SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS NONE
LE	3. DISTRIBUTION/AVAILABILITY OF REPORT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.
AD-A220 937 R(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) AFIT/CI/CIA- 90-012
6a. NAME OF PERFORMING ORGANIZATION AFIT STUDENT AT University of Arizona	6b. OFFICE SYMBOL (If applicable)
	7a. NAME OF MONITORING ORGANIZATION AFIT/CIA
6c. ADDRESS (City, State, and ZIP Code)	7b. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB OH 45433-6583
6a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (If applicable)
	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification) (UNCLASSIFIED)
Innovator A Financial Expert System

12. PERSONAL AUTHOR(S)
James V. Nutley

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
THESIS/[redacted]	FROM _____ TO _____	1990	114

16. SUPPLEMENTARY NOTATION
APPROVED FOR PUBLIC RELEASE IAW AFR 190-1
ERNEST A. HAYGOOD, 1st Lt, USAF
Executive Officer, Civilian Institution Programs

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
ERNEST A. HAYGOOD, 1st Lt, USAF	(513) 255-2259	AFIT/CI

INNOVATOR

A FINANCIAL EXPERT SYSTEM

by

James V. Nutley

A Report Submitted in Partial Fulfillment
of the Requirement for the Degree
of Masters of Science
(Management Information Systems)
in The University of Arizona

February 1990

Master committee:

Professor Sudha Ram, Chairman
Professor Doug Vogel

ABSTRACT

INNOVATOR

A FINANCIAL EXPERT SYSTEM

by

James V. Nutley

Chairman: Professor Sudha Ram

The INNOVATOR expert system was designed to assist in the evaluation of new Financial Service Product ideas. It is written in the Expert System Environment (ESE) on an IBM 4381. During the course of this project I rewrote INNOVATOR under the direction of Professor Sudha Ram. INNOVATOR was modified during this project to take advantage of a new interface to DB2 and lessons learned from the first implementation.

ACKNOWLEDGEMENTS

I would like to thank Professor Sudha Ram for her guidance and direction.

And my lovely wife, Dorothy for moral and physical support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INSTRUCTION

Overview of the area of Expert Systems

Artificial Intelligence (AI) is the discipline which tries to build machines that can be considered intelligent. Definitions of the word "intelligent" have proved slippery, still, some of the products of man's effort to duplicate himself in silicon have born useful fruit. One such product is the Expert System.

Expert Systems are a departure from the historical direction of AI. Previous systems have attempted to implement generalized problem solving programs on wide varieties of problems, expert Systems select a specific domain in which they will function, defined by the Knowledge Base of the system. Knowledge Bases are translations of domain specific knowledge which a human expert would apply to a problem in his/her field of expertise. While a procedural computer program will process the data given it and deliver some output state, an expert System searches for a solution to a problem. The goal of a successful expert system is not to replace human experts in their field, but rather to handle problems that are ill-structured, and combine quantitative and judgmental data, which would ordinarily require the domain specific expertise of a human being, but are well enough understood to translate to machine processable form.

Once translated, the Knowledge Base can be accessed to solve these problems while leaving the expert time to focus on less well understood problems, or it can assist in the training of new experts, or it can make the knowledge of a single expert available to a larger audience than a single human being might otherwise assist.

The screening of potential new financial service products is a problem domain which fits the criteria for using Expert System technology.

> "... the decision making process is not necessarily well structured, it uses variables that are both categorical and continuous, and relationships between the variables that are both numerical and logical."[1]

Their is also a clear need for this type of a system.

> "... the process of introducing new products is very unstructured and hence very risky. Failure rates of new products are substantial. Past figures on new product launches reveal that 33 to 60% of new products are commercially unsuccessful. This is a disturbingly high failure rate considering the amount of organizational resources, such as dollar investment and R & D talent,

---

[1] Sudha Ram and Sundaresan Ram, "INNOVATOR: AN EXPERT SYSTEM FOR NEW PRODUCT LAUNCH DECISIONS," Applied Artificial Intelligence, 2:129-148, 1988.

channeled into new product development. Some organizations consistently outperform others in the new product area, thanks to their group of new product experts. An expert system that could faithfully reproduce the judgement of these experts could save organizations tremendous amounts of resources and simultaneously improve the success rate of their innovations."[2]

It was with these factors in mind that Professors Sudha and Sundaresan Ram began the coding of the INNOVATOR Expert System. INNOVATOR's knowledge base was culled from five acknowledged experts in the field, and a wide ranging literature search. A version of the program was implemented by Mr. Jaafar Husain and a description of the system and its Knowledge Base has been published in the journal Applied Artificial Intelligence, #2 1988.

The INNOVATOR implementation was performed in the ESE (Expert Systems Environment) Shell, an IBM product. ESE is designed to run on IBM mainframes (a 4381 in this case). ESE allows developers to enter rules directly in an IF...Then format. It also allowed external routines to store and retrieve the Attribute states of the product being considered, although this was in a flat file format. The implementation worked, but the professors Ram believed that

---

[2] Ibid.

it could be improved. The following features were planned for a follow up version of INNOVATOR:

- connectivity to a true relational database for management of information
- ending reliance on routines which were constructed in third generation languages

These goals became achievable when IBM released an upgrade of ESE, including the Access command for FCB control text (see section 4.1.1) which allows direct access to SQL/DS and DB2.

The Purpose and Objective of this project was to build a new version of INNOVATOR, taking advantage of the direct database access I was expected to deliver a working, expandable ESE application which implemented the Knowledge Base of the INNOVATOR application without resort to external third generation language calls and with suitable modifications, supervised by Professor Sudha Ram, to take advantage of lessons learned in implementing the first INNOVATOR and implement good database design.

II. Previous Research

Knowledge elicited for INNOVATOR came from Financial Services Experts. Two of the experts chosen worked in national firms specializing in financial investments, one was from a small independent financial advisory firm, one worked in the investment division of a savings and loan association, and one worked for a commercial bank. The literature search

that followed covered Investment surveys, and financial magazines such as Money, Forbes, and the Wall Street Journal.

The Knowledge Base of Innovator can be represented for discussion as sets of text tables, which have a hierarchical relationship to each other. The root table is the list of product lines.

TABLE 1.0

Major Product Lines

- Annuities
- Bonds
- Insurance
- Mutual Funds
- Options
- Precious Metals
- Stocks
- Real Estate Partnerships

These represent the broad categories in to which INNOVATOR divides the domain of Financial Service Products. If we follow the Mutual Funds path, we come to the recognized types of Mutual Funds.

TABLE 1.1

Mutual Fund Product Types

- Balanced Funds
- Bond Funds
- Convertible Funds
- Growth Funds
- Growth and Income Funds
- Income Funds
- International Funds
- Money Market Funds
- Sector/Specialty Funds

Then established brand names of mutual fund products are identified.

TABLE 1.2

Mutual Fund Brand Names

- AARP
- American
- Bull-Bear
- Calvert
- Century
- Dean Witter
- Delaware
- Dreyfus
- Eaton Vance
- Fidelity
- Financial
- IDS
- Kemper
- MFS
- Merrill Lynch
- New England
- Oppenheimer
- Putnam
- Shearson
- Templeton
- USAA
- Value Line
- Vanguard

After the types of Financial Services had been identified, tables of key attributes were developed for evaluating the viability of Product Line, Product, or Brand.

TABLE 1.3

Product Line Evaluation Attributes

Attribute Classes				
Market Competition	Financial	Product	Corporate	
Estimated Market Size	Volume of Business	Distinctive-ness	Compatibility with existing products	Number of Competitors
Estimated Growth	Gross Margin	Consumer Appeal	Financial Outlay	Quality of Competitors Rate
Estimated Suscep-tibility	Return on Investment	Ease of Imitation	Organizational Structure	Vulner-ability of Competitors
Season-ality of demand	Payback Period	Contribution to Company Image	Personnel Skills in Marketing	
			Personnel Skills in Technical Product Development	

TABLE 1.4

Product Evaluation Attributes

Attribute Classes	
Performance	Rating
Past Year	Risk
Past 5 Years	
Past 10 Years	
Future (estimate)	

TABLE 1.5

Product Brand Evaluation Attributes

Attribute Classes				
Company Standing	Company Size	Earning Potential	Past Performance	Quality of Service
Age	Number of Funds Offered	Commission Rate	Growth Rate of Assets	Facility for Switching Between Funds
Reputation	Range of Funds Offered	Minimum Investment Amount	Average Return Compared to Industry Average	Service Representative's Availability
Management Turnover	Total Assets		Number of its Relationship Funds in the Industry	with
Brokers/Agents			Top 20	

With the Knowledge Acquisition complete, it only remains to translate this knowledge into ESE objects and debug. I had

this information and the previous ESE application, as well as guidance from Prof. Sudha Ram. I was ready to begin implementation.

