

サービスベースシステムの概念と 構成法に関する一考察

4D-11

深沢 友雄 田中 英彦 元岡 達
(東京大学 工学部)

1. はじめに

将来の計算機の利用形態として、共同利用センタ(網)を、ユーザの手許にある知能端末(パーソナルコンピュータ)を通して使用するという形態が普及すると想定できる。その時に、ユーザから見える、システムが提供するサービスと、センタ側で提供するサービスの実現形態とが独立であり、端末-センタ間のインタラクションもサービスの定義とその要求/応答というインタフェースで統一したシステムを、「サービスベースシステム」と呼ぶ事にする。

本稿では、「サービスベースシステム」の概念を提案し、その実現の一形態について述べる。

2. サービスベースシステムの概念

2.1 サービスのモデル

計算機の提供するサービスの内容は、模式的に
サービス=データ+ロジック+コントロール
と、とらえる事ができる。このロジックとコントロールを「関数」として取り扱う。すなわち、
・入力データに対し、何らかの操作をほどこし、その結果を値として呼ばれたところに返す。
・ユーザからのサービス要求は、ユーザや、システムによって定義された関数を呼び出す事によって実行される。

・原則として、副作用がない。
等が、サービスの関数としての意味である。ユーザは、関数を組み合わせる事により、新たな関数を定義する。又、システムが最初から提供しているサービスを、「サービスプリミティブ」と呼ぶ事にする。

データの取り扱いに関しては、データベースの機能を有する事が望ましいが、現段階では通常のファイルシステムを想定している。

2.2 サービススキーマとサービス定義言語

ユーザからの関数呼び出しによってサービスを提供する為には、システムに、予めそのサービスの内容を定義しておく必要がある。すなわち、「サービススキーマ」を登録しておく必要がある。この記述言語が「サービス定義言語」である。記述内容は、

・サービス名
・仮引き数
・記述対象となっているサービスを構成する為に用いるサービスと、そのサービス間のコントロール
・記述対象となっているサービスの登録場所
(センタ側/端末側)
等である。

端末側、センタ側で、それぞれスキーマの登録簿を備え、それぞれの提供するサービスを管理する。(或るサービスを構成する為のサービスは、どちらにあってもよい。)

2.3 サービス要求言語

ユーザが実際にシステムにサービスを要求する時に使用する言語を「サービス要求言語」と呼ぶ。この言語内で指定された名前に対応するスキーマに従って、システムはサービスを提供する。複数のサービスの指定や、そのサービス間のコントロールの記述も可能である。又、端末側からセンタ側に処理を依頼する時も、この言語で統一する。

2.4 既存システムとの対応関係

本節では、サービスベースシステムと関連のある既存技術との関係について述べる。

<コンピュータ間のプロトコルとの関係>

サービスベースシステムにおける計算機間の通信は、全て、サービスの要求とその応答というレベルで統一されている。これは、既存のコンピュータ間の通信プロトコルの上に位置するものである。

<OSとの関係>

サービスの基本単位は、OSの扱う基本処理単位に相当する。機能的には、サービススキーマはカタログドプロシージャやコマンドプロシージャに、サービス定義言語はJCLやコマンド言語に、サービス要求言語はコマンドに対応する。但し、サービスベースシステムにおいては、端末に於ける処理も、センタにおける処理も、ユーザに対して「サービス」という共通のViewを持ち、同じ言語で記述できる。すなわち、現在の種々のOSのJCLやコマンド体系をハイレベル化し、統一したものがサービス定義/要求言語である。

<データベースとの関係>

DBMSは、計算機が提供する記憶機能の物理構造とは独立に、論理的なデータモデルを設定し、それに基づくデータの取り扱いを提供するものである。これに対し、サービスベースシステムは、データに対する種々の加工機能についても、その実現形態とは独立した論理的なモデルと、操作方法を提供するものである。概念的には、サービススキーマ/サービス定義/要求言語は、データスキーマ/DBDL/DMLと双対の関係にある。又、データベースの機能を仮定すると、サービスベースシステムは、データベースにおけるデータの取り扱いに、ロジックや、そのコントロールを付加したものと考える事もできる。

