する Trail termination 機能 (赤い三角) と各レイヤ間を結ぶ Adaptation 機能 (青い台形) から構成され，またネットワークのトポロジは Subnetwork や Link で表すことができる。一般に，どの機能を管理項目とし，どのように管理情報モデルを作るかは人によって様々であるが，このようなネットワークモデルで機能を整理すると，管理情報モデルの統一化が図れる。なお文献 6) はこのほかに，装置やソフトウェアの管理情報モデルも規定している。



Copyright ©2008 by ITU-T, Recommendation G.852.2 "Enterprise viewpoint description of transport network resource model"

図 2・7 ネットワークモデルと管理情報モデルの関係

### (2) ビジネス／サービス管理情報モデル

ITU-T 及び TM Forum では，ビジネスプロセスの勧告<sup>7)</sup>をベースにして，マーケット／セールス，プロダクト，カスタマ，サービス，リソース，サプライヤー／パートナー，エン

タプライズ、共通ビジネスなどの各ドメインでの管理情報 SID (Shared Information and Data Model) を規定している。ネットワークリソース管理情報モデルが TMN アーキテクチャの設備管理やネットワーク管理で使用されるのに対し、サービス管理やビジネス管理で使用される管理情報モデルとして、SID のなかの製品とサービスドメインの管理情報モデルが文献 8) として勧告化されている。

### (3) 共通管理サービス管理情報モデル

状態通知、故障通知、ログ、性能監視など、管理システムで共通に使用する管理サービスのモデル化については、従来は CMIP 版が X.700 シリーズで勧告化されており、これが参考になるが、現在はこれをベースにした技術非依存管理情報モデルが作成されつつある。

### (4) 共有情報アーキテクチャ

以上、管理情報モデルを概説したが、実際にはこれ以外にも管理情報モデルは様々なところで検討されている。文献 9) から引用した下記の図 2・8 は、他の標準化団体なども含めた管理情報モデルの一覧を示しているが、いままで述べた管理情報モデルは、2 段目の“Specialized Information Models”の ITU-T や TM Forum に相当し、ITU-T や TM Forum 以外にも様々な標準化団体で管理情報モデルを規定していることが分かる。このため、最上位の“Generic Information Models”の必要性もいわれている。一方、最下位の“Specialized Data Models”が設計フェーズの各技術対応の規定であり、データモデルと呼ばれる。



Copyright C2008 by ITU-T

図 2・8 共有情報アーキテクチャ

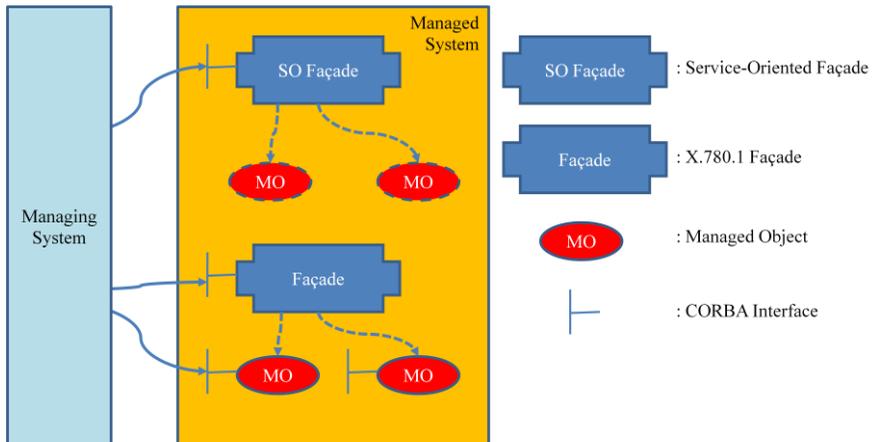
## 2-2-3 管理インタフェース規定方法論の最新動向

以上、管理インタフェース仕様を決めるための従来の方法論を述べたが、最近では SOA (Service Oriented Architecture) / BPM (Business Process Management) も取り入れようとしている。本節では、これらの動向について説明する。

