



DSSの動向とある試み

松田 寿子

動向についての考察

DSSの動向については拙著“特集に当って”の中でもその一端についてふれたのであるが、やはり T. H. Naylor の投げかけた波紋に対して各研究者の反応の仕方をとらえることにより、ひとつの尺度としての動向が把握できるのではないかと考える。

最も積極的に反応している例として R. W. Blanning は“State of the Art”としてのDSSの将来を論じ H. J. Watson, M. M. Hill 等はDSSの独自性と価値を主張している。

Blanning の論じる“DSSの今後の研究分野”は次の5つの柱から成っている。

1. 知識ベースにもとづく対話型システムの構築
2. DBMS (Data Base Management System) と同様に、モデルに対してもモデル・マネジメント・システムの枠組みを開発する
3. データ・マネジメントとモデル・マネジメントの統合化による情報管理
4. 現存するDSSの評価基準を与えるために、情報経済学の中での概念を拡張すること
5. DSSの設計及び実施に関連した行動研究の追求

1.の知識ベースDSSはいうまでもなく組織の中の因果関係(たとえば生産・在庫・発注・出荷といった部門間の関連等)を知識として獲得した後に、推論機構によって結論を導くもので、論理形式で表現された関係をAI (Artificial Intelligence) の技法で処理することに他ならない。

2.のモデル・マネジメント・システムはモデルをデータと同様に効果的に管理することに端を発し、2つの方向性を示している。ひとつは CODASYL 形式であり、

他のひとつは関係形式によるもの。CODASYL 形式では CODASYL DBMS がその記述形態で包含できるようなデンジョン・モデルを支援しており、関係形式ではモデルの実行時に仮想的な関係から導き出される様式になる。

3.の情報管理はデータの検索とその処理という従来からのやり方に、近年になって知識ベースのモデルを統合化するものが加わってきている。

4.のDSSのための情報経済学に対する関心は、J.C. Emery によるMISの費用対利益に関する考察で用いられたベイズの決定理論以来、主としてその適用上の普遍性を中心に議論が高まっている。

近年の情報サービスの発達によって意思決定過程もより複雑に拡がってきており、Peter Keen のいうようにDSSの動きもかつての供給主導型から需要促進型へと変化している状況にうまく合わせた新しい情報経済学の枠組が必要だとしている。

5.のDSSにおける行動研究については過去10~15年にわたってMISについてなされた研究の副産物として得られたもの。情報処理における認知形式からみた個人的な特性や、非構造的な不確実性下における意思決定のような環境特性、それにMISによる意思決定における特性として、情報の表現方法およびMIS開発過程でのユーザーの参加の度合等がMISにまつわる研究に関連して報告されている。このように大方のDSSの研究はMISについてなされた研究の伝統に立脚するものであるが、問題構造の把握やDSS設計の認識、非構造的な問題へのDSS設計、DSSスタッフの役割分担の混乱、学習理論を用いて分析と行動観察の統合を試みたものなど、豊富な参照例があがっている。

以上は Naylor の指摘の1.~3.までをカバーすると思われるのであるが、最後に4番目のオフィスとの関係についてDSSとOAについて論じているのでふれておこう。

Blanning は2つのことをあげているが、その1つは情報処理の観点からみた言語的に表現されたデータの扱いと、他の1つはワークステーションの統合化にともなうモデリングとその記述技法である。オフィスの将来は単なる神話ではなく、アプローチの対象が多く存在する一方で魔手もひそんでいる分野であるからDSSの方向性に大きなインパクトをもたらすものといえよう。

もちろん経営者はプログラマー等になる必要などなく、電話通信を主体としたオフィス機器の充実によって環境が整えば似たような条件は満たされてこよう。

Naylorのクレームに対する反論のもうひとつがWatsonとHillによるものである。論文の解説をするのが本論の主旨ではないので、詳述はさしひかえるが、結論としてはMISとDSSとの共存共栄を肯定し、DSSはタフな情報の要求に応え得るものだとしている。Blanningが主にコンピュータ技術の面から反応したのに対し、Watson & Hillは実際の利用面から反論しているのが特徴的である。そのうちの最たるものがNaylorの指摘の2番にある“DSSの枠組 (conceptual framework)”に関するもので、NaylorがSprague & Carlsonを評価したのに対し、普遍的に受け入れられるDSSの定義など存在しないという前提に立って、世に発表されているパッケージ類を例にあげ、個々のDSS (Sprague等のSpecific DSSに相当する)の枠組の豊富さを強調している。