III. User's Guide

My Goal was to implement the INNOVATOR ESE application, and to do so in such a way that it accessed SQL/DS. This chapter is designed to assist a user in using the application.

3.1 Names and Titles

The first thing the user will see is the welcome screen. After that, a user is asked questions. Just like a human expert, INNOVATOR must elicit data from the end-user in order to determine the degree to which the product being considered measures up. At first though, the user will have to answer questions about himself or herself. Data is gathered about the individual users primarily for product improvement at a later time, although usage monitoring could also be accomplished. An entry for each complete session is made, storing who used the system, what company they work for, when the consultation took place, and what the final evaluation is. The plan is for INNOVATOR's record of usage to guide future developers in evaluating who uses INNOVATOR, and in indicating who to elicit additional Knowledge or user interface improvement suggestions from. Security is not

supported by ESE, however the mainframe environment has ample password protections available.

It is important, in the current implementation, that great care be exercised while entering the company name, and that the name be entered in exactly the same manner for every session. INNOVATOR will search for existing "Profiles" using the company name, and a name which was spelled correctly but had capital letters the other lacked would be treated as an entirely different word.

FIGURE 1.0

```
 III  N   N  N   N  OOOO  V    V   AA   TTT  OOOO  RRRR
  I   NN  N  NN  N  O  O   V V    A  A   T   O  O  R  R
  I   N N N  N N N  O  O   V V    AAAA   T   O  O  RRR
 III  N   N  N   N  OOOO    V     A  A   T   OOOO  R  R

 ┌──────────────────────────────────────────────────────┐
 │ Please enter your company name (30 characters or less)│
 └──────────────────────────────────────────────────────┘

PF1 Help PF2 Review PF4 What  PF7 Up PF8 Down  PF10 How  PF11 Why
==>
```

Once the User has entered this information, He will be asked weather or not an existing profile should be used.

3.2 Profiles

INNOVATOR contains Knowledge about what attributes of a new product are important, it also knows that some things, such as Seasonality of Demand, are best when they are low, while others,

such as Product Distinctiveness, are best when they are high. However, INNOVATOR as an Expert System, has no way to tell what the financial status of the End-Users company is at the time of a consultation, and no means to determine the financial resources that a company might or might not be willing to devote to a new product, unless the end-user tells it. This is done by entering a "Profile".

A Profile is a collection of attributes with numeric values. These attributes represent the judgement of the end-user, or his company on what is acceptable for a new product for their specific company. Storing their own preferences in this fashion is a powerful feature, as it allows the company to establish their own levels at which they would commit funds to a new product. Multiple profiles can be stored for a single company to allow a variety of views to be applied to any product.

The user is asked if he/she would like to use an existing profile from his/her company, or if he/she would like to enter a new one. Depending on the type of product, INNOVATOR will prompt the user for a value for each necessary attribute. Profile attributes are usually expressed in millions of dollars, or as a percentage of some Financial concept, such as the possible market for the product.

Once the user has chosen or entered a profile, the product in question has a standard to be compared to and the consultation can begin.

3.3 Using the Program

FIGURE 1.1

```
       III  N  N  N  N  OOOO  V   V   AA   TTT  OOOO  RRRR
        I   NN N  NN N  O  O   V V   A  A   T   O  O  R  R
        I   N NN  N NN  O  O   V V   AAAA   T   O  O  RRR
       III  N  N  N  N  OOOO    V    A  A   T   OOOO  R  R
```

```
┌──────────────────────────────────────────────────────────────┐
│ What do you wish to evaluate?                                │
└──────────────────────────────────────────────────────────────┘
__ Product Line
__ Product
__ Brand

PF1 Help PF2 Review PF4 What  PF7 Up PF8 Down  PF10 How  PF11 Why
==>
```

From this point on, the user will be asked questions in a multiple choice fashion, until INNOVATOR is ready to deliver an evaluation. There are two distinct segments to this portion of the consultation.

First, INNOVATOR has the end-user establish a priority for each Attribute Class in the product he/she is considering. Although the user can specify that each attribute class is equally important, INNOVATOR arranges the rest of the consultation according to differences in the priorities, highest to lowest, and will simply choose a default order if the user considers all Attribute Classes of equal importance.

Next, the attributes of each class are considered in turn. The user must choose the appropriate value for each attribute according to the product being evaluated at the time. Once these

attributes are established, INNOVATOR will compare profile and product values, as well as evaluating the product against the established attributes in it's Knowledge Base, and deliver an evaluation to the screen (as well as the results log).

IV. Technical Manual

ESE is an established product from IBM. An ESE application is composed of five structures; Parameters, Rules, Groups, Screens, and Focus Control Blocks (FCBs). Although this chapter cannot be considered a substitute for the ESE Reference Manual, it contains descriptions of each of the ESE structures, and examples of how they were used in INNOVATOR.

4.1 FCB's

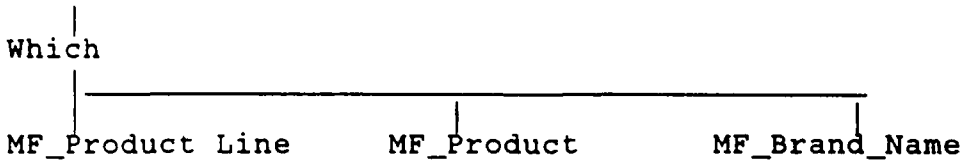
Expert systems use inference engines to perform their functions, but these engines do not process their rules sequentially. Non-sequential processing can have advantages for some problems, but not every problem can be solved without resort to some sequential processing. In order to address the possibility of dual approaches, ESE controls its applications through FCBs. An FCB can be thought of as a short script which the ESE application will perform. The script of what will be performed by an FDB is called it's Control Text. Control Texts

are written in Expert System Development Environment Control Language, which is composed of eight commands:

ACCESS	interface with SQL/DS and DB2
ACQUIRE	Obtain parameter values form an external data source
ASK	Ask End-User for data to obtain parameter values
DETERMINE	Invokes Backward Chaining Inference Engine to find a specific parameter value
DISCOVER	Invoke Forward Chaining Inference Engine to use a set of rules to find parameter values
DISPLAY	Displays parameters and their values to end users
ESTABLISH	Initiate processing of another FCB
PROCESS	Pass control to external data routines

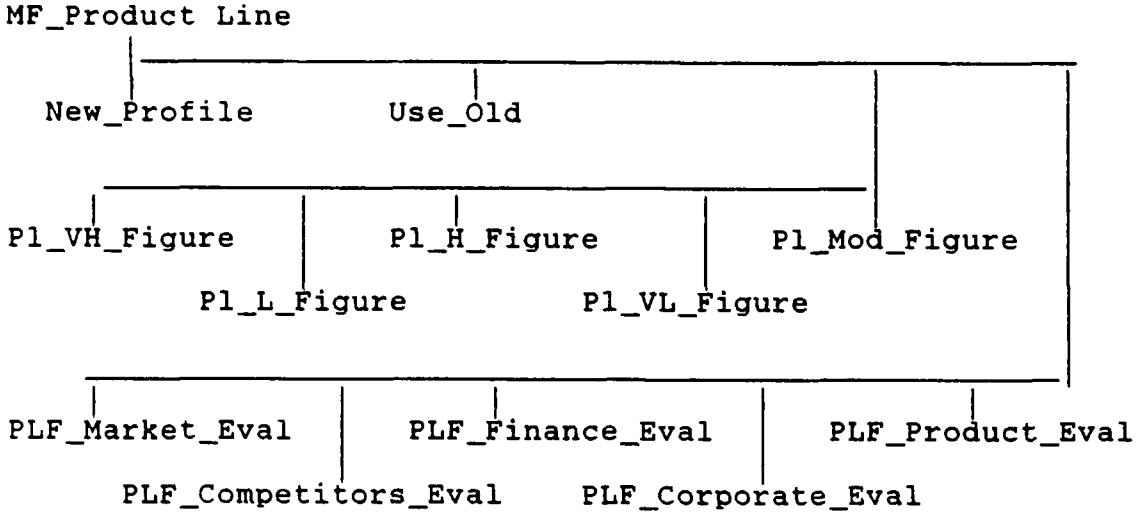
When an application contains more than one FCB, then the FCBs are arranged in a hierarchy beginning with a root FCB. In INNOVATOR, OVERMODULE is the root FCB for the application. It establishes WHICH in which the type of product is chosen. IF the product is a mutual fund, then either a Product, Product Line or Brand Name is chosen and pursued. The figure below shows the hierarchy to that point.