### (1) SOA

“SOA の現状と展望”<sup>10)</sup>によれば、SOA とは『サービスと呼ぶ機能単位をネットワークを介して連携させ、業務全体を実現するアーキテクチャ』といわれており、異システム-プロセス間にサービスという抽象度の高いインタフェースを定義するため、その連携がスムーズに行える。TM Forum の MTNM (Multi-Technology Network Management) は、設計フェーズにおける CORBA 版 SOA 管理インタフェースを規定しており、ITU-T 勧告にもなった<sup>11), 12), 13)</sup>。

図 2・9 の下段に示すように、ITU-T では、従来は終端点のようなネットワークリソース (MO) 対応に管理インタフェース (MO 直接または Façade) を規定してきたが、MTNM では上段に示すようにサービス指向インタフェース (SO Façade) を新たに規定している。例えば、性能管理関係の MO へアクセスする PM Manager などが規定されている。一般に、ネットワークプロバイダは、各種ネットワーク技術で構成されるネットワークを複数のベンダで構築する。また、ネットワーク技術及び新規サービスが次々と出てくるため、それにも対応したネットワーク管理を迅速に実現しなければならない。従来のように個々の技術対応の管理インタフェースでネットワーク管理システムを開発する場合、管理インタフェースごとに開発コストや開発時間がかかるが、MTNM のような抽象度の高い SOA 管理インタフェースを採用すると、各種装置を統一的に管理可能になるため、開発コストの低減が図れる。



Copyright C2008 by ITU-T, Recommendation X.780.2

図 2・9 従来の管理インタフェースと SOA 管理インタフェース

## (2) BPM

“BPM がビジネスを変える”<sup>14)</sup>によれば、BPM とは『業務の流れを整理・分析して、より効率的・効果的に仕事ができる方法を継続的に工夫・改善すること』といわれている。従来の管理インタフェース仕様方法論では、管理されるリソースモデルと管理機能をベースに、管理システム間のインタフェースを規定してきたが、最近ではこの BPM を実現するためにビジネスプロセスレベルでの管理インタフェース規定方法論も研究されつつある。TM Forum の NGOSS がその一例といえる。

### ■参考文献

- 1) ITU-T Recommendation G.851.1, “Management of the transport network - Application of the RM-ODP framework,” 1996.
- 2) ITU-T Recommendation M.3020, “Management interface specification methodology,” 2008.
- 3) ITU-T Recommendation Z.100, “Specification and Description Language (SDL),” 2007.
- 4) ITU-T Recommendation G.805, “Generic functional architecture of transport networks,” 2000.
- 5) ITU-T Recommendation G.809, “Functional architecture of connectionless layer networks,” 2003.
- 6) ITU-T Recommendation M.3160, “Generic Management Information Model - Protocol Neutral,” 2008.
- 7) ITU-T Recommendation M.3050-series, “Enhanced Telecom Operations Map (eTOM),” 2007.
- 8) ITU-T Recommendation M.3190, “Shared Information and Data Model (SID),” 2008.
- 9) ITU-T, “NGN Management Specification Roadmap,” 2008
- 10) 野村佳秀, 原裕貴, “SOA の現状と展望,” 電子情報通信学会誌, vol.89 no.6, 2006.
- 11) ITU-T Recommendation M.3170-series, “Multi-technology network management,” 2007.
- 12) ITU-T Recommendation X.780.2, “TMN guidelines for defining service-oriented CORBA managed objects and façade objects,” 2007.
- 13) ITU-T Recommendation Q.816.2, “CORBA-based TMN services: Extensions to support service-oriented interfaces,” 2007.
- 14) 日沖博道:“BPM がビジネスを変える,” pp.69-75, 日経 BP 企画, 2006.

