A COMPARATIVE ANALYSIS OF FINANCIAL MODELING LANGUAGES

John M.Lewis

ABSTRACT

The increasing popularity of financial reporting and modeling systems, along with the myriad of opportunities in this field, has left many a prospective user asking the question: "Where do I begin?" The user needs guidelines for a selection process and a general comparison of alternatives available in the financial systems area. The steps of the selection process include: defining financial requirements and needs; establishing general and specific criteria; and, finally, comparing alternatives offered by vendors and their products.

Using a "new product" problem as a means of exploring the various systems' approaches, this paper presents examples of coding so that the user can evaluate the difference in working with several types of systems. Included are programming and APL based languages, FORTRAN based systems, numerically coded systems and interpretive English language systems. The actual cost comparisons of running the new product problem on twenty-one systems concludes the paper.

INTRODUCTION

Over the last few years, there has been a strong movement by users of Remote Computer Services towards the use of various applications packages. For example, Data Base Management Systems and Statistical packages have become increasingly popular. One of the most rapid growth areas in applications has been interactive financial planning systems. In fact, financial reporting and modeling systems have been proliferating so rapidly that a prospective user is confronted with a bewildering array of options in this field.

Faced by a confusing assortment of choices and inundated by sales brochures from multiple vendors, what should the user's first step be?

The number one priority of the prospective user should be to take a close look at the needs and requirements of his own business. It is absolutely crucial that the user make the establishment of his own criteria his very first step -- prior to becoming involved with selection between vendors and/or systems. By establishing his individual criteria before all else, the user lays the groundwork for screening and eliminating those financial systems which are obviously over- or under-qualified for his requirements. The immediate benefit which the user reaps is the reduction of the number of systems he must investigate further. The analysis and choice process becomes manageable and "do-able" for the user to undertake.

USER CRITERIA

To establish his individual criteria, the user must create a profile of his company by delineating his general and specific needs in several key areas.

High on the list of important considerations for the user are his company's personnel characteristics. The functional departments involved and their degree of participation should be specified: strategic planning, budget and controls, investment analysis, forecasting - all have different kinds of requirements from a system. The end users should be profiled, management levels should be considered along with classification by systems/programming, scientific/engineering, business/administration with specific types. The level of programming proficiency and experience with individual programming or planning languages should be investigated. Finally, the style of operation is most important - model building, level of sophistication, degree and frequency of change, and ad hoc reporting needs.

In-house computer facilities may be a significant factor in the user's decision process. Can the user access the facility directly, request runs and schedule changes, or are the in-house systems completely unavailable for modeling and planning? If the latter is the situation and long term plans indicate little likelihood of change, the requirement that a package be for sale has no importance. Otherwise, the makes and models and the interactive/remote batch facilities and plans can be integrated in the overall decision process for smooth transition in the event that an ultimate in-house conversion is planned.

Data Management

For many users, data management is an area assuming more and more importance. External data

needs should be assessed for items like economic models and time series, securities data bases, and any specialized requirements for market and geographic data. The amount of corporate data varies substantially - numbers of items and historical periods, volumes and significant digits can all be important. Once the data base is established, the frequency, source and means of the capture and update operations have to be identified. Finally, the characteristics of retrieval and reporting - frequency of access, degree of screening, purposes of data, types of users - must be considered.

Special and specific needs also deserve some thoughts. Network locations can be identified for interaction of remote batch terminals, sales and technical support coverage. Foreign activities for international network needs, consolidations, currency exchange rates, and common applications play a role. Are statistical requirements extensive - basic statistics, regressions, smoothing techniques, curve-fitting, progressions? Finally, current service contracts must be itemized and evaluated for terms and conditions, amount of usage, types of applications and degree of satisfaction. Prior and current vendor relationships can be an extremely valuable consideration in the decision process.

Although all the foregoing considerations are important in the user's decision, of primary importance is the user's determination of the exact type of reporting and modeling he will be doing. The extent of periodic needs for reports must be evaluated in terms of degree of repetition, formal procedures, multiple usages and volumes. The distinction between custom and standard display format can be significant - extent of spread and whether reports are for executive presentation or just informational. Do graphics requirements exist, and to what degree? And, what type, frequency, and priority exists for ad hoc reports?

Model Characteristics

Model characteristics can now be addressed. What size are the models, how many lines and periods, what level of detail and what comparisons are necessary? Again, the number of different models should be reviewed - corporate planning for strategies, markets, facilities, and financing; divisional planning for revenue, profit, budget and controls; and unit planning for marketing, product, R&D, cash flows and investments. Computational complexity can be most significant - financial and statistical routines, ratios, risk analysis, comparisons, lead and lag periods, and the importance of features like automatic reordering of equations and solution of simultaneous equations. The current approach to these problems (all manual, batch processed on computers, fixed models, programming language oriented) should be reviewed.