FIGURE 1.2



After the Type of Mutual Fund is established, INNOVATOR asks the user weather they want to use one of the existing profiles or create a new one. Separate FCBs, NEW_PROFILE and USE_OLD handle creation and retrieval of profiles. These FCBs contain the commands which read from and write to the database table PROFILE_MF_EVAL, where mutual fund profiles are kept. Once the profile to be used is loaded, five FCBs, named by format PL_<XX>_Figure, establish the users priority for the Attribute Classes to be used. Then the FCBs named by format PLF_<attribute>_EVAL control the evaluation, one attribute class at a time, of (in this case) the product line being considered. After they have finished control passes back up to WHICH, where the final results are displayed and saved in the Database Table RESULT_LOG.

FIGURE 1.3



This number of FCBs is larger than the minimum necessary to do this type of job, but I proceeded on the principle of structured programing that code should be modular in construction. The *_Figure modules are usable throughout the application with small modifications for priority setting, while the *_Eval modules

must be customized for each attribute class, but can be copied and modified from one branch to another.

FCBs have a number of properties that can be individually altered for specific effects. See Appendix A for a list of those properties.

4.2 Rules, and Parameters

The heart of a heuristic Knowledge Base are the facts (Parameters) that it consists of and the Rules that relate one fact to another. Like FCBs, both Parameters and Rules in ESE have properties that can be altered, see Appendix A.

Parameters are normally either character strings or numbers. Parameters may be assigned by the user during the session, or they may assume a value defaulted to them, or they may be assigned a value through the actions of an inference engine processing rules.

Rules are expressions of relationships between parameters. They are represented by statements which must follow the format below: (<> indicate that the contents are required and may be any string of the appropriate format, [] indicate that the string within is optional)

IF <premise_statement [and/or premise_statement[and/or ...]>

THEN <action_statement [and action_statement[and ...] >

Where the IF and THEN, one premise statement and one action statement are the minimum requirement for a rule. As many

additional Premise and Action statements can be assigned to the rule as one wishes, action statements will always be used, hence only AND may be used on them. Premise statements may be linked either with ANDs, or ORs. OR allows the rule to apply to different premises one at a time.

The premise statement itself is most often a numeric or logical relation. For instance, "<parameter> = <parameter>". Rules fire when premises are true, a false condition can be achieved by adding NOT to a premise as in "not <parameter> = <parameter>". Action statements can be of the same format as premise statements or they can direct a subset of the ESE command language, such as "Establish <FCB>".

There is an alternative form of rule, used when the premise statement of the rule must include statements of the relative levels of certainty in the parameters. This is a Fuzzy IF statement or FIF. An example of a FIF statement from INNOVATOR is:

```
fif  certainty of  (pl_finance_eval)    > .5 and
     certainty of  (pl_market_eval)     > .5 and
     certainty of (pl_corporate_eval)   > .5 and
     certainty of (pl_competitor_eval) > .5 and
     certainty of  (pl_product_eval)    > .5
then there is strong evidence that pl_final_eval is 'approve'
```

4.3 Naming Convention

As the number of ESE Objects increase, the difficulties of editing or modifying the Knowledge Base become more acute. There

is, however a simple method to maintain some control over increasing ESE Knowledge Bases, through the simple expedient of adopting a naming convention.

ESE's built in editor allows the convenience of wildcard searches, using the * character, so that a user may access any and more importantly, all parameters, or rules, or FCBs, groups and screens, whose names share identical characters.

I chose the following for my naming convention:

<Product Type>_<Attribute Name>_[Additional Id]_<Goal>

Where <Product Type> can be pl, which stands for product line, or P which stands for product, or b which stands for brand, <Attribute Name> stands for the attribute the object works with or assigns value to, [Additional Id] stands for whatever is necessary and appropriate if the object would otherwise have a duplicate of an existing name, and <Goal> refers to the goal of that part of the Knowledge Base which the object is associated with.

This convention was quite satisfactory for the development of the prototype of INNOVATOR, but as I began to expand it beyond mutual funds, I realized that it was not enough, an additional two character prefix is needed to identify differences in product catagories such as Mutual Funds vs. Real Estate Limited Partnerships vs. Stocks. The final form of the name of a Mutual Fund, product Line Market value evaluation rule would be:

MF_PL_Market_A_EVAL

Where "A" would indicate that the rule would test for conditions appropriate for the approval of the product line, while a "D" would indicate a test for conditions inappropriate for approval.

4.4 Database Properties

Two Database tables are used by the current form of INNOVATOR, although as more product types and catagories come into the data base, more will be needed. The first one stores profiles of the Mutual Fund Product Line, and should be of appropriate format for other types of profiles with only a name change. The advantage of this form of table, as pointed out by Professor Sudha Ram, is that name changes of attributes need not be propagated to the table columns.

```
CREATE TABLE PROFILE_PL_EVAL (COMPANY CHAR(30) NOT NULL,
COUSER CHAR(30), PROFILE_NAME CHAR(30), PARAMETER_NAME
CHAR(30), PARAMETER_VALUE INTEGER)
```

Where COMPANY is the column where the users company name is kept, COUSER is the column where the user's name is kept, PROFILE_NAME is the column where the name of the profile being stored is kept, PARAMETER_NAME is the column where the names of attributes are stored and PARAMETER_VALUE.

The second table is where records of the user session are kept. This is a relatively straightforward recording of the users name,

```
CREATE TABLE RESULT_LOG (COMPANY CHAR(30) NOT NULL, COUSER
CHAR(30), COTITLE CHAR(30), PROFILE_TYPE CHAR(12),
PROFILE_NAME CHAR(30), RESULT CHAR(40), STARTDATE CHAR(20),
FINISHDATE CHAR(20)).
```

Where COMPANY, COUSER, COTITLE, PROFILE_TYPE, PROFILE_NAME are the same as in the OTHER TABLE. RESULT stores the final results of the session. While STARTDATE and FINISHDATE store the beginning and ending times for the user consultation.

V. CONCLUSION

One would like to finish a software project believing that the best code possible at the time had been implemented. I regret that I cannot make that claim for my version of INNOVATOR. I embarked on this project with the tasking to implement access to DB2. This was accomplished. However, I am certain that improvements can be made in the performance and friendliness of the program. Clearly this version of INNOVATOR should not be the last one written.

The older version of INNOVATOR was a hybrid program, using procedural language routines to store and retrieve data where the present version directly accesses DB2. Perhaps this strategy should be reconsidered, not to abandon DB2, but to reincorporate the abilities of a procedural language.

INNOVATOR requires that a number of features relating to a single product be entered during each session. Rather than causing the expert system to query the user concerning each feature, a procedural program could assist the user in setting the values, and then pass the file to DB2. Or perhaps a form editor such as DBEDIT could be used. The ESE portion of the program could be summoned after the appropriate database tables had been updated or constructed, which would leave only the reasoning tasks to be done by ESE. This strategy might produce a faster application requiring fewer lines of code.

Another aspect of INNOVATOR which could be improved are the explanation facilities, the HOW and WHY commands. The chief difficulty here is that ESE does not support context sensitive explanation. HOW or WHY display the names of rules and parameters currently being operated on. It is more reasonable to believe that a user, activating the WHY facility, would want an explanation of what facts the program was seeking and how those facts would effect the heuristic being processed. Some expert system environments already support this type of facility, perhaps ESE will eventually be upgraded to support it also. While the programmer would obviously need to spend considerable extra time building effective explanation messages, the user would receive much better support.

"Nice to have" improvements would include a more modern user interface. For instance, when presented with a multiple choice list, the user might prefer highlighting the choice with a cursor

key and selecting it rather than entering an X or numeric character in the space in front of the selection.

A graphical output might be useful; perhaps with a bar graph representing how each category of attribute had exceeded or fallen short of the required value.

The section of my own code which least satisfies me are the rules for making the final determination on the product. I divided the possible results into a set of rangers and guessed which should be approved, marginally approved etc. Academic studies with guidelines that would indicate the cutoffs for each range would be better for this purpose. I neglected to look for this type of information and the accuracy of this version of INNOVATOR suffers for it.