## ■5 群-9 編-2 章

### 2-3 新たなインテグレーション技術

(執筆者：高木孝之) [2008年9月 受領]

サービスプロバイダの OSS/BSS (Operations Support Systems / Business Support Systems) を構成するソフトウェアやシステムの相互運用性の課題に対し、標準化という観点から、ITU-T や ETSI/TISPAN (European Telecommunications Standards Institute / Telecoms & Internet converged Services & Protocols for Advanced Networks) など他の主要な標準化団体とも連携しながら最も包括的・体系的に取り組んできたのが 1988 年に設立された TM Forum (設立時名称 OSI/Network Management Forum) である。本節では、TM Forum の策定した業界標準 NGOSS (New Generation Operations Systems and Software) にフォーカスを当て、その実装のための各種インテグレーション技術を解説する。

#### 2-3-1 NGOSS 策定の背景

デジタル交換機が登場してもしばらく 1980 年代の半ば頃までは現在のような OSS/BSS という明確な概念は存在しなかった。加入者管理や料金管理は、情報システムの一部として構築され、帳票・電子帳票でバッチ処理された。また、オーダは個別の専用端末から交換機そのものに投入するようなかたちであった。網やサービスの種類が限定され、事業者間競争も少なかった当時においては、こうした仕組みで成立していた。

しかし、現代のサービスプロバイダは、デジタル網・移動体網・インターネットの普及、サービスの細分・多様化、規制緩和による競争激化と買収・合併・提携、更には通信/放送の融合など、1980 年代初頭には考えられなかった厳しい環境に置かれている。必然、サービス提供にかかわるシステム全体をよりビジネス寄りの視点で管理できる“統合型”の OSS/BSS が必要となる。また、品質担保した競争力ある新サービスをいち早く、コスト効率良く開発・統合・展開・管理していくため、プロセスの自動化と相互運用性も要求される。TM Forum はその実現のための業界標準フレームワークとして 2001 年に「NGOSS リリース 1」を策定、その後も更改を続け、2008 年 8 月にはリリース 7.5 としている。また、課題を克服して End-to-End に効率化されたエコシステムとバリューチェーンを実現した理想的なサービスプロバイダのあり方を“Lean Operator” (Lean: 贅肉をそぎ落とした、企業など無駄なく活力のある) と表現する。

#### 2-3-2 eTOM/SID の概要

NGOSS は、OSS/BSS の構築を Business (ビジネス)、System (システム設計)、Implementation (実装)、Deployment (商用展開) の四つのビューで見ると一連の「ライフサイクル」として捉え、ビジネス領域・技術領域の双方をカバーする業界仕様と、コンプライアンスのガイドラインやテスト基準を一連のツールキットとして規定、確立された再利用可能なノウハウとして提供している。NGOSS はビジネスプロセスフレームワーク (eTOM)、情報フレームワーク (SID)、システムインテグレーションフレームワーク (TNA)、アプリケーションフレームワーク (TAM) の四つのコンポーネントで構成されているが (1 章 1-3 節参照)、本項では特にビジネスプロセスの概念上重要な eTOM と SID をとりあげる。

## (1) ビジネスプロセスフレームワーク (eTOM)

eTOM (Enhanced Telecom Operation Map) は、サービスプロバイダの運用管理に必要な主要なビジネスプロセスに関するプロセスマップと共通言語、プロセスフローをフレームワークとして規定したものである。もともと 1998 年にネットワーク設計、建設、サービスオーダ、開通、故障受付など、通信事業者の基幹業務を対象にした TOM がリリースされたが、その後、企業としての経営管理や戦略策定面にまでプロセスの範囲を Enhance して eTOM となった。eTOM はサービスプロバイダの既存プロセスの見直しに利用できるほか、更改に向け SI 事業者との対話においても相互理解を助ける効果がある。eTOM は 2004 年に ITU-T 勧告 M.3050 にとり入れられ、正式な国際標準ともなっている。

eTOM の Business View は、レベル 0 からレベル 3 の 4 階層の細分化レベルを規定しており、レベル 0 が最も抽象度が高いが、ここではレベル 1 のマップを紹介する (図 2・10)。“Strategy, Infrastructure & Product” が縦三つのプロセスに、“Operations” が縦四つのプロセスに分類され、それぞれ事業者側～顧客側のポジショニングを示す横軸に該当する管理要素が示されている。