Once these reporting and modeling requirements are defined, the user can investigate the available financial systems. These range from the more limited "fixed" packages to the more powerful and flexible "high-level" languages. The user should select a package that will meet his needs and fulfill his ease of use and cost requirements.

FINANCIAL SYSTEMS

Several fundamentally different approaches have been taken to utilize the computer for financial planning and modeling activities. Many models have been, and still are, written in standard programming languages like FORTRAN, BASIC, COBOL, and even Assembler. But these programming languages are very different from modeling languages. For example, even if financial functions (e.g., rate of return, amortization) were added to the easy BASIC language, it would still not be a modeling language. The central difference between the two is that financial modeling languages are on a much higher level. A modeling language arranges loops, dimensions, and branches automatically, and reduces the number of instructions necessary to order and solve the equations contained in the model.

In general, financial models written in a standard programming language require significant development time and effort for subsequent modification. With the exception of very unique requirements, this approach is not recommended for anything but occasional program optimization for heavily used routines.

Other financial modeling packages closely resemble programming languages. Their usage is much similar to that of FORTRAN, COBOL or BASIC. In this respect they truly are languages rather than products or systems. It takes longer time to become competent in using a language like this. There is a much higher quantity of coding and programs naturally appear somewhat more complicated to set up. In the hands of a competent user, though, a financial system which is close to a programming language can have a wide range of capabilities and rather few limitations. A high degree of flexibility is inherently maintained. A good example of this is Tymshare's BBL (Basic Business Language).

Perhaps the differences between programming and modeling languages can be better demonstrated by showing an actual example of how various financial modeling systems would handle a simple and common financial application: new product planning This new product "problem" is one of the seven common user applications benchmarked in a report by Real Decisions Corporation. (All of the coding and cost information on the following systems has been extracted from RDC's copyrighted report, "A COMPARATIVE ANALYSIS OF FINANCIAL REPORTING AND MODELING SYSTEMS.")

NEW PRODUCT PROBLEM

The problem concerns an evaluation of the profitability of a new product over the next four years. The new product planning problem is described as follows:

Market research is evaluating the profitability of a new product over the next four years.

Units sold are anticipated to be 50,000 increasing by 15% per year. The selling price of $8.50 increased $0.05 per year after the initial year.

Variable cost per unit included in the cost of goods sold are as follows:

1. Raw Material = $3.00
2. Direct Labor = $2.00
3. Packaging = $0.50
4. Distribution = $0.75

Inflation over the next three years is expected to be 7%, 8% and 6%. Fixed costs involved in releasing the product fall into two categories:

1. Factory = $25,000
2. Other = $15,000

Administration feels that these costs can be held constant over the next four years, even in the face of inflation.

The effective tax rate is 22%.

The output of such a model appears as follows:

NEW PRODUCT MODEL

RUN 1	YEAR 1	YEAR 2	YEAR 3	YEAR 4
UNITS SOLD	50000	57500	66125	76044
SELLING PRICE	8.50	9.00	9.50	10.00
REVENUE	425000	517500	628187	760436
RAW MATERIAL	150000	184575	229242	279445
DIRECT LABOR	100000	123050	152828	186297
PACKAGING	25000	30762	38207	46574
DISTRIBUTION	37500	46144	57310	69861
GROSS PROFITS	112500	132969	150600	178259
FIXED COSTS	40000	40000	40000	40000
NET BEFORE TAXES	72500	92969	110600	138259
TAXES PAYABLE	15950	20453	24332	30417
NET INCOME	56550	72516	86268	107842

This problem will be used as an example to demonstrate the different approaches to financial modeling. By viewing the actual coding used by various systems, the prospective user can gain a feel for the general style of the financial planning system. For instance, a comparison of the programming necessary to solve this simple and straight forward modeling problem in BBL (Tymshare's programming oriented language) with that needed by an interpretive English modeling language will definitely underscore the differences between the two approaches.