I did learn a great deal about expert systems coding from this project. I hope to have another chance to apply it in the near future.

BIBLIOGRAPHY

Ram, Sudha and Sundaresan Ram, "INNOVATOR: AN EXPERT SYSTEM FOR NEW PRODUCT LAUNCH DECISIONS," Applied Artificial Intelligence 2:129-148, 1988.

Addis, T.R. Designing Knowledge-Based Systems. New Jersey: Prentice Hall, 1985.

Bowerman, Robert G., B.S. and Glover, David E., Ph.D. Putting Expert Systems Into Practice. New York: Van Nostrand Reinhold Co. Inc, 1988.

Walters, John and Nielson, Norman R. Crafting Knowledge Based Systems: Expert Systems Made Realistic. New York: John Wiley and Sons, Inc, 1988.

APPENDIX A

Rules and Parameters

FCB	Parameter	Rule
Additional Instance	Author	Author
Query	Comment	Comment
Announce	Constraint	Justification
Author	Default constraint	Name
Comment	Expect value	Owning FCBs
Control Text	Format Mask	Print Name
Display Screen	Name	Rule Text
Dispose whendone	long Prompt	Rule Type
Distinguishing	Owning FCB's	
Features	Print Name	
Dynamic Rule Order	Procedure Arguments	
Enter Value Screen	Procedure Name	
External Value	Prompt	
Screen	Screen	
External Data	Sourcing Sequence	
Goals	Value Can Change	
Initial Data	Flag	
Initial Query		
Max Instances		
Multiple choice		
screen		
Name		
Parameters		
Parent		
Print Name		
Results		
Rules		