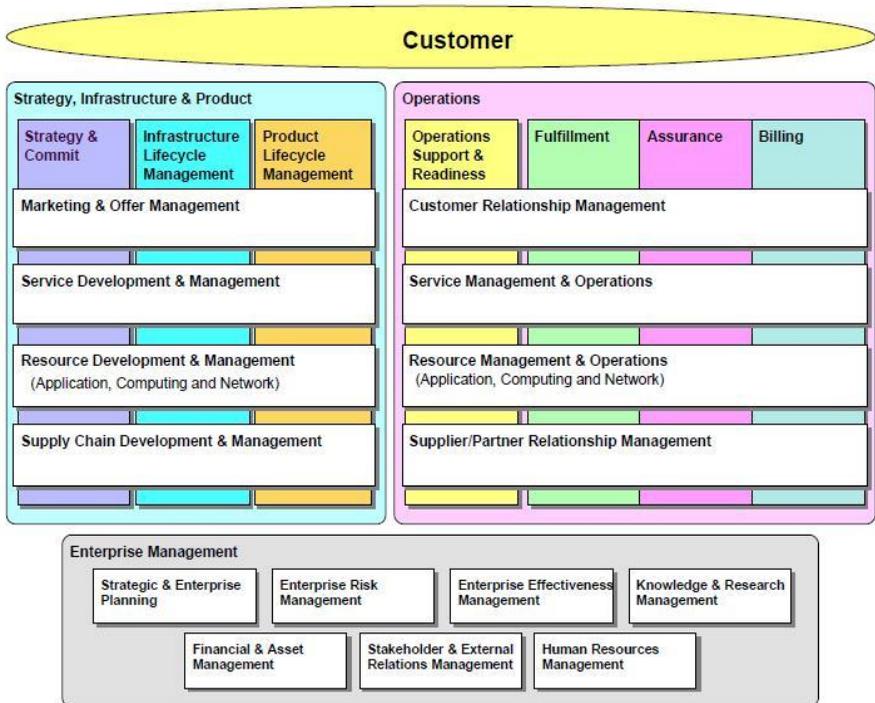


図 2・10 eTOM Release 7.5 - Level 1 Processes (出典：TM Forum GB921 v7.3)

また、近年、eTOM は広く IT サービスの領域や一般的なサービス産業にも適用できる汎用性をもっているのではないかと、eTOM の成果を他業界に反映させること、あるいは逆に

他のサービス業でのビジネスプロセス検討の成果を eTOM に取り込むことが広い範囲でのビジネスプロセスの共有化を促し、IP/e ビジネスのシームレスな管理や効率の良い管理システムの構築を促進することになるとの期待もあり、CobiT (Control Objectives for Information and related Technology) や CMMI (Capability Maturity Model Integration) など、ビジネスプロセスや管理のフレームワークを推進する他団体とも共同検討の試みが始まっている。また、情報システムの管理基準として 1989 年に英国中央コンピュータ・電気通信局 (CCTA) が初版を発行した ITIL (Information Technology Infrastructure Library) は、IT 世界のビジネスプロセスフレームワークの規定として著名であるが、eTOM とも協調関係にある。TM Forum は 2003 年に eTOM/ITIL Working Party を立ち上げ、ITIL を推進する itSMF (IT Service Management Forum) と両者の対照や協調関係を検討、TR 143 “Building Bridges: ITIL and eTOM” というドキュメントに成果をまとめている。