技術的に表現するならばBlanningはSprague等のDSS Generatorに注目したのに対し、Watson & HillはそのGeneratorによって生成されるSpecific DSSに注目していることになる。この違いはDSSについて論じている時にしばしば起こりうる問題で、筆者もよくデモンストレーションのとき等の訴求対象の反応から伺い知るのであるが、物の見方が上述のような2通りに分れていて、“私、作る人。私、使う人”的な立場の違いからくる受け留め方の相違が見受けられる。

アプリケーションの生成過程に目をむける場合は生成器であるGeneratorに注目するが、そのアプリケーションで何ができるのかという結果に注目する場合は生成された結果のSpecific DSSに注目するのである。したがって前者はコンピュータ・サイエンスの観点からアプリケーション生成時の生産性 (ユーザー・インターフェースを含めて)に価値を求めるとし、後者はそのアプリケーションで処理される適用可能性 (現場で使いものになるかならないか)に価値を求めるとであろう。生成器が異なれば生成物も異なるのは道理であるが、コンピュータ・サイエンスの発達が生産器の普遍性を高めるにちがいないという予見を感じている1人として筆者もGenerator

の構造に興味をもっており、以上の論争をあえてフォローした次第である。この点についてはもう少し深く考察してみたいので、後述の項で再度ふれることにする。

他にも論争の材料は存在するのかもしれないがBlanningの“10年後にはすべてがDullとなる”という締めくくりはいかにも暗示的である。

Sprague & Carlsonのフレームワーク

SpragueとCarlsonによる業績はSpragueが以前から提唱してきた構想にCarlsonがシステム・アーキテクチャの面からその実現性を可能なものにより近づけたものとして一般に評価されたと考えてよい。H. A. Simonが30年前に今日の知識ベース時代の到来を予言した (ワシントンのIFORSでの当人の発言)ごとく、「構想」自体に、何らかの具体性・実証性をみるまでには時間を要する。

事実SpragueがDSS開発構想を出したのは1980年であるが、当時IBMにいたCarlsonとの共同作業によってその裏づけを得たことによって評価が得られたといえよう。Spragueのフレームワークは技術レベルをSpecific DSS (生成されたアプリケーション)、DSS Generators (アプリケーション生成器)、DSS Tools (生成時に必要とする道具類)の3種に分け、各技術レベルを担当する人たちの役割を次の5段階に分類している (図1)。

1. Specific DSSのレベルにおけるエンド・ユーザー (管理者も対象になる)は意思決定すべき問題に直面し、みずから生成されたアプリケーションを駆使して行動をとり、結果に責任をもつという役割を果たす。
2. Specific DSSとDSS Generatorの中間に位置する仲介者はエンド・ユーザーに対して希望のアプリケーションからうまく結果を抜き出すことができるように手助けをする役割を果たす。この場合、単に技術的な手助けばかりでなく、意思決定者の問題意識構造を的確に把握して助言をする等の行動も含まれよう。
3. DSS GeneratorからSpecific DSSを生成するDSS構築者はアプリケーションのタイプに応じて必要な道具類をとり出して生成器としての組み立てを行なう。したがってこの役割をはたす人はアプリケーションの問題領域に通じている他に、情報システム技術に必要な要因を技術的に把握できる能力を必要とする。
4. DSS ToolsとDSS Generatorの中間に位置する技術的援助者は生成器の能力の向上にたえず気を

配っており、現行のGeneratorで不足する機能を（Specific DSSから）要求された場合には、これを補ってゆく役割を有す。新しいデータ・ベースやモデル、ディスプレイ・フォーマット等技術水準に注意を払う必要がある。

5. DSS Tools のレベルにおける役割は常に新しい技術を開発し、新しい言語、新しいハードウェアおよびソフトウェアによってサブシステムの統合をはかる際の効率の向上を旨とする。

上記のフレーム・ワークを現実の問題に当てはめるとき各ケースによって（特にエンドユーザーのレベルや目的によって）その妥当性が云々されるとは思うが、少なくともひとつの企業体にアプローチする場合の基本的な考え方のモデルとして示唆に富んだ構想と考えるのである。これまでの情報システムの発展過程からみて1.の役割を中心とした人たちと5.の役割を中心とした人たちがそれぞれ固有の使命にもとづいてシステム作りにならずさわって、前者はアプリケーションの開発、後者はシステム技術の開発を中心としており、両者の仲介をミッションとする位置づけがあいまいであったことがいえるのではあるまいか。もちろん3.の役割を果たすにはそれなりの能力を養うまでに時間を要するが、このフレーム・ワークはその役割を中核にもってきている所が特徴的である。