APPENDIX B

INNOVATOR

PROGRAM LISTING

```
CONTROLCHARACTER
INONUT 12/18/89 09:27:22
 TITLE
 PARAMETER EGF_SCALE
 PROP Constraint
= ('Excellent', 'Good', 'Fair', 'Poor', 'Very Poor')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PROP Comment
This is a 5 point scale for evaluations
 PARAMETER LMV_SCALE
 PROP Constraint
= ('Very High', 'High', 'Average', 'Low', 'Very Low')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PROP Comment
This is a 5 point scale for evaluations
 PARAMETER PLINE_LST
 PROP Constraint
= ('Annuities', 'Bonds', 'Insurance', 'Mutual_Funds',
 'Options', 'Precious Metals', 'Stocks',
 'Real_Estate_Partnerships')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER MF_PROD_LST
 PROP Constraint
= ('Balanced_Funds', 'Bond_Funds', 'Convertible_Funds',
 'Growth_Funds', 'Growth_and_Income_Funds',
 'Income_Funds', 'International_Funds',
 'Money_Market_Funds', 'Sector/Specialty_Funds')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
```

```
FALSE
 PARAMETER PL_M_SIZE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the total possible size of the market for this
 product line (As a percentage of the market for this t
ype of product).
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = m_size
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_M_GRW_RATE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the annual percentage rate of growth of the ma
rket for the product line.
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = m_growth_rate
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_M_S_ECON
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Prompt
Estimate the susceptibility of the market for the produ
ct line to changes in the economy.
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = m_suscep_econ
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_M_SEASON
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the seasonality of demand for this product lin
e.
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = m_season
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_F_VOL
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the total volume of business from this product
 line (in Millions of Dollars).
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = f_vol
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_F_GMARGIN
 PROP Constraint
```

```
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the Gross Margin of this Product Line (as a pe
rcentage).
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_mf_eval
COLUMN     = f_g_margin
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_F_RETURN
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the Return on Investment for this Product Line
(as a percentage).
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_usr_log
COLUMN     = f_return
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_F_PAYBACK
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate this Product Line's Pay Back Period in years.
 PROP Screen
SCREEN:STRENT
```

```
 PROP Prompt
How easy is it to imitate the features of this Product
Line?
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = p_imitation
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_P_IMAGE
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What contribution does this Product Line make to the Im
age of the company which offers it?
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = p_image
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORP_COMP
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How high a compatiblity does this product line have
with other product lines offered by the company?
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = corp_compat
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORP_OUTLAY
 PROP Constraint
is a number
```

```
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the extent of the Financial Outlay to offer thi
s product line (in Millions of Dollars)?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = corp_outlay
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORP_ORG
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What degree of change to the Organizational Structure w
ill be necessary to support this product line?
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = corp_org
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORP_PER_MARKETING
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the level of retraining and/or hiring necessar
y for this company to provide marketing support for thi
s product line.
 PROP Procedure name
SQL
 PROP Procedure args
```

```
TABLE     = profile_usr_log
COLUMN    = corp_per_mark
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORP_PER_TECH
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Estimate the degree of retraining and/or hiring necessa
ry in this company to support the product line's techni
cal development.
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = corp_per_tech
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_COMP_QUALITY
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Describe the quality of competing offerings in this Pro
duct line.
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = comp_quality
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_COMP_VULNERABILITY
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Prompt
What level of vulnerability do the competing
firms in this product line have to a new product offeri
ng?
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_usr_log
COLUMN    = comp_vulnerability
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER P_PAST_1YR
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the performance of the product over the last
year with the performance of products of the same type
or with the performance of substitutes for the product.
 PROP Val can chg flg
FALSE
 PARAMETER P_PAST_5YR
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the performance of the product over the last fi
ve years with the performance of other products of the
same type or of substitutes for the product.
 PROP Val can chg flg
FALSE
 PARAMETER P_PAST_10YR
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the performance of the product over the last te
n years with the performance of other products of the s
ame type or substitutes for the product.
```

```
 PROP Val can chg flg
FALSE
 PARAMETER P_FUTURE
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What are the future prospects of this product.
 PROP Val can chg flg
FALSE
 PARAMETER P_RISK
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the Risk rating of this product to other produc
ts of this type or substitutes for this product.
 PROP Val can chg flg
FALSE
 PARAMETER DB_ST_AGE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How old is the company offering this product (in years)

 PROP Val can chg flg
FALSE
 PARAMETER DB_ST_REPUTATION
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What reputation does the company offering this product
have in the industry?
```

```
 PROP Val can chg flg
FALSE
 PARAMETER DB_ST_TURNOVER
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the management turnover in the company offering
 the product with turnover in the industry as a whole.
 PROP Val can chg flg
FALSE
 PARAMETER DB_SIZE_NUMBER
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How many products of this type does this company offer?
 PROP Val can chg flg
FALSE
 PARAMETER DB_SIZE_RANGE
 PROP Constraint
taken from pline_lst; multivalued
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What product lines are offered by this company?
 PROP Val can chg flg
FALSE
 PARAMETER DB_SIZE_ASSETS
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the value of the total assets of the company in
 Millions of dollars?
 PROP Val can chg flg
```

```
FALSE
 PARAMETER DB_EARN_COMMISSION
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What percentage do employees of the company recieve as
a commission?
 PROP Val can chg flg
FALSE
 PARAMETER DB_EARN_MIN_AMOUNT
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the minimum amount in millions of dollars
needed to invest in this companys product?
 PROP Val can chg flg
FALSE
 PARAMETER DB_PAST_GROWTH
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the growth rate of this company's assets to the
 average growth rate of assets in the industry.
 PROP Val can chg flg
FALSE
 PARAMETER DB_PAST_RETURN
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Compare the average return on investment of products of
fered by this company with the industry average.
```

```
 PROP Val can chg flg
FALSE
 PARAMETER DB_PAST_TOP20
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How many products offered by this company have been rat
ed in the industry's top 20?
 PROP Val can chg flg
FALSE
 PARAMETER DB_QUAL_SWITCH
 PROP Constraint
= ('Unlimited Switching', 'Unlimited Switching/Charge p
er switch', 'Limited Switching', 'Limited Switching/Cha
rge per switch', 'No Switching')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What type of switching between company products are ava
ilable to investors?
 PROP Val can chg flg
FALSE
 PARAMETER DB_QUAL_AVAILABILITY
 PROP Constraint
is a boolean
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Is a service representative available for consultation
24 hours a day?
 PROP Val can chg flg
FALSE
 PARAMETER DB_QUAL_RELATIONSHIP
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Prompt
Rate the relationship between the company and the broke
rs/agents employed by it.
 PROP Val can chg flg
FALSE
 PARAMETER ASK_PL
 PROP Constraint
taken from pline_lst
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please choose a Product Line to consider.
 PROP Screen
SCREEN:CHOSEI
 PROP Val can chg flg
FALSE
 PARAMETER ASK_P_MF
 PROP Constraint
taken from mf_prod_lst
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please choose a Mutual Fund Type to consider.
 PROP Val can chg flg
FALSE
 PARAMETER EASE_SCALE
 PROP Constraint
= ('Very Easy', 'Easy', 'Average', 'Hard', 'Very Hard')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PROP Comment
This is a 5 point scale for evaluations
 PARAMETER ERCODE
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER RPL_CORP_OUTLAY
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION = 'profile_name = :vq *rpl_name AND
parameter_name = :vq *dpl_corp_outlay_n'
 PROP Val can chg flg
FALSE
 PARAMETER FUSERNAME
 PROP Constraint
= username
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter your name (30 characters or less).
 PROP Screen
SCREEN:STR[illegible]
 PROP Pro   ure name
SQL
 PROP Procedure args
TABLE      = result_log
COLUMN     = couser
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER FUSERCO
 PROP Constraint
= userco
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter the name of your Company/Firm
(30 characters or less).
 PROP Screen
SCREEN:STRENTI
```

```
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = result_log
COLUMN    = company
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER RPL_F_GMARGIN
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION = 'profile_name = :vq *rpl_name AND
parameter_name = :vq *dpl_f_gmargin_n'
 PROP Val can chg flg
FALSE
 PARAMETER RPL_F_PAYBACK
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION = 'profile_name = :vq *rpl_name and
parameter_name = :vq *dpl_f_payback_n'
 PROP Val can chg flg
FALSE
 PARAMETER RPL_F_RETURN
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
```

```