## (2) 情報フレームワーク (SID)

OSS/BSS の中で使われる Product, Customer Bill, Service Usage, Resource などの用語やデータ定義に、サービスプロバイダやシステムインテグレータ、ソフトウェアベンダ、ネットワーク機器ベンダの間で統一性が図られていない状況では、OSS/BSS を構成するシステムやアプリケーション間のインテグレーションや事業者間のシステム連携において多大な支障が出ることは想像に難くないであろう。TM Forum は、ビジネス主導の視点からそのデータ定義を行い、単一の情報モデル SID (Shared Information and Data Model) として提供している。

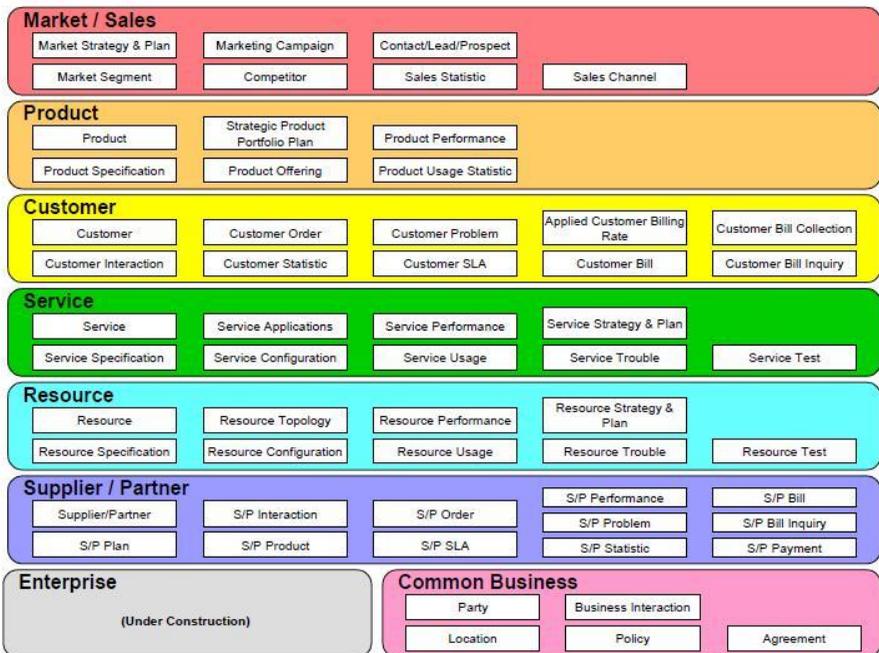


図 2・11 SID Release 7.5 - Business View Framework (出典：TM Forum GB 922 v7.6)

SID は企業全体の情報体系フレームワークで、Business View (図 2・11) では eTOM にてらしてサービスプロバイダで扱われ、共有される管理情報体系を、System View ではそれを実現する UML class モデルを、Implementation View では CORBA など特定プラットフォーム依存の実装法を提供している。2008 年 8 月、SID が国際標準化機関 ITU-T の情報管理モデルとして採用され、ITU-T 勧告 M.3190 として参照されることとなった。

### 2-3-3 TM Forum インタフェースプログラム

TM Forum は、NGOSS に準拠しながら OSS/OSS のソフトウェア開発成果物に相互連動性をもたせ、また再利用可能ならしめるため、TM Forum Interface Program として幾つかのインタフェース群を規定している。

#### (1) MTNM/MTOSI

ITU-T は 2001 年 1 月に CORBA 技術に基づく TMN サービス (Q.816)、同年 8 月に CORBA 技術に基づく TMN サービスの粗粒度インタフェース拡張 (Q.816.1) を勧告化した。同 8 月、TM Forum では MTNM (Multi-Technology Network Management) の初版がリリースされたが、これは、SONET (Synchronous Optical Network) や ATM (Asynchronous Transfer Mode) などの伝送装置の管理のための CORBA ベースのインタフェースである。また、ベンダやネットワークプロトコル非依存を特徴とする CORBA は昨今の光通信分野でも注目され、光通信ネットワークに関するシステムの経済化を推進する業界団体 FSAN (Full Service Access Network) から ITU-T への提言がなされ、UML インタフェース要求条件に基づく B-PON (Broadband Passive Optical Networks) の NMS-EMS 間の CORBA インタフェース (Q.834.4) が 2003 年に規定されている。