技術的なシステム・アーキテクチャの観点からこのフレーム・ワークの具体化にはデータベース・マネジメント、モデルベース・マネジメント、それに対話をつかさどるダイアログ・マネジメントの機能が少なくとも必要で、世に発表されているDSSパッケージ群は必ず具備している機能であるが、汎用性がどこまで広がるかを検証してみた時に先のフレームワークの3.の役割を意識したかしないかの差が出るように思われる。1.の役割の観点からだけシステム作りを行なっている場合は“汎用的なものなどできっこない”ということになるが、5.の役割の観点から仕事をしている人からみれば“インフラストラクチャだけ用意してやれば十分。お好きなようにどうぞ”と超汎用的な見解も存在するわけで、DSS研究者のあいだでも見解の統一をみるのはだいぶ先のこと

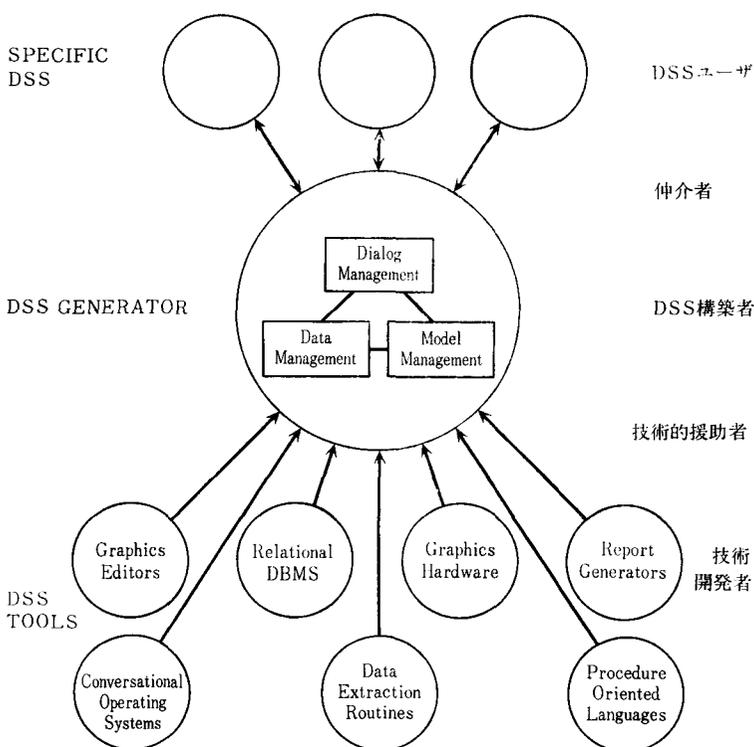


図1 DSS開発のフレームワーク ([3] p.304より)

であろう。

ある試み

“Specific DSS としてのモデル開発と関連するデータ抽出支援の例”

DSS Generatorの姿を模索する手段として、まずはじめに Specific DSS を作り上げ、それらを発生させる要件を探求するやり方を試みた。対象とするエンド・ユーザーは地方自治体のプランナーで、モデルは都市環境保全の計画支援を目的とする。Specific DSS のカラーはモデルの性質によって左右されるので従来からある統計的パッケージや数学的モデル、フィナンシャル・アナリシス、予算モデルといった個々のパッケージを即、DSS パッケージとしてイメージすることはたやすい。

計画支援モデルといった場合にも、たとえば図2のように土地利用計画や商圈拡張計画、施設計画、下水道普及計画といったように実際に意思決定を要するいくつかのモデルをとりあげてモデル化をすることになるが、DSSの中でも特に戦略的なモデルの場合は同じモデルをたびたび使用することは少ないから、いちど使ったモデルの更新のしやすさとか追加、削除の容易さ、心ゆくまでシミュレーションができる環境等考えるとパーソナル・