Tymshare's BBL

The following BBL program solves the new product planning problem:

```
0100 REM*****************************************************************
0200 REM
0300 REM          VARIABLE TABLE
0400 REM
0500 REM     R$  RUN NAME                        G1  UNIT % GROWTH
0600 REM     R1  BASE QUANTITY UNITS             G2  PRICE GROWTH (IN $$)
0700 REM     R2  BASE PRICE                      I1(2-4) INFLATION RATES
0800 REM     C1(1)  RAW MATERIAL COST            F1  FACTORY COSTS
0900 REM     C2(1)  DIRECT LABOR                 F2  OTHER FIXED COSTS
1000 REM     C3(1)  PACKAGING                    T1  TAX RATE
1100 REM     C4(1)  DISTRIBUTION
1200 REM
1300 REM
1400 REM     THIS MODEL DOES BOTH UNIT AND PRICE SENSITIVITY
1500 REM
1600 REM     Q1  STARTING QUANTITY % DIFF        P1  STARTING PRICE % DIFF
1700 REM     Q2  ENDING % QUANTITY               P2  ENDING PRICE %
1800 REM     Q3  QUANTITY INCREMENT              P3  PRICE INCREMENT
```

```
2200 Y(*)=1,2,3,4
2300 FIX COL=7 CHA=9
2400 DEFINE REP1(12,4),REP2(12,4)
2500 SHOW//><"NEW PRODUCT MODEL"
2600 IF Q1<>Q2 SHOW ))-(1-K1).XX%;"  UNIT SENSITIVITY"
2700 IF P1<>P2 SHOW ))-(1-K2).XX%;"  PRICE SENSITIVITY"
2800 SHOW /R$,>)"YEAR 1",>>"YEAR 2",))"YEAR 3",>>"YEAR 4"
2900 SHOW 4:>>"------"
3000 SHOW
3100 SHOW "UNITS SOLD",ROW1.
3200 SHOW "SELLING PRICE",ROW2.XX
3300 SHOW 4:>)"-----"
3400 SHOW
3500 SHOW "REVENUE",ROW3.
3600 SHOW / "RAW MATERIAL",ROW4.
3700 SHOW "DIRECT LABOR",ROW5.
3800 SHOW "PACKAGING",ROW6.
3900 SHOW "DISTRIBUTION",ROW7.
4000 SHOW 4:>)"-----"
4100 SHOW
4200 SHOW "GROSS PROFITS",ROW8.
4300 SHOW "FIXED COSTS",ROW9.
4400 SHOW 4:>)"------"
4500 SHOW
4600 SHOW "NET BEFORE TAXES",ROW10.
4700 SHOW "TAXES PAYABLE",ROW11.
4800 SHOW /<<<"-------------------------------------------------"
4900 SHOW
5000 SHOW "NET INCOME",ROW12.
5100 SHOW <<<"-------------------------------------------------"
5200 SHOW
5300 REPEND
5400 PRINT "INPUT DATA FILE NAME (OR DONE)
5500 FIX LINEREAD ON
5600 INPUT A1$,A4$
5700 IF LFT$(A1$,2)="DO" THEN 14500
5800 IF LFT$(A4$,1)="Y" THEN 6300
5900 IF LFT$(A4$,1)="N" THEN 6300
6000 PRINT / " CONSOLIDATE    ";
6100 INPUT A4$
6200 GOTO5800
6300 FIX LINEREAD OFF
6400 Q1=1
6500 Q2=1
6600 Q3=1
6700 P1=1
6800 P2=1
6900 P3=1
7000 OPEN A1$ ON 20
7100 READ (20) R$
7200 READ (20) R1,G1,R2,G2
7300 READ (20) C1(1),C2(1),C3(1),C4(1),I1(2;4),F1,F2,T1
7400 PRINT " INPUT  WHAT-IF FILE   ";
7500 INPUT A2$
7600 IF A2$=" " THEN 8200
7700 IF LFT$(A2$,2)="NO" THEN 8200
7800 OPEN A2$ ON 21
7900 READ(21) R$
8000 READ(21) Q1,Q2,Q3
8100 READ(21) P1,P2,P3
8200 I1(1)=1
8300 I1(2;4)=I1(2;4)+1
8400 C1(2;4)=C1(1;3)*I1(2;4)
8500 C2(2;4)=C2(1;3)*I1(2;4)
8600 C3(2;4)=C3(1;3)*I1(2;4)
8700 C4(2;4)=C4(1;3)*I1(2;4)
8800 FOR K1=Q1 TO Q2 STEP Q3
8900 FOR K2=P1 TO P2 STEP P3
9000 ROW1(1)=R1
9100 ROW2(1)=R2
9200 ROW1(2;4)=ROW1(1;3)*(1+G1)
9300 ROW2(2;4)=ROW2(1;3)+G2
9400 ROW1=ROW1*K1
9500 ROW2=ROW2*K2
9600 ROW3=ROW1*ROW2
9700 ROW4=ROW1*C1(*)
9800 ROW5=ROW1*C2(*)
9900 ROW6=ROW1*C3(*)
10000 ROW7=ROW1*C4(*)
10100 ROW8=ROW3-ROW4-ROW5-ROW6-ROW7
10200 ROW9(*)=F1+F2
10300 ROW10=ROW8-ROW9
10400 ROW11=ROW10*T1
10500 ROW12=ROW10-ROW11
10600 IF K1>Q1 THEN 11000
10700 IF K2>P1 THEN 11000
10800 PRINT " WHICH REPORT (SHORT,LONG,GRAPH,TARGET) ";
10900 INPUT A1$
11000 IF LFT$(A1$,1)="L" THEN 11600
11100 IF LFT$(A1$,1)="S" THEN 11900
11200 IF LFT$(A1$,1)="G" THEN 12200
11300 IF LFT$(A1$,1)="T" THEN 13300
11400 PRINT " INVALID RESPONSE  "
11500 GOTO10800
11600 REM*********************************
11700 DISPLAY REP1
11800 GOTO12800
11900 DISPLAY REP1 USING 2600-3000,3100,3200,3500,4200,4600,5000
12000 PRINT ////
12100 GOTO12800
12200 REM GRAPHICS
12300 PRINT /////
12400 DISPLAY REP1 USING 2500,2600,2700,5200
12500 PRINT //
12600 GRAPH ROW3;ROW10 VS Y(*) STEP=1
12700 PRINT /////
12800 IF LFT$(A4$,1)<>"Y" THEN 14300
12900 REP2=REP2+REP1
13000 REP2(2,*)=REP1(2,*)
13100 K7=K7+1
13200 GOTO143000
13300 PRINT //" WHAT YEAR,TARGET INCOME  ";
13400 INPUT Y1,T9
13500 U1=(T9/.78+ROW9(Y1))/(ROW2(Y1)-C1(Y1)-C2(Y1)-C3(Y1)-C4(Y1))
13600 U2=INT(U1+.999)
13700 N6=.78*(U2*ROW2(Y1))-ROW9(Y1)-(U2*(C1(Y1)+C2(Y1)+C3(Y1)+C4(Y1))
13800 PRINT ///>> "UNITS SOLD",>>"PRICE", >> "NET INCOME"
13900 PRINT / >>  " ",U1.XX,ROW2(Y1),$T9.XX
14000 PRINT / >>  " ",U2.XX,ROW2(Y1),$N6.XX
14100 PRINT //
14200 GOTO14300
14300 NEXT K2,K1
14400 GOTO5400
14500 PRINT
14600 IF K7=0 THEN 15000
14700 R$="CONSOLIDATED"
14800 PRINT //
14900 DISPLAY REP2
15000 END
```

```
TYPE PROD.DAT
BASE RUN
50000,.15
8.5 ,.5
3,2,.5,75
7%,8%,6%
25000
15000
22%
```