<プログラミング言語との関係>

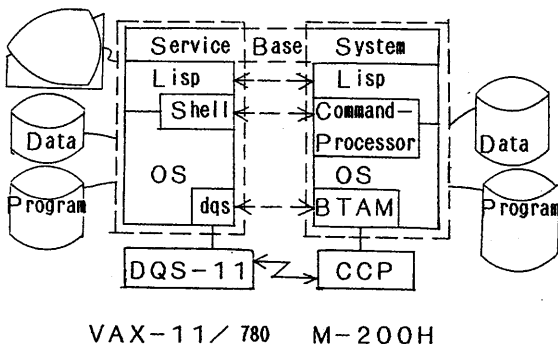
エディタやコンパイラ、ローダをサービスとして要求する事により、既存のプログラミング言語によるプログラムの作成、及びそのコンパイル、実行が可能である。エディタにおけるコマンドや入力テキストは、エディタという関数の実引き数であり、エディット終了後のソースプログラムが、関数の返す値である。作成したプログラムを新たにサービスプリミティブとして登録する事もできる。

3. Implementation

3.1 システム構成

東大大型計算機センタのM-200H、VAX-11/780を、それぞれセンタ、知能端末のプロトタイプとして使用している。これらは、M-200H側はCCPを通して、VAX側はCommunication-Controller (DQS-11)を通し、接続されている。(図1) 通信ソフトウェアは、M200-H側では、BTAMにおけるSYN同期手順、及びビデオデータシステムとのインタフェース用のソフトウェアを使用している。この仕様に合わせたソフトをVAX側で作成した。

図1 システムの構成



サービスプリミティブとしては、既存のOS (V OS3/M-200H, UNIX/VAX-11) が提供する種々のサービス機能、及びライブラリを利用している。

システムの記述には、Lisp (Franz Lisp [1]/VAX-11, UTILISP [2]/M-200H) を使用している。Lisp 内では、サービススキーマが、P-List の中の一つの属性の値として表現されている。

3.2 サービス定義/要求の方法

サービス定義/要求は、Lisp のTop-level からの関数呼び出しにより行なう。(サービス定義自体もサービス要求の一つである。)(図2)

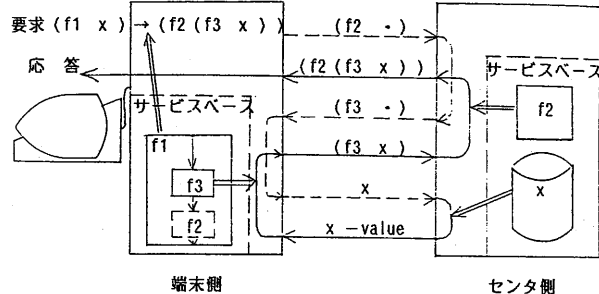
図2 サービスの定義/要求

<定義方法>

```
(def-service name arg-list body)
      body : lambda-expression
```

```
[例1] f1 := f2 * f3 の定義 (f2, f3: defined)
      (def-service f1 (x) (f2 (f3 x)))
```

[例2] サービス f1 の要求と処理の流れ



4. おわりに

本研究では、計算機の提供する機能を、全て「サービス」としてとらえ、端末-センタ間のハイレベルインタフェースと、User's View を、「関数」で統一した「サービスベースシステム」を提案し、その構成法について考察した。

今後の課題は、DB機能のサポート、種々のライブラリや応用プログラムの操作の記述、より高位の言語の開発、プログラム機能の形式的記述法の検討等である。

< 参考文献 >

[1]. J. K. Federaro, "Franz Lisp Manual".
 [2]. T. Chikayama, "CCUT Lisp Manual".