一方、2005 年に TM Forum にて初版リリースとなった MTOSI (Multi-Technology Operations System Interface) は、異なるベンダの提供する OSS 同士を結ぶ XML (eXtensible Markup Language) ベースのインタフェースである。この二つの標準は、現在 mTOP (multi-Technology OSS Program) という一つの傘に取められ、それぞれ mTOP-MTNM、mTOP-MTOSI と呼ばれている。

#### (2) OSS/J

OSS through Java. EML/NML と上位アプリケーション間の標準インタフェースを策定するため、2000 年 9 月に米 Sun Microsystems 社が発起人となり、SI 事業者やソフトベンダ、ネットワーク機器ベンダなどと立ち上げた業界イニシャティブ。設立当初より TM Forum の NGOSS (eTOM/SID) の「実装技術」の提供に注力してきた。Trouble Ticket, Inventory, Fault Management, Order Management など、OSS/BSS に特化したインテグレーション用の API を Java で標準化している点の特徴である。Java EE (J2EE)、XML など業界で十分に普及した技術に立脚しているため、リスクヘッジやコスト削減が期待できる。Web Services でのインテグレーションプロファイルももっているため、後述の「SOA ベースの OSS」を実現する実装技術の選択肢ともなる。2006 年 5 月、OSS/J は TM Forum に取り込まれた。

### (3) IPDR

IPDR は Internet Protocol Detail Record の略。1999 年 7 月に設立された業界団体 IPDR.org が策定した IP ネットワーク上の各種サービスの使用量などを OSS/BSS に提供するデータ形式仕様。IPDR.org は 2007 年に TM Forum に吸収された。

### (4) CO-OP

Co-operative Open Operation Support System Project. モバイルネットワークにおけるマルチベンダのネットワーク機器とその管理システムのインテグレーションを容易にするため、2004 年 11 月に TM Forum 内で設立されたプロジェクト。メンバは、アーキテクチャ仕様の作成、テスト、検証を共同で行う。

## 2-3-4 NGN と SOA ベースの OSS

NGN (Next Generation Network) の標準化は、2003 年 7 月に ITU-T がジュネーブで NGN ワークショップを開催して以来、ITU-T と ETSI のプロジェクト TISPAN の双方で並行して進められていたが、ITU-T にやや先立ち、TISPAN が 2006 年 2 月、約 60 の技術仕様を NGN リリース 1 として承認・完成させた。特に TISPAN のワーキンググループ WG 8 が NGN のネットワーク管理の仕様書群を固めたが、これが TM Forum の eTOM を参照し、また、SOA (Service Oriented Architecture) 技術に基づいている点は注目に価する。

SOA は、大規模なシステムを再利用可能な「サービス」の集まりとして構築する設計手法である。業務上の一処理に相当するソフトウェアの機能をサービスと見立て、そのサービスをネットワーク上で連携させてシステムの全体を構築していく。SOA = Web サービスではないが、SOA 実現の具体的な技術基盤として Web Services が事実上のデファクト技術となっていることは否定できない。ソフトウェアを Web Service 化し、各サービスが XML で記述されたメッセージを SOAP (Simple Object Access Protocol) でやり取りし、連携して動作する。

先述のとおり、昨今のサービスプロバイダには、競争力ある多様なサービスを次々と生み出し、それを管理していく OSS/BSS の仕組みが要求されている。しかし、オーダ処理、リソース割当て、料金請求などは、複数のサービスに共通するプロセスである。そこで、eTOM のようなビジネスプロセスの考え方を導入して各ビジネスプロセスを Web Service 化し、複数のサービスで使い回すことが効率的である。NGN 時代の OSS/BSS には、ビジネスプロセス概念と SOA によるインテグレーションが有望視されている。