The APL language is not included in the grouping of programming languages since it contains many characteristics which provide efficiencies in vector and matrix oriented computations. While APL is a precise mathematical language, it still requires a level of programming proficiency far beyond the higher-level modeling systems, and a degree of expertise is necessary to "read" a program. APL is also used as the base language for several planning packages. An example of this is FPS, an APL-based financial language on STSC. Here is how FPS handles the new product problem:

STSC's FPS

MODEL - BASE CASE

```
    ∇ NEWPROD
[1]   →SEGMENT 1
[2]   UNITS←YDS 1
[3]   INFLATION←CUMPROD YDS 3 ⍝ INFLATION FACTOR
[4]   1 EQ UNITS
[5]   2 EQ YDS 2 ⍝ PRICES
[6]   3 EQ UNITS×L 2 ⍝ REVENUE
[7]   4 5 6 7 MEQ(P 3 4 5 6) MPROD UNITS×INFLATION
[8]   8 EQ(L 3)-SUM 4 5 6 7
[9]   9 EQ YDS 4
[10]  10 EQ(L 8)-L 9 ⍝ NET INCOME BEFORE TAXES
[11]  11 EQ 0.22×L 10 ⍝ TAX AMOUNT
[12]  12 EQ(L 10)-L 11 ⍝ NET INCOME AFTER TAXES
[13]  →END
    ∇
```

DATA INPUT

```
    ∇ BASECASE
[1]   P 1 STORE 1.15 ⍝ ANNUAL PCT SALES INCREASE
[2]   YDS 1 STORE⌈50000×(P 1)* 0 1 2 3 ⍝ SALES VOLUME
[3]   YDS 2 STORE 8.5 9 9.5 10 ⍝ SELLING PRICE
[4]   YDS 3 STORE 1 1.07 1.08 1.06 ⍝ INFLATION RATES
[5]   YDS 4 STORE 4⍴40000 ⍝ FIXED COSTS
[6]  ⍝ RAW MATL, DIR LABOR, PACKAGING, DISTRIB, TAX RATES
[7]   P 3 4 5 6 7 STORE 3 2 0.5 0.75 0.22
[8]   P 8 9 10 11 STORE YDS 1 ⍝ SALES VOLUMES
    ∇
```

```
DATABASE   4213773 NUPROD


      1  UNITS SOLD
      2  SELLING PRICE
      3  REVENUE
      4  RAW MATERIAL
      5  DIRECT LABOR
      6  PACKAGING
```

```
 7  DISTRIBUTION
 8  GROSS PROFITS
 9  FIXED COSTS
10  NET BEFORE TAXES
11  TAXES PAYABLE
12  NET INCOME

HEADING1: OUTPUT FROM MODEL
HEADING2: NEW PRODUCT
HEADING3:
HEADING4:
HEADING5:

COLUMN HEADINGS:

PAGE  1 ROW  1:     YEAR 1/   YEAR 2/   YEAR 3/   YEAR 4/


COLUMN DEFINITIONS:

PAGE 1:  Y1/Y2/Y3/Y4/


LINE PRINTING SEQUENCE:

ROW 1:  H1    H2     S     S   CH     S     1     2    CU
ROW 2:   3     S     4     5    6     7    CU     S     8
ROW 3:  CU     S    10    11   CU     S    12  DBCU

TIME SLICE CODES:

  EVERYTHING NORMAL (N)

FORMAT CODES:

FORMAT F0 :     1     3     4     5     6     7     8
    10    11    12
FORMAT F2 :     2

  EVERYTHING ELSE NORMAL (F1)
```

FPS-type packages do not require APL expertise by the end user. Although the implementation simplifies the use of APL for the laymen, extended use of these APL based financial systems necessitates more and more involvement with the underlying language.

