

## 3V-4 インテリジェントネットワークにおける ユーザ志向型サービス制御法の提案

水澤 純一 坂井 博 秋山 稔  
東京大学工学部

### 1. はじめに

高度な通信サービスの提供を目指して提唱されたインテリジェントネットワーク(IN)技術は、わが国でもフリーダイヤルなどで一部開発が進められているが、それらはデータベース参照型の比較的単純な制御の域を越えていないのが現状である。さらに柔軟なINへの動きは、すでに各方面において進められており、Bell coreによって提唱されたAIN(Advanced IN)などが現在のINのトレンドとなっているが、まだ多くの問題が残されているのが現状である。

本稿ではAINに残された問題点を指摘し、次世代INの制御技術として、ユーザ志向型のサービス制御手法の提案を行う。

### 2. AINの制御技術とその問題点

AINは、仮想プロセッサを持ちネットワーク制御の頭脳として動作するSCP(Service Control Point)、およびSCPが送るサービスシナリオに従ってハードウェア制御を行うSSP(Service Switching Point)から構成される(図1)。

サービスシナリオは、サービス全体の流れを記述したNシナリオ、分岐の条件等を詳述したSシナリオ、さらに実際にハードウェアを起動するためのAシナリオに分類される。

AINではこれらの3つのシナリオと、Nシナリオで起動する各種機能部品(FO:Functional Object)はすべてSCPに用意され、発呼があるとユーザのデータを参照しながら、要求されたサービスに必要なシナリオを選択してSSPへとダウンロードする。SSPでは送られてきたシナリオに従ってサービスを実行する。

この方式においては、次に挙げる問題が残されている。

(1) 全ユーザの細かな要求をデータとして一括して保有すると、必要なデータ量が増大する。またそのデータの変更が頻繁に行われることを想定すると、ユーザとネットワーク間にその変更のためのトラヒックが増大し、ユーザカスタマイズが困難である。

(2) ユーザのデータから必要なシナリオのセットを選択するアルゴリズムは、かなり複雑になることが予想される。

(3) ユーザの要求が多様化するに従って、S、N両シナリオが肥大し、状態の爆発がおきることが予想される。

(4) サービス数が増えてくると、新しいサービスのシナリオをS、Nシナリオに追加しようとする場合、従来のシナリオとの間に矛盾が生じることが考えられるが(サービス干渉)、そのために新サービスの導入ができなかったり、仕様の変更を行うことはあまり望ましくない。

(5) 多数の呼が同時に発生した場合、SCP~SSP間のダウンロードの負荷が増大する。

(6) 全ユーザのシナリオを一括して管理する場合、あるユーザが変更した仕様になんらかの不都合があった場合、その影響が全ユーザに及ぶ。

(7) SCPがダウンするとシステム全体が機能しなくなってしまう。

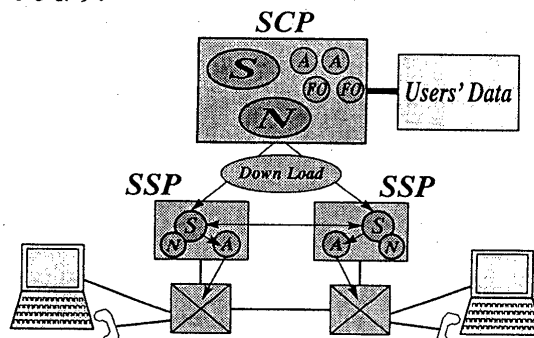


Figure 1 The ordinary IN call control

### 3. ユーザ志向型制御方式のコンセプト

通信サービスが、平面的に分散しているネットワークリソースにアクセスすることによって実現されるということを考えて、交換制御のソフトウェア機能やデータは、元来それらが必要とされるハードウェアのすぐ近くに分散して置かれるのが理想である。さらに、近年における汎用計算機の高性能化、ATM導入に伴う共通線信号リンク容量の向上を考えると、ネットワーク制御は各ノードに分散したオブジェクトにおける処理と、処理の継承・依頼を伝えるオブジェクト間通信によって実現できると考える。

ここでいうオブジェクトは、具体的には各シナリオの処理を管理する核となるオブジェクトと、サービスの共通の機能部品である複数の機能オブジェクト(FO)を指す。

機能オブジェクトについては、その機能が使われる頻度等の条件の変化に従って最適なノードへと配置されることが望ましい。一方、シナリオに対応する核オブジェクトについては、前述の問題点を考慮すると、全ユーザを一括管理するのではなく、あらかじめユーザ毎にシナリオを分割し、Nシナリオについては加入者が収容されているSSPに、Sシナリオは各ユーザの端

末（パソコン電話などの高性能端末）に、またAシナリオについてもできる限りハードウェアに近いところにおくことが望ましいと考える。これに伴って、各ユーザのデータについても課金などの重要なものを除いてユーザ端末へと移動させる（図2）。

この考えをさらに進め、各呼に関する処理の管理をユーザの核オブジェクトに担当させる。各核オブジェクトの並列処理はOSのマルチタスク機能を利用する。

ユーザは自身の端末において、必要なデータを変更することで容易にサービスをカスタマイズすることが可能である。交換機の処理についても、発呼時に利用ユーザのシナリオを起動することに単純化できる。