## 2-3-5 NGOSS の実装を促進する prosspero

TM Forum の NGOSS はもともとシーズ志向で作成されてきたため、既存のシステムを気にせずこれを取り込める新興事業者での実績や、一部の大手事業者での NGOSS 準拠インタフェースの採用実績があるものの、まだ参照レベルの利用の仕方をしている事業者が多く、NGOSS のフルアーキテクチャをベースに OSS/BSS 全体にわたるエコシステムを構築している例はまだほとんどない。これを改善すべく、市場での NGOSS アーキテクチャの採用と NGOSS 準拠製品の普及浸透を図る目的で 2006 年 5 月発表された TM Forum のイニシアティブが prosspero である。prosspero はソフトウェア製品の準拠認定システムによるガバナンスを一つの重要な柱としている。準拠製品のほか、トレーニング、ユースケース、ケーススタ

ディ、API のソースコード、オンラインサポートコミュニティを含む、実装のためのワンス トップショップを提供している。

一方、OSS/J は、もともと TM Forum の NGOSS (eTOM/SID) の実装技術を提供することにフォーカスしていた OSS/J は、API とそのサンプル実装、ユースケースなどの関連ドキュメントの提供、準拠製品認定制度、大手サービスプロバイダでの導入実績などの点で *prosspero* のコンセプトに合致する諸条件を備えていた。そこで、TM Forum は OSS/J Initiative の吸収と同時に *prosspero* の立ち上げを発表した。かくして、OSS/J は *prosspero* の最初の実装技術の柱となった。無論、OSS/J 以外に優れた技術選択肢が登場すれば *prosspero* に追加されていくことになるだろう。

### 2-3-6 Service Delivery Framework

冒頭で述べたような厳しい競争環境に置かれているサービスプロバイダは、差別化サービスをいち早くローコストに開発・配備・運用すること、また、新規及び既存のサービス（サービスコンポーネント）を柔軟にバンドルして提供できる能力、また、それをライフサイクル視点で管理する能力が求められているが、これを実現する一元的プラットフォームが一般に SDP (Service Delivery Platform) として抽象概念化されている。Time-to-Market とコストを意識して、SDP の実装には汎用的な IT 技術の適用が期待されているが、これは、通信史上 OSS/BSS 領域のみならずサービスそのものの作成・実行の領域にも IT 技術が浸透してきていることを意味するものである。

いまや SDP は通信業界の最も重要なキーワードの一つであることに疑いの余地はないが、TM Forum もこの動向に注目し、サービスのデリバリ管理における業界標準を策定すべく、SDF (Service Delivery Framework) と SDF Reference Model の策定に着手している。本節では詳細は省略させていただくが、同 Forum の提供する“Service Delivery Framework Overview” (TR 139)、“Service Delivery Framework Business Agreement” (TMF 519) などのドキュメントを参照されたい。

#### ■参考文献

- 1) 多田 壽, “ビジネスプロセスの変革へ向けた TM Forum の最近の活動-Prosspero や TAM などの新しい動き,” 電子情報通信学会技術研究報告, vol.107, no.135, pp.25-30, 2007.
- 2) 北井 敦, 山下陽一, 榎 一光, “テレマネジメントフォーラムの最新動向,” 電子情報通信学会技術研究報告, vol.105, no.163, pp.19-24, 2005.
- 3) 柳田 誠, 高木博文, 山下陽一, 高野 誠, “TMF 規定の社内情報システム開発標準適用に関する一考察,” 電子情報通信学会技術研究報告, vol.106, no.462, pp.55-60, 2007.
- 4) 高木孝之, “SDP (Service Delivery Platform) の最新動向とそれを支える Java テクノロジー,” 電子情報通信学会技術研究報告, vol.107, no.313, pp.41-46, 2007.
- 5) 高木孝之, “NGN+S Review vol.1,” pp.50-55, リックテレコム社, 2006.
- 6) TM Forum Prosspero Initiative, <http://www.prosspero.com/>
- 7) TM Forum: TR139, “Service Delivery Framework Overview Release 2.0,” 2008.