Some financial packages are FORTRAN based systems. Early FORTRAN based packages were designed for heavy usage. The logic sections of these systems are modified versions of FORTRAN, thus enabling a user with that background to easily work with them. A classic example of this type of system is Rapidata's FISCAL package:

Rapidata's FISCAL

```
00010 :DATA
00020 1010 50000
00030 1020 8.5 9.0 9.5 10.0
00040 1035 1 1.07 1.08 1.06
00050 1040 3
00060 1060 2
00070 1080 0.5
00080 2000 0.75
00090 2030 40000 40000 40000 40000
00100 :REPORT
00110 "NEW PRODUCT MODEL"
00115 "-----------------"
00130 :COLUMNS
00140 "YEAR 1"
00150 "YEAR 2"
00160 "YEAR 3"
00170 "YEAR 4"
00180 :ACCOUNTS
01010 "UNITS SOLD"
01020 "SELLING PRICE"
01030 "REVENUE"
01035 S"INFLATE"
01040 S"RMATF
01050 "RAW MATERIALS"
01060 S"DLF"
01070 "DIRECT LABOR"
01080 S"PACKF"
01090 "PACKAGING"
02000 S"DISTF"
02010  "DISTRIBUTION"
02020 "GROSS PROFITS"
02030 "FIXED COSTS"
02040 "NET BEFORE TAXES"
02050 "TAXES PAYABLE"
02060 "NET INCOME"
04000 :EDITING
04005 SO 1010
04010 SU 1020 2010 2030 2050 2060
04020 SP 1030 1050 2020 2040 2060
04040 .2 1020
04050 PAGE 35
05000 :LOGIC DEMOIL
05010 DO 5200 COL=1,4
05020 IF (COL.EQ. 1) GOTO 5090
05030 A(1010)=A(1010,COL-1)*1.15
05050 A(1040)=A(1040,COL-1)*A(1035)
05060 A(1060)=A(1060,COL-1)*A(1035)
05070 A(1080)=A(1080,COL-1)*A(1035)
05080 A(2000)=A(2000,Col-1)*A(1035)
05090 A(1030)=A(1010)*A(1020)
05100 A(1050)=A(1040)*A(1010)
05110 A(1070)=A(1060)*A(1010)
05120 A(1090)=A(1080)*A(1010)
05130 A(2010)=A(2000)*A(1010)
05140 A(2020)=A(1030)-A(1050)-A(1070) &
05145       -A(1090)-A(2010)
05150 A(2040)=A(2020)-A(2030)
05160 IF(A(2040).LT.0) GOTO 5190
05170 A(2050)=A(2040)*.22
05180 GOTO 5200
05190 A(2050)=0
05200 A(2060)=A(2040)-A(2050)
```

More recently developed systems are compiled FORTRAN systems. Notice in the next example of ADP's FML package how it much more closely resembles a pure modeling system than FISCAL:

ADP's FML

```
00100  *COLUMNS
00110  1,"YEAR 1"
00120  2,"YEAR 2"
00130  3 "YEAR 3"
00140  4 "YEAR 4"
00150  *ROWS 20
00159  BASEUNITS
00160  UNITS,"UNITS SOLD"
00161  UNITFACT
00162  PRICEFACT
00170  PRICE,"SELLING PRICE"
00180  REV,"REVENUE"
00190  RAW,"RAW MATERIAL"
00200  DIR,"DIRECT LABOR"
00210  PACK,"PACKAGING"
00220  DIST,"DISTRIBUTION"
00230  GROSS,"GROSS PROFIT"
00240  FACTORY,"FACTORY FIXED"
00250  OTHER,"OTHER FIXED COSTS"
00260  FIXED,"FIXED COSTS"
00270  NET,"NET BEFORE TAXES"
00280  TAX,"TAXES PAYABLE"
```

```
00290  NETINC:"NET INCOME"
00300  INFLAT,"INFLATION"
00310 *LOGIC
00320  COLUMN 2 TO 4
00330  BASEUNITS=@GROW(BASEUNITS,.15)
00340  PRICE=PRICE(PRIOR)+.5
00350  DO FOR ROW=RAWTO DIST
00360  A(T,ROW)=A(T-1,ROW)*(1+INFLAT)
00370  END
00371  COLUMN 1 TO 4
00380  UNITS=BASEUNITS
00381  UNITS=UNITS*(1+UNITFACT)
00383  PRICE=PRICE*(1+PRICEFACT)
00390  DO FOR ROW=RAW TO DIST
00400  A(T,ROW)=A(T,ROW)*UNITS
00410  END
00420  REV=PRICE*UNITS
00430  GROSS=REV-@CSUM(RAW,DIST)
00440  FIXED=FACTORY+OTHER
00450  NET+GROSS-FIXED
00460  TAX=.22*NET
00470  NETINC=NET-TAX
00480 *END
       REPORT
00090  WIDTH 132
00091  CENTER
00100  TITLE "SAMPLE OUTPUT"
00110  TITLE "-------------"
00120  TITLE "NEW PRODUCT MODEL"
00121  TITLE "-----------------"
00130  LEFT
00140  SUBTITLE "BASE RUN"
00150  SUBTITLE "---------"
00160  PRINT UNITS,PRICE(2),=REV,#RAW,&
00162     DIR,PACK,DIST
00165  PRINT=,GROSS,FIXED,=,NET,TAX
00170  LINE
00180  PRINT NETINC
00181  LINE
       DATA
00010  BASEUNITS,50000
00020  PRICE,8.5
00030  RAW,3
00040  DIR,2
00050  PACK,.5
00060  DIST,.75
00070  INFLAT(2),.07,.08,.06
00080  FACTORY,25000*
00090  OTHER,15000*
```