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION = 'profile_name = :vq *rpl_name and
parameter_name = :vq *dpl_f_return_n'
 PROP Val can chg flg
FALSE
 PARAMETER RPL_F_VOL
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION = 'profile_name = :vq *rpl_name and
parameter_name = :vq *dpl_f_vol_n'
 PROP Val can chg flg
FALSE
 PARAMETER RPL_M_GRW_RATE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION = 'profile_name = :vq *rpl_name and
parameter_name = :vq *dpl_m_grw_rate_n'
 PROP Val can chg flg
FALSE
 PARAMETER TABLE_VALUE
 PROP Constraint
taken from table_lst
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_M_SIZE
```

```
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION = 'profile_name = :vq *rpl_name and
parameter_name = :vq *dpl_m_grw_rate_n'
 PROP Val can chg flg
FALSE
 PARAMETER TABLE_LST
 PROP Constraint
= ('profile_pl_eval', 'result_log')
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER DATETIME
 PROP Constraint
= ':date1 /:time1'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = result_log
COLUMN     = datetime
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER FRESULT
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Screen
```

```
SCREEN TITLE
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = result_log
COLUMN     = result
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER MF_PL_MESSAGE_D_EVAL
 PROP Constraint
= 'The data for this factor has been DISAPPROVED by INN
OVATOR. This will contribute to the new product being
disapproved.'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER DPL_CORP_OUTLAY
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the maximum extent, in millions of dollars,
of the financial outlay your company would make for a n
ew product line?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_GMARGIN
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
```

```
What minimum gross margin is acceptable to your company
 for a new product line?  (estimate as a percentage)
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_PAYBACK
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What payback period (in years) would your company find
acceptable for a new product line?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_RETURN
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What minimum return on investment is acceptable in your
 company for a new product line (in millions of dollars
)?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION =
```

```
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_VOL
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What minimum total volume of business does your company
 desire from a new product line (in millions of dollars
)?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_M_GRW_RATE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What minimum annual growth rate of the market for
 a new product line is acceptable to your company (as a
 percentage)?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_value
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_M_SIZE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Prompt
What is the minimum market share acceptable to your com
pany for a new product line? (as a percentage)?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_value
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER ASK_TYPE
 PROP Constraint
taken from type_lst
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What do you wish to evaluate?
 PROP Screen
SCREEN:CHOSEI
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = result_log
COLUMN     = profile_type
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER TYPE_LST
 PROP Constraint
= ('Product Line', 'Product', 'Brand')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Screen
SCREEN:CHOSE
 PROP Val can chg flg
FALSE
 PARAMETER PROFILE_OLD
 PROP Constraint
taken from ext_lst
 PROP Sourcing seq.
  User will input from terminal
```

```
 PROP Prompt
Your company maintains the following profiles of desira
ble Product Lines:
:CE ON
:VL *profl_lst
:CE OFF
Do you wish to use one of these profiles or do you wish
 to create a new profile?
 PROP Screen
SCREEN:CHOSEI
 PROP Val can chg flg
FALSE
 PARAMETER PROFL_LST
 PROP Constraint
is a string; multivalued
 PROP Sourcing seq.
  Rule Consequent



 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = profile_name
CONDITION = 'company = :vq *userco'
 PROP Val can chg flg
FALSE
 PARAMETER USERNAME
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter your name (30 characters or less).
 PROP Screen
SCREEN:STRENTI
 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = couser
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_NAME
 PROP Constraint
```

```
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What will the name of this Product Line Profile be?
 PROP Screen
SCREEN:STRENT
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = profile_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DBERST
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER EXT_LST
 PROP Constraint
= ('Existing Profile', 'New Profile')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_NAME
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_M_VALUE
 PROP Constraint
taken from ian_scale
```

```
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
In evaluating a new product line, how important to your
 organization is the Market factor?
 PROP Val can chg flg
FALSE
 PARAMETER PL_F_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
In evaluating a new product line, how important to your
 organization is the Finance factor?
 PROP Val can chg flg
FALSE
 PARAMETER PL_P_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
In evaluating a new product line, how important to your
 organization is the Product factor?
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORP_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
In evaluating a new product line, how important to your
 organization is the Corporate factor?
 PROP Val can chg flg
FALSE
 PARAMETER PL_COMP_VALUE
 PROP Constraint
```

```
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
In evaluating a new product line, how important to your
 organization is the Competitor factor?
 PROP Val can chg flg
FALSE
 PARAMETER FINAL_SCALE
 PROP Constraint
= ('Approve', 'Approve Marginally', 'Reevaluate the dat
a and resubmit', 'Disapprove')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_FINAL_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Screen
SCREEN:TITLE
 PROP Val can chg flg
FALSE
 PARAMETER PL_MV_SEASON
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_MV_S_ECON
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER RPL_MV_S_ECON
 PROP Constraint
= 2
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_MV_SEASON
 PROP Constraint
= 2
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER USERTITLE
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter the title of your position in your Company
/Firm (30 characters or less).
 PROP Screen
SCREEN:STRENTI
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = result_log
COLUMN    = cotitle
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER PL_MARKET_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent
```

```
 PROP Val can chg flg
FALSE
 PARAMETER PL_FINANCE_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER MF_PL_FINMSG_A_EVAL
 PROP Constraint
= 'The New Product has been APPROVED by INNOVATOR.'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_PV_APPEAL
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_PV_DIST
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_PV_IMAGE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER PL_PV_IMITATION
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_PV_APPEAL
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_PV_DIST
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_PV_IMAGE
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_PV_IMITATION
 PROP Constraint
= 2
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORPV_COMP
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_CORPV_ORG
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_CORPV_PER_MARKETING
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_CORPV_PER_TECH
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_CORPV_COMP
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER RPL_CORPV_ORG
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_CORPV_PER_MARKETING
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_CORPV_PER_TECH
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_COMPV_QUALITY
 PROP Constraint
= 3
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RPL_COMPV_VULNERABILITY
 PROP Constraint
= 4
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER PL_COMPV_QUALITY
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_COMPV_VULNERABILITY
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_FAC_INT_FLAG
 PROP Constraint
is a number
 PROP Sourcing seq.
  Default will be taken
  Rule Consequent



 PROP Val can chg flg
TRUE
 PARAMETER MF_PL_FINMSG_AM_EVAL
 PROP Constraint
= 'The New Product has been MARGINALLY APPROVED by INNO
VATOR.'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER MF_PL_FINMSG_RE_EVAL
 PROP Constraint
= 'The data for this New Product if to ambiguous for IN
NOVATOR to approve or disapprove.  Please re-evaluate t
he information of the factors included and resubmit the
 information to INNOVATOR at your earliest opportunity.
```

```
,
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER MF_PL_FINMSG_D_EVAL
 PROP Constraint
= 'The New Product has been DISAPPROVED by INNOVATOR.'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER PL_FINAL_FLAG
 PROP Constraint
is a number
 PROP Sourcing seq.
  Default will be taken
  Rule Consequent



 PROP Default constr.
= 0
 PROP Val can chg flg
TRUE
 PARAMETER PL_INT_FLAG
 PROP Constraint
is a number
 PROP Sourcing seq.
  Default will be taken
  Rule Consequent



 PROP Default constr.
= 0
 PROP Val can chg flg
FALSE
 PARAMETER PL_PRODUCT_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent
```

```
 PROP Val can chg flg
FALSE
 PARAMETER PL_CORPORATE_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER PL_COMPETITOR_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER MF_PL_MESSAGE_RE_EVAL
 PROP Constraint
= 'The data for this factor is too ambiguous for INNOVA
TOR to approve or disapprove.  It will not contribute t
o the approval of the new product.  Please re-evaluate
the information this factor considers and re-submit the
 information to INNOVATOR at your earliest oppurtunity.
'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER MF_PL_MESSAGE_AM_EVAL
 PROP Constraint
= 'The data for this factor has been MARGINALLY APPROVE
D by INNOVATOR. This could contribute to the new produc
t being approved.'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER B_EARN_COMMISSION
 PROP Constraint
```

```
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What commission rate can be earned on this brand?
 PROP Val can chg flg
FALSE
 PARAMETER B_EARN_MIN_AMOUNT
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the minimum investment amount for this brand?
 PROP Val can chg flg
FALSE
 PARAMETER B_PAST_GROWTH
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What rate of growth have the assets of this brand susta
ined in the past?
 PROP Val can chg flg
FALSE
 PARAMETER B_PAST_RETURN
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the brands performance on returns compared to t