また加入者が利用可能な全サービスに加入することは考えにくい。各ユーザのシナリオ内における状態の爆発を防ぐことが可能である。さらに、各核オブジェクトの処理は全く独立に行われるため、ある呼における事故の影響は他のユーザには及ばない。

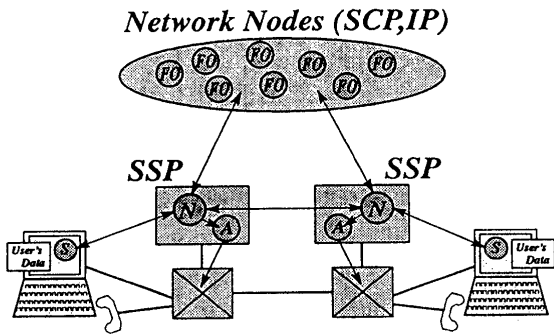


Figure 2 The user oriented IN call control method

基本的なシナリオはRBOCが用意するものとする。この場合、大部分のユーザは提供されるシナリオをそのまま利用することが予想されるため、これらについてはシナリオの実体自身は一つだけ用意し、発呼時にそのシナリオをユーザのシナリオにコピーして用いる。

4. サービスシナリオの記述

IN環境においては、複数のベンダによってサービスが開発される。従って、各ベンダおよびユーザにおけるサービスの開発に際しては、他のベンダやユーザとの間の切り口をある程度統一する必要があると考える。また仮に特別なオーダを必要とする場合は、例えば企業内の内線番号サービスを東京と大阪で使用する場合などのように、相手はそのオーダを理解できる場合はよいが、それ以外のユーザとの通信の事態を予測して、相手がオーダを理解できなかった場合の処理を記述する必要がある。Nシナリオ内の分岐の処理は、すべてユーザの端末内に配置されたSシナリオが返すオーダの種類によって行われる。

これらのシナリオの記述は、ユーザの要求に応じて各ベンダが提供することになるが、各ユーザがすべての利用可能サービスに加入することは考えにくく、加入したサービス内でのユーザの仕様に対する希望は、SCPが全利用サービスのシナリオを一括して保有した場合と比べかなり反映されることになる（図3）。

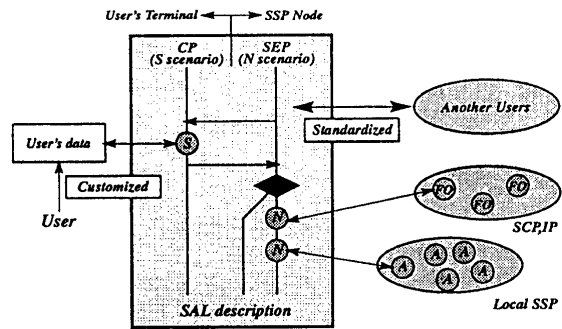


Figure 3 User's scenarios by SAL

5. プロトタイピングシステムTIPSの試作

前述の概念に基づき、INプロトタイプシステムTIPS（Tokyo university IN Prototyping System）を試作した。

TIPSはLANで接続された3台のWS（Sparc Station1）と2台の交換機、ISDN端末から構成される（図4）。

WS1がINのSMS（Service Management System）およびSCPに、WS2,3がSSPにそれぞれ対応する。

仕様記述言語には、SDE（Systems Design Environment）で提案されているSAL（Service Addition Language）を利用し、端末にはICカードの読み書きが可能なCanonのNAVIを利用した。

ユーザのSシナリオおよびデータについては、WS内に疑似的にユーザ用の領域を設けてそこに収容している。

現在のところ、基本接続、転送電話サービス、着信リダイヤルサービス（留守中の呼に対して、着信者が特番で順に発信者を呼びかえす）等をインプリメントし、適宜ユーザのシナリオに登録して実行してみた。

同時発呼数が2の場合について、発呼者のダイヤル終了後、相手のベルがなるまで約0.5秒であった。

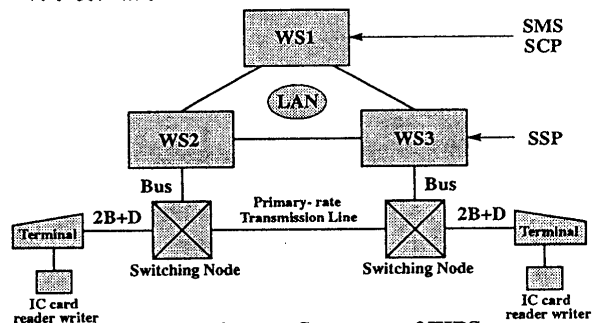


Figure 4 System Structure of TIPS

6. おわりに

提案概念は、各呼の制御を多重化しないため処理効率の低下は免れない。従って同時発呼数が増大した場合にどの程度のパフォーマンスが得られるかについて、今後測定評価を行う必要がある。

また、筆者らはサービスの増加に伴って生じるサービス干渉の問題についても検討を行っているが、その解決のための処理を加えた上での総合的なパフォーマンスの測定も今後の課題である。