Fixed-format programs provide pre-defined reports from pre-specified input. Many of these programs have a limited set of alternative approaches for the user, and the standardization can minimize his cost and effort. While a number of these fixed systems currently enjoy a level of popularity, the increasing awareness of the power and flexibility of more advanced systems - and the programmable hand calculator - will gradually displace their usage.

Yet another type of financial modeling package is a numerically coded system. For example, one numerical system (SBC's PROPHIT II) includes on each line: the line number; a line title for report use or identification; a numerical operation code; a print index for decimals, scaling, or suppression; a line index to identify procedures for totals; factors and line references specifying constants and prior line calculations. Here is how PROPHIT II handled the new product problem:

SBC's PROPHIT II

```
1000 UNITS SOLD, 40,1,0
1010 SELLING PRICE,40,1,0
1020 "      ",27,1
1030 "      ",25,1
1100 REVENUE,34,1,1,1000,1010,0
1100 "      ",25,1
1200 INFLATION,40,-0,0
1210 RAW MATERIAL,19,1,0,3,1000,1200
1220 DIRECT LABOR,19,1,0,2,1000,1200
1230 PACKAGING,19,1,0,.5,1000,1200
1240 DISTRIBUTION,19,1,0,.75,1000,1200
1250 "      ",27,1
1300 GROSS PROFITS,33,1,0,1100,-1210,&
1305     -1220,-1230,-1240,0
1310 FIXED COSTS,40,1,0
1320 "     ",27,1
1330 "     ",25,1
1400 NET BEFORE TAXES,7,1,1,1300,1310
1410 TAXES PAYABLE,11,1,0,.22,1400
1420 "      ",27,1
1500 NET INCOME,7,1,0,1400,1410
1510 :      ",30,1
```

```
NPPPRO

0 PRO,1,4
1 3
2 1/1/77
11 NEW PRODUCT MODEL,;
12 "      ",;
1000 1.8,1,50000,15
1010 1,8.5,9.0,9.5,10.0
1200 1,1.00,1.07,1.1556,1.224936
1310 5,40000
```

Although initially easy to use in many cases, more complex operations in numeric code packages can become difficult or impractical due to the rigid structure of the system.