he industry average?
 PROP Val can chg flg
FALSE
 PARAMETER B_PAST_TOP20
 PROP Constraint
is a number
```

```
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How many funds of this brand have been in the industry
top 20?
 PROP Val can chg flg
FALSE
 PARAMETER B_QUAL_AVAILABILITY
 PROP Constraint
is a boolean
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Is a service representative available 24 hours a day?
 PROP Val can chg flg
FALSE
 PARAMETER B_QUAL_RELATIONSHIP
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Is the relationship between brokers and agents and the
rest of the brand's firm a good one?
 PROP Val can chg flg
FALSE
 PARAMETER B_QUAL_SWITCH
 PROP Constraint
= ('Unlimited Switching', 'Unlimited Switching/Charge p
er switch', 'Limited Switching', 'Limited Switching/Cha
rge per switch', 'No Switching')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Which facilities for switching fund are supported in th
is brand?
 PROP Val can chg flg
FALSE
 PARAMETER B_SIZE_ASSETS
```

```
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What are the total assets available to the company offe
ring this brand?
 PROP Val can chg flg
FALSE
 PARAMETER B_SIZE_NUMBER
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How many funds are offered by the company?
 PROP Val can chg flg
FALSE
 PARAMETER B_SIZE_RANGE
 PROP Constraint
taken from pline_lst
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Which of these types of financial services does this co
mpany offer?
 PROP Val can chg flg
FALSE
 PARAMETER B_ST_AGE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How many years has this company existed?
 PROP Val can chg flg
FALSE
 PARAMETER B_ST_REPUTATION
 PROP Constraint
```

```
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What reputation does this company have in the industry?
 PROP Val can chg flg
FALSE
 PARAMETER B_ST_TURNOVER
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
How would you characterize the managemant turnover in t
his company?
 PROP Val can chg flg
FALSE
 PARAMETER RB_EARN_COMMISSION
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_EARN_MIN_AMOUNT
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_PAST_GROWTH
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER RB_PAST_RETURN
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_PAST_TOP20
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_QUAL_AVAILABILITY
 PROP Constraint
is a boolean
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_QUAL_RELATIONSHIP
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_QUAL_SWITCH
 PROP Constraint
= ('Unlimited Switching', 'Unlimited Switching/Charge p
er switch', 'Limited Switching', 'Limited Switching/Cha
rge per switch', 'No Switching')
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
```

```
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_SIZE_ASSETS
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_SIZE_NUMBER
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_SIZE_RANGE
 PROP Constraint
taken from pline_lst
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_ST_AGE
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_ST_REPUTATION
 PROP Constraint
taken from egf_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
```

```
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_ST_TURNOVER
 PROP Constraint
taken from lmv_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER B_PASTV_GROWTH
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER B_PASTV_RETURN
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER B_QUALV_RELATIONSHIP
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER B_STV_REPUTATION
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
```

```
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER B_STV_TURNOVER
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER RB_NAME
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER DB_NAME
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter the brand name in thirty characters or les
s.
 PROP Val can chg flg
FALSE
 PARAMETER MF_B_EARN_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the relative importance of the earnings associa
ted with the brand name?
 PROP Val can chg flg
FALSE
```

```
 PARAMETER MF_B_PAST_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the relative importance of the past performance
 of this brand?
 PROP Val can chg flg
FALSE
 PARAMETER MF_B_QUAL_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the relative importance of the quality of the b
rand name?
 PROP Val can chg flg
FALSE
 PARAMETER MF_B_SIZE_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the relative importance of the size of the bran
d?
 PROP Val can chg flg
FALSE
 PARAMETER MF_B_ST_VALUE
 PROP Constraint
taken from ian_scale
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
What is the relative importance of the brand's companys
 standing in the industry?
 PROP Val can chg flg
```

```
FALSE
 PARAMETER B_FINAL_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER B_EARN_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER B_PAST_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER B_QUAL_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER B_SIZE_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
```

```
FALSE
 PARAMETER B_ST_EVAL
 PROP Constraint
taken from final_scale
 PROP Sourcing seq.
  Rule Consequent



 PROP Val can chg flg
FALSE
 PARAMETER USERCO
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter the name of your Company/Firm
(30 characters or less).
 PROP Screen
SCREEN:STRENTI
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = company
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER NAMEHOLD
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = result_log
COLUMN    = profile_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_CORP_OUTLAY_N
 PROP Constraint
= 'corp_financial_outlay'
 PROP Sourcing seq.
```

```
   Rule Consequent
   User will input from terminal
   Default will be taken


  PROP Procedure name
SQL
  PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_name
CONDITION =
  PROP Val can chg flg
FALSE
  PARAMETER RB_PASTV_GROWTH
  PROP Constraint
is a number
  PROP Sourcing seq.
   Rule Consequent
   User will input from terminal
   Default will be taken


  PROP Val can chg flg
FALSE
  PARAMETER RB_PASTV_RETURN
  PROP Constraint
is a number
  PROP Sourcing seq.
   Rule Consequent
   User will input from terminal
   Default will be taken


  PROP Val can chg flg
FALSE
  PARAMETER RB_QUALV_RELATIONSHIP
  PROP Constraint
is a number
  PROP Sourcing seq.
   Rule Consequent
   User will input from terminal
   Default will be taken


  PROP Val can chg flg
FALSE
  PARAMETER RB_STV_REPUTATION
  PROP Constraint
is a number
  PROP Sourcing seq.
   Rule Consequent
   User will input from terminal
   Default will be taken
```

```
 PROP Val can chg flg
FALSE
 PARAMETER RB_STV_TURNOVER
 PROP Constraint
is a number
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_GMARGIN_N
 PROP Constraint
= 'financial_gross_margin'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_PAYBACK_N
 PROP Constraint
= 'financial_payback_period'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_RETURN_N
 PROP Constraint
= 'financial_return_on_investment'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken
```

```
 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_F_VOL_N
 PROP Constraint
= 'financial_volume_of_business'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = profile_pl_eval
COLUMN    = parameter_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DP_NAME
 PROP Constraint
is a string
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Prompt
Please enter the product name in thirty characters or
less.
 PROP Val can chg flg
FALSE
 PARAMETER EX_PROFILE_PL
 PROP Constraint
taken from profl_lst
 PROP Sourcing seq.
  User will input from terminal



 PROP Prompt
Which Product Line Profile would you care to use?
 PROP Screen
SCREEN:CHOSEI
 PROP Procedure name
SQL
```

```
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = profile_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_M_GRW_RATE_N
 PROP Constraint
= 'market_growth_rate'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER DPL_M_SIZE_N
 PROP Constraint
= 'market_size'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Procedure name
SQL
 PROP Procedure args
TABLE      = profile_pl_eval
COLUMN     = parameter_name
CONDITION =
 PROP Val can chg flg
FALSE
 PARAMETER FDATETIME
 PROP Constraint
= datetime
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PARAMETER IAN_SCALE
 PROP Constraint
= ('Very Important', 'Important', 'Average', 'Not Impor
tant', 'Not At All Important')
```

```
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Val can chg flg
FALSE
 PROP Comment
This is a 5 point scale for evaluations
 PARAMETER MF_PL_MESSAGE_A_EVAL
 PROP Constraint
= 'The data for this factor has been APPROVED by INNOVA
TOR.  This will contribute to the new porduct being app
roved.'
 PROP Sourcing seq.
  Rule Consequent
  User will input from terminal
  Default will be taken


 PROP Screen
SCREEN:PROFDISP
 PROP Val can chg flg
FALSE
 PARAMETER MF_B_PR_OLD
 PROP Constraint
taken from ext_lst
 PROP Sourcing seq.
  User will input from terminal



 PROP Prompt
Will you want to use an Existing Profile, or enter a Ne
w Profile?
 PROP Val can chg flg
FALSE
 PARAMETER MF_B_PR_LST
 PROP Constraint
is a string; multivalued
 PROP Sourcing seq.
  Rule Consequent



 PROP Procedure name
SQL
 PROP Procedure args
TABLE     = pr_brnd_mf_eval
COLUMN    = name
CONDITION =
 PROP Val can chg flg
```

```
FALSE
 RULE PL_EVAL_6
 PROP Rule text
fif certainty of (pl_finance_eval) <= .0 and
certainty of (pl_market_eval) <= .0 and
certainty of (pl_corporate_eval) <= .0 and
certainty of (pl_competitor_eval) <  .0 and
certainty of (pl_product_eval) <= .0
then establish pl_fin_msg_d_eval immediate and
pl_final_flag = 4
 PROP Rule type
  Inference
 RULE PL_EVAL_2A
 PROP Rule text
fif certainty of (pl_finance_eval) > .5 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) > .5 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) > .5 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) > .5 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_corporate_eval) > .5 and
```

```
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5

then establish pl_fin_msg_am_eval immediate and
```

```
pl_final_flag = 2
 PROP Rule type
  Inference
 RULE PL_EVAL_4
 PROP Rule text
fif certainty of (pl_finance_eval) < .2 and
certainty of (pl_finance_eval) > .0 and
certainty of (pl_market_eval) < .2 and
certainty of (pl_market_eval) > .0 and
certainty of (pl_corporate_eval) < .2 and
certainty of (pl_corporate_eval) > .0 and
certainty of (pl_competitor_eval) < .2 and
certainty of (pl_competitor_eval) > .0 and
certainty of (pl_product_eval) < .2 and
certainty of (pl_product_eval) > .0

then establish pl_fin_msg_re_eval immediate and
pl_final_flag = 3
 PROP Rule type
  Inference
 RULE PL_LOG_RESULT
 PROP Rule text
if ask_type = 'Product Line'
then fresult = pl_final_eval
 PROP Rule type
  Inference
 RULE GET_NEW_PROFILE
 PROP Rule text
if profile_old is 'New Profile'
then establish new_profile immediate
 PROP Rule type
  Inference
 RULE GET_OLD_PROFILE
 PROP Rule text
if profile_old is not 'New Profile'
then establish ask_old immediate
 PROP Rule type
  Inference
 RULE NEW_ROW
 PROP Rule text
if profile_old is 'New Profile'
then rpl_name = dpl_name
 PROP Rule type