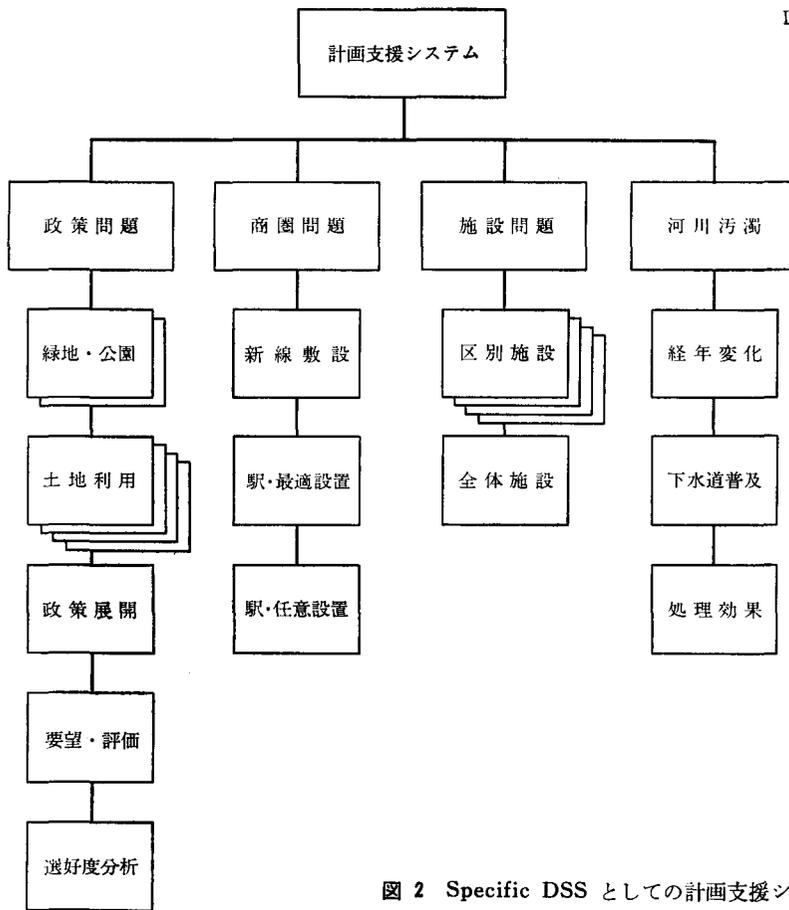


図 2 Specific DSS としての計画支援システム

コンピュータ上に簡易言語を使ってモデルを構築することが望ましい。このことはとりもなおさずエンドユーザー志向の分散処理形態をとることに通じるので、HOST・コンピュータへのアクセス、分散型データベースの構築へと発展する(図3)。問題解決を目的とする Specific DSS では自分で表現したいモデルが決定すると自在にデータを抽出して試行錯誤をくりかえすのが普通であるから、その際エンド・ユーザーの手元にデータが存在しているのが便利であるけれども、いつどのようなデータが必要になるかもわからない状況 (unstructured とか ill-defined とか いわれる問題構造) に対して十分なデータをパーソナル・コンピュータに具備するのは不可能であるから、中央のコンピュータからデータを抜き出してそれをパーソナル・コンピュータにダウンロードする機能が簡単に使えると便利である。Toolsmiths としては、たとえば IBM 社製品を例にとるならば中央のデータベース管理システムとして関係形式の SQL/DS* を使い、データの検索およびレポート作成を容易ならしめる QMF**, そしてダウンロードには 3270 PC*** といっ

DATA EXTRACTION/DOWN-LOAD

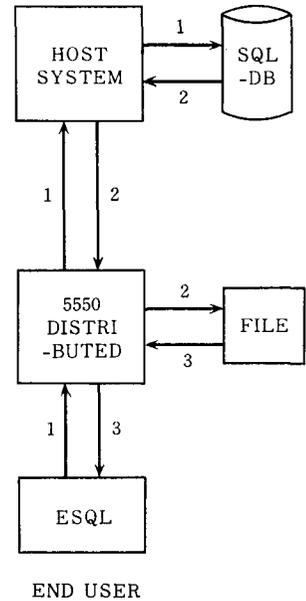


図 3 分散処理のためのデータ抽出

1. 検索事項のHOST側への転送
2. 検索結果のダウンロード
3. 分散データベース使用

た具合に一応揃っているかに思われるが、エンドユーザーがこれらの道具を使ってみずからのアプリケーションを動かすには、それなりの労力と時間を要する。

そこで生成機能のひとつとしてユーティリティを準備し、HOST側の演算処理を軽減して分散処理によってアプリケーションを実行することを試みた。

このアイデアは、本特集号で報告されているパネル討論会に出席したパネリストの1人である Dr. A. D. Narasimhalu 氏によるところが大きい。氏は当社サイエンス・インスティテュートの客員研究員として1984年4月から1年間日本に滞在し、筆者とDSSの研究にたずさわった。現在はシンガポールのISS (Institute of Systems Science) で研究をつづけておられる。アイデアの骨子は検索事項をフォーマット化し(図4)、HOST側に送ってHOST・コンピュータが検索を実行した後、結果のデータをパーソナル・コンピュータ側にダウンロードすることであるが、HOST側の処理において論理データの独立性に対する検索に工夫がなされているので、エンドユーザーがデータの所在を知らなくても検索がで