Finally, there are interpretive English language systems. Probably the most misleading claim made today by vendors of financial modeling systems concerns the "easy-to-use, English oriented systems." The user must be aware that almost every system today is advertised as being English language based, with power and flexibility for the businessman with no computer knowledge. The degree to which that statement applies varies considerably between systems. The higher-level English language financial systems are diverse in nature, easy to use, and more readily understandable to the laymen. They may contain extensive financial and statistical functions, thus making them very flexible and powerful while maintaining a simple English orientation. One of the best examples of this type of package is CDC's IFPS:

```
        MODEL     FACTORYA
 1 COLUMNS YEAR 1, YEAR 2, YEAR 3, YEAR 4
 2 UNITS SOLD = 50000, PREVIOUS UNITS SOLD * 1.15
 3 SELLING PRICE A = 8.5, PREVIOUS SELLING PRICE A + 0.5
 4 *
 5 REVENUE = UNITS SOLD * SELLING PRICE A
 6 *
 7 RAW MATERIAL = UNITS SOLD * RAW MAT COST
 8 DIRECT LABOR = LINE 2 * DIRECT LABOR COST
 9 PACKAGING = L2 * PACKAGING COST
10 DISTRIBUTION = UNITS SOLD * DISTRIBUTION COST
11 *
12 GROSS PROFITS = REVENUE - SUM(RAW MATERIAL THRU DISTRIBUTION)
13 FIXED COSTS = FACTORY + OTHER
14 *
15 NET BEFORE TAX = GROSS PROFITS - FIXED COSTS
16 TAXES PAYABLE = 0.22 * NET BEFORE TAX
17 *
18 * ------------------------------------------------
19 NET INCOME = NET BEFORE TAX - TAXES PAYABLE
20 * ------------------------------------------------
21 *   FACTORS USED IN CALCULATIONS
22 INFLATION RATE = 1.00, 1.07, 1.08, 1.06
23 RAW MAT COST = 3, PREVIOUS * INFLATION RATE
24 DIRECT LABOR COST = 2, PREVIOUS * INFLATION RATE
25 PACKAGING COST = 0.5, PREVIOUS PACKAGING COST * L12
26 DISTRIBUTION COST = 0.75, PREVIOUS DISTRIBUTION COST * L22
27 FACTORY = 25000
28 OTHER = 15000
```

```
        REPORT    NEWPRODUCT
 1 FORMAT..... 999,999  999,999  999,999  999,999
 2 CENTER SAMPLE OUTPUT
 3 UNDERLINE
 4 WENTER NEW PRODUCT MODEL
 5 UNDERLINE
 6 *PROMPT TYPE OF SOLUTION
 7 UNDERLINE
 8 COLUMN TITLES
 9 UNDERLINE
10 *
11 L2
12 FORMAT..... 9999.99  9999.99  9999.99  9999.99
13 L3
14 UNDERLINE
15 *
16 RL1
17 L5
18 *
19 RAW MATERIAL THRU DISTRIBUTION
20 UNDERLINE
21 *
22 GROSS PROFITS,FIXED COSTS
23 UNDERLINE
24 *
25 NET BEFORE TAX,TAXES PAYABLE
26 UNDERLINE
27 *
28 NET INCOME
29 UNDERLINE=
```

COMPARING ALTERNATIVES

Contrary to the opinion of some system developers and sales representatives, no system is the best for all financial planning requirements. Each system type mentioned here is considered superior to all others by some part of the user community. How, then, does a prospective user choose the interactive financial planning system that is best suited for him? The user must have a means of comparing the alternatives.

Real Decisions Corporation's report, "A Comparative Analysis of Financial Reporting and Modeling Systems," provides users with an understanding of the alternatives available in this area. It contains a Vendors and Product Information section in which users can get answers to specific questions on the overall capabilities of the companies and the specific products involved in the report. The Benchmark Problems and Analysis section of the report deals with common user applications. In addition to the New Product Planning problem, there are six other common applications benchmarked:

- o Consolidation of Financial Results
- o Cash Flow Projections
- o Project Financing
- o Long Range Plan
- o Sales Driven Model
- o Econometric Exercise

This section of the report not only deals with different application problems and their cost comparisons, but gives the user a flavor of how easy or difficult the system is to use and whether it conforms to his standards. The report also contains a commentary of each system with regard to:

- o Relative Costs
- o Ease of Use
- o Completeness
- o Documentation
- o Storage and Connect Charges
- o Special Analyses

The following pages describe a Real Decisions suggested procedure for prospective users of financial systems. It should be keptin mind that they are also users of our report!

Vendor Overview

After setting up his individual criteria, a user doing an RCS vendor overview can rather quickly and easily determine whether a vendor has the general characteristics a user wants or needs. A Vendor's history and outlook can provide a comfort factor for subsequent dealings, and the type of hardware and operating software may or may not be compatible with needs. Of more direct significance is the network relative to corporate locations - this applies to sales and technical support activities as well as local dial and/or speed terminal support.

Application availability is of prime importance. What major financial systems are available and how do they relate to one another? In addition, what about data management (or data base management) systems? Are the areas of importance able to be accommodated either with major systems or in companion products - such things as time series analysis, advanced statistics, graphics and risk analysis? The potential user should have a reasonable understanding of the overall vendor application capabilities and where practical, future plans as well.

Data Bases Available

External data bases are of supreme importance to some users and of absolutely no concern to others. A few vendors provide macro-economic models and advanced econometric services. Many others provide economic data bases, at least access to National Bureau of Economic Research (NBER) data. Securities data bases are available in a variety of forms by several vendors. In addition, special purpose data bases and a variety of access mechanisms are also available. It is worthwhile to consider this point for current and future needs, and to make it a part of the decision process.

RCS vendor characteristics can have a significant effect on long term relationships. Pricing structures vary and so do service options - both important and dependent upon specific usages. System reliability and response of both facility and personnel can vary dramatically, as can user demand for those characteristics. Finally, documentation and training are crucial elements to many end users, and they are handled in an assortment of ways by the vendors - sometimes with rather startling differences within a single vendor by product and geographic location.

Financial Systems Components

Once a user determines that a vendor meets his general criteria, he may begin investigating whether his specific criteria are met by the financial system itself.

Financial systems components are quite different according to individual planning systems, both in regard to specific integral components and in style of implementation. The language characteristics must be viewed rather than just described, since most claim to be "English language, requiring no programming experience." The range starts rather close to a true programming language and some are really not language at all. Special attention should be directed to report generation, to available functions and routines for financial and statistical operations, and to the ease of use of the documentation.

Interface versus components - these are the two approaches used to satisfy various system requirements. One philosophy says that everything needed by the user should be embedded in a single system, since once you are in it everything is readily available for use. Other systems adopt the philosophy of concentrating on specific activities; they provide interfaces with data base management systems, with graphics packages, with risk analysis, time series analysis, statistics and others. Again, user criteria determine what style is most appropriate to individual requirements.

Benchmarks

An in-depth benchmark problem review should be done in order to get a general feel for the style of the financial planning system. A great deal can be learned by examining how different systems accomplish identical tasks. This is obviously not a complete answer, since different approaches can be used within a single system and no problem can test all system attributes, but it can provide some solid comparisons. Commonly used activities can then be viewed. How does one ask "what-if" questions, perform sensitivity analysis or do backward iteration or goal seeking? Handling these questions subsequent to model building varies in time, effort and expense.

Selected benchmark usages should now be observed in light of the personalized criteria previously established. Items like consolidation capabilities, handling of complex computations, and the means for totaling and averaging in various combinations are handled in a surprising number of ways by different systems. If the user needs features such as automatic reordering of logic statements and the solution of simultaneous equations, the field of choice narrows dramatically.

Cost Factors

Finally, comparative cost factors need to be understood. The Real Decision's Financial Report can also greatly assist a prospective user in this area. As a demonstration portions of the previously referenced new product problem will be used. The vendors were asked to code and run all report problems on their systems in order to insure an accurate representation of them. In this example, vendors were required to show how the data and logic files are read in and displayed and how a formal report is printed on each system. Also requested were several sensitivity analyses to be done with complete and partial print outs. Individual CPU costs were taken for each "run" and exhibited in comparative charts.

This part of the paper is concerned with runs 1 and 4 of the new product problem. Run 1 involves coding and solving the complete problem with the entire model being printed out in a formal report. This is the same coding as was found in the earlier examples of this paper. Run 4 is a "what-if" type question which shows the effect of changing the number of units sold over a range from -10% to 5% (stepped sensitivity). Instead of displaying the entire model as in Run 1, though, Run 4 requires only selected rows to be printed out. The CPU cost for Runs 1 and 4 of the new product problem are displayed in the table on the following page.

Understanding comparative cost factors, though, is definitely not a simple matter of looking at the different costs of CPU time to run specific models and perform varying what-ifs and sensitivity analyses. Depending again on individual criteria, connect charges and storage costs can far outweigh the cost of computations. In other cases, the man time required to develop and modify the models turns out to be the overriding factor and costs of operation become rather insignificant. System costs can vary between building and operating the models. The ultimate usage of the system must be kept in mind.

In conclusion, it is essential that users of financial modeling and reporting systems evaluate and periodically re-evaluate their criteria and the ever changing means of satisfying those criteria. Once a system is carefully selected, the job is not complete. Rules of usage should be established. Improvements within the chosen system must be sought constantly. Periodic reviews must be undertaken for changes in criteria and available systems to satisfy them. In this dynamic segment of the computer industry, new solutions regularly appear - and with them comes the opportunity for improvement.

CPU COSTS FOR NEW PRODUCT

PROBLEM - RUNS 1 & 4

VENDOR	PACKAGE	CPU COSTS RUN 1	% OF HIGHEST	CPU COSTS RUN 4	% OF HIGHEST
ADP Network Services	FML	$ 6.96	68%	$ .92	5%
Boeing Computer Ser.	EIS	.54	5%	.99	6%
CallData Systems	CALLPLAN	1.42	14%	3.67	21%
CompuServe	CUFFS	7.02	68%	8.52	49%
Comshare	FCS	1.12	11%	.91	5%
Computer Sciences	FLARES	7.60	79%	5.53	31%
Control Data	IFPS	1.92	19%	1.63	9%
Data Resources	EPS	3.00	29%	3.45	20%
General Electric	FAL II	3.82	37%	7.36	42%
Informatics	SIMPLAN	2.05	20%	3.21	18%
Interactive Data	XSIM	4.06	39%	5.65	32%
On-Line Systems	PAMS	10.30	100%	6.50	37%
On-Line Systems	OLSFMS	2.65	26%	4.40	25%
Rapidata	FISCAL	4.68	45%	9.66	55%
Ross Systems	MAPS	1.04	10%	1.33	8%
Scientific TS	FPS	1.39	13%	3.09	18%
Service Bureau	PROPHIT II	1.08	10%	3.60	21%
Time Sharing Resources	INSIGHT	5.85	57%	17.44	100%
Tymshare	BBL	5.20	50%	1.70	10%
Tymshare	EXPRESS	1.82	18%	3.29	19%
United Computing	FORESIGHT	.53	5%	1.56	9%