  Inference
 RULE OLD_ROW
 PROP Rule text
if profile_old is not 'New Profile'
then rpl_name = ex_profile_pl
 PROP Rule type
  Inference
 RULE PL_VH_FIND
 PROP Rule text
if 1 of the following (pl_m_value = 'Very Important',
pl_f_value = 'Very Important',
pl_p_value = 'Very Important',
```

```
pl_comp_value = 'Very Important',
pl_corp_value = 'Very Important')
then establish pl_vh_figure immediate
 PROP Rule type
  Inference
 RULE PL_H_FIND
 PROP Rule text
if 1 of the following (pl_m_value = 'Important',
pl_f_value = 'Important', pl_p_value = 'Important',
pl_comp_value = 'Important', pl_corp_value = 'Important
') then establish pl_h_figure immediate
 PROP Rule type
  Inference
 RULE PL_MOD_FIND
 PROP Rule text
if 1 of the following (pl_m_value = 'Average',
pl_f_value = 'Average', pl_p_value = 'Average',
pl_comp_value = 'Average', pl_corp_value = 'Average')
then establish pl_mod_figure immediate
 PROP Rule type
  Inference
 RULE PL_L_FIND
 PROP Rule text
if 1 of the following (pl_m_value = 'Not Important',
pl_f_value = 'Not Important', pl_p_value = 'Not Importa
nt', pl_comp_value = 'Not important',
pl_corp_value = 'Not Important')
then establish pl_l_figure immediate
 PROP Rule type
  Inference
 RULE PL_VL_FIND
 PROP Rule text
if 1 of the following (pl_m_value = 'Not At All Importa
nt', pl_f_value = 'Not At All Important',
pl_p_value = 'Not At All Important',
pl_comp_value = 'Not At All Important',
pl_corp_value = 'Not At All Important')
then establish pl_vl_figure immediate
 PROP Rule type
  Inference
 RULE PL_M_VH
 PROP Rule text
if pl_m_value = 'Very Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE PL_F_VH
 PROP Rule text
if pl_f_value = 'Very Important'


then establish plf_finance_eval
 PROP Rule type
```

```
  Inference
 RULE PL_P_VH
 PROP Rule text
if pl_p_value = 'Very Important'


then establish plf_product_eval
 PROP Rule type
  Inference
 RULE PL_COMP_VH
 PROP Rule text
if pl_comp_value = 'Very Important'


then establish plf_competitors_eval
 PROP Rule type
  Inference
 RULE PL_CORP_VH
 PROP Rule text
if pl_corp_value = 'Very Important'


then establish plf_corporate_eval
 PROP Rule type
  Inference
 RULE PL_M_H
 PROP Rule text
if pl_m_value = 'Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE PL_F_H
 PROP Rule text
if pl_f_value = 'Important'


then establish plf_finance_eval
 PROP Rule type
  Inference
 RULE PL_P_H
 PROP Rule text
if pl_p_value = 'Important'


then establish plf_product_eval
 PROP Rule type
  Inference
 RULE PL_COMP_H
 PROP Rule text
if pl_comp_value = 'Important'


then establish plf_competitors_eval
```

```
 PROP Rule type
  Inference
 RULE PL_CORP_H
 PROP Rule text
if pl_corp_value = 'Important'


then establish plf_corporate_eval
 PROP Rule type
  Inference
 RULE PL_M_MOD
 PROP Rule text
if pl_m_value = 'Average'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE PL_F_MOD
 PROP Rule text
if pl_f_value = 'Average'


then establish plf_finance_eval
 PROP Rule type
  Inference
 RULE PL_COMP_MOD
 PROP Rule text
if pl_comp_value = 'Average'


then establish plf_competitors_eval
 PROP Rule type
  Inference
 RULE PL_CORP_MOD
 PROP Rule text
if pl_corp_value = 'Average'


then establish plf_corporate_eval
 PROP Rule type
  Inference
 RULE PL_P_MOD
 PROP Rule text
if pl_p_value = 'Average'


then establish plf_product_eval
 PROP Rule type
  Inference
 RULE PL_M_L
 PROP Rule text
if pl_m_value = 'Not Important'
```

```
then establish plf_market_eval
 PROP Rule type
  Inference
 RULE PL_F_L
 PROP Rule text
if pl_f_value = 'Not Important'


then establish plf_finance_eval
 PROP Rule type
  Inference
 RULE PL_P_L
 PROP Rule text
if pl_p_value = 'Not Important'


then establish plf_product_eval
 PROP Rule type
  Inference
 RULE PL_COMP_L
 PROP Rule text
if pl_comp_value = 'Not Important'


then establish plf_competitors_eval
 PROP Rule type
  Inference
 RULE PL_CORP_L
 PROP Rule text
if pl_corp_value = 'Not Important'


then establish plf_corporate_eval
 PROP Rule type
  Inference
 RULE PL_M_VL
 PROP Rule text
if pl_m_value = 'Not At All Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE PL_F_VL
 PROP Rule text
if pl_f_value = 'Not At All Important'


then establish plf_finance_eval
 PROP Rule type
  Inference
 RULE PL_P_VL
 PROP Rule text
if pl_p_value = 'Not At All Important'
```

```
then establish plf_product_eval
 PROP Rule type
  Inference
 RULE PL_COMP_VL
 PROP Rule text
if pl_comp_value = 'Not At All Important'


then establish plf_competitors_eval
 PROP Rule type
  Inference
 RULE PL_CORP_VL
 PROP Rule text
if pl_corp_value = 'Not At All Important'


then establish plf_corporate_eval
 PROP Rule type
  Inference
 RULE PL_SEASON_VH_MV
 PROP Rule text
if pl_m_season is 'Very High'
then pl_mv_season = 5
 PROP Rule type
  Inference
 RULE PL_SEASON_H_MV
 PROP Rule text
if pl_m_season is 'High'
then pl_mv_season = 4
 PROP Rule type
  Inference
 RULE PL_SEASON_MOD_MV
 PROP Rule text
if pl_m_season is 'Average'
then pl_mv_season = 3
 PROP Rule type
  Inference
 RULE PL_SEASON_L_MV
 PROP Rule text
if pl_m_season is 'Low'
then pl_mv_season = 2
 PROP Rule type
  Inference
 RULE PL_SEASON_VL_MV
 PROP Rule text
if pl_m_season is 'Very Low'
then pl_mv_season = 1
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_PAM_EVAL
 PROP Rule text
if certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2
then establish pl_msg_am_eval and
```

```
pl_fac_int_flag = 2
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_PRE_EVAL
 PROP Rule text
if certainty of (pl_product_eval) <= .2 and
certainty of (pl_product_eval) > .0
then establish pl_msg_re_eval and
pl_fac_int_flag = 3
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_PD_EVAL
 PROP Rule text
if certainty of (pl_product_eval) < .0
then establish pl_msg_d_eval and
pl_fac_int_flag = 4
 PROP Rule type
  Inference
 RULE PL_S_ECON_VH_MV
 PROP Rule text
if pl_m_s_econ is 'Very High'
then pl_mv_s_econ = 5
 PROP Rule type
  Inference
 RULE PL_S_ECON_H_MV
 PROP Rule text
if pl_m_s_econ is 'High'
then pl_mv_s_econ = 4
 PROP Rule type
  Inference
 RULE PL_S_ECON_MOD_MV
 PROP Rule text
if pl_m_s_econ is 'Average'
then pl_mv_s_econ = 3
 PROP Rule type
  Inference
 RULE PL_S_ECON_L_MV
 PROP Rule text
if pl_m_s_econ is 'Low'
then pl_mv_s_econ = 2
 PROP Rule type
  Inference
 RULE PL_S_ECON_VL_MV
 PROP Rule text
if pl_m_s_econ is 'Very Low'
then pl_mv_s_econ = 1
 PROP Rule type
  Inference
 RULE PL_F_GMARGIN_A_EVAL
 PROP Rule text
if pl_f_gmargin >= rpl_f_gmargin
then there is some evidence that pl_finance_eval is
'approve'
 PROP Rule type
  Inference
```

```
 RULE PL_F_GMARGIN_D_EVAL
 PROP Rule text
if pl_f_gmargin < rpl_f_gmargin
then there is some negative evidence that
pl_finance_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_F_PAYBACK_A_EVAL
 PROP Rule text
if pl_f_payback <= rpl_f_payback
then there is some evidence that pl_finance_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_F_PAYBACK_D_EVAL
 PROP Rule text
if pl_f_payback > rpl_f_payback
then there is some negative evidence that
pl_finance_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_F_RETURN_A_EVAL
 PROP Rule text
if pl_f_return >= rpl_f_return
then there is some evidence that pl_finance_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_F_RETURN_D_EVAL
 PROP Rule text
if pl_f_return < rpl_f_return
then there is some negative evidence that
pl_finance_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_F_VOL_A_EVAL
 PROP Rule text
if pl_f_vol >= rpl_f_vol
then there is some evidence that pl_finance_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_F_VOL_D_EVAL
 PROP Rule text
if pl_f_vol < rpl_f_vol
then there is some negative evidence that
pl_finance_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_M_GRW_RATE_A_EVAL
 PROP Rule text
if pl_m_grw_rate >= rpl_m_grw_rate
then there is some evidence that pl_market_eval is
'approve'
 PROP Rule type
```

```
  Inference
 RULE PL_M_GRW_RATE_D_EVAL
 PROP Rule text
if pl_m_grw_rate < rpl_m_grw_rate
then there is some negative evidence that
pl_market_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_M_SIZE_A_EVAL
 PROP Rule text
if pl_m_size >= rpl_m_size
then there is some evidence that pl_market_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_M_SIZE_D_EVAL
 PROP Rule text
if pl_m_size < rpl_m_size
then there is some negative evidence that
pl_market_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_M_SEASON_A_EVAL
 PROP Rule text
if pl_mv_season <= rpl_mv_season
then there is some evidence that pl_market_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_M_SEASON_D_EVAL
 PROP Rule text
if pl_mv_season > rpl_mv_season
then there is some negative evidence that
pl_market_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_M_S_ECON_A_EVAL
 PROP Rule text
if pl_mv_s_econ <= rpl_mv_s_econ
then there is some evidence that pl_market_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_M_S_ECON_D_EVAL
 PROP Rule text
if pl_mv_s_econ > rpl_mv_s_econ
then there is some negative evidence that
pl_market_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_EVAL_1
 PROP Rule text
fif certainty of (pl_finance_eval) > .5 and
certainty of (pl_market_eval) > .5 and
certainty of (pl_corporate_eval) > .5 and
```

```
certainty of (pl_competitor_eval) > .5 and
certainty of (pl_product_eval) > .5
then establish pl_fin_msg_a_eval immediate and
pl_final_flag = 1
 PROP Rule type
  Inference
 RULE PL_EVAL_2
 PROP Rule text
fif certainty of (pl_finance_eval) > .5 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) > .5 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) > .5 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_market_eval) > .5 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .2 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .2 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .2 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2 and
certainty of (pl_product_eval) > .5
then establish pl_fin_msg_am_eval immediate and
pl_final_flag = 2
```

```
 PROP Rule type
  Inference
 RULE PL_APPEAL_VH_PV
 PROP Rule text
if pl_p_appeal is 'Very High'
then pl_pv_appeal = 5
 PROP Rule type
  Inference
 RULE PL_APPEAL_H_PV
 PROP Rule text
if pl_p_appeal is 'High'
then pl_pv_appeal = 4
 PROP Rule type
  Inference
 RULE PL_APPEAL_MOD_PV
 PROP Rule text
if pl_p_appeal is 'Average'
then pl_pv_appeal = 3
 PROP Rule type
  Inference
 RULE PL_APPEAL_L_PV
 PROP Rule text
if pl_p_appeal is 'Low'
then pl_pv_appeal = 2
 PROP Rule type
  Inference
 RULE PL_APPEAL_VL_PV
 PROP Rule text
if pl_p_appeal is 'Very Low'
then pl_pv_appeal = 1
 PROP Rule type
  Inference
 RULE PL_DIST_VH_PV
 PROP Rule text
if pl_p_dist is 'Very High'
then pl_pv_dist = 5
 PROP Rule type
  Inference
 RULE PL_DIST_H_PV
 PROP Rule text
if pl_p_dist is 'High'
then pl_pv_dist = 4
 PROP Rule type
  Inference
 RULE PL_DIST_MOD_PV
 PROP Rule text
if pl_p_dist is 'Average'
then pl_pv_dist = 3
 PROP Rule type
  Inference
 RULE PL_DIST_L_PV
 PROP Rule text
if pl_p_dist is 'Low'
then pl_pv_dist = 2
 PROP Rule type
```

```
  Inference
 RULE PL_DIST_VL_PV
 PROP Rule text
if pl_p_dist is 'Very Low'
then pl_pv_dist = 1
 PROP Rule type
  Inference
 RULE PL_IMAGE_VH_PV
 PROP Rule text
if pl_p_image is 'Very High'
then pl_pv_image = 5
 PROP Rule type
  Inference
 RULE PL_IMAGE_H_PV
 PROP Rule text
if pl_p_image is 'High'
then pl_pv_image = 4
 PROP Rule type
  Inference
 RULE PL_IMAGE_MOD_PV
 PROP Rule text
if pl_p_image is 'Average'
then pl_pv_image = 3
 PROP Rule type
  Inference
 RULE PL_IMAGE_L_PV
 PROP Rule text
if pl_p_image is 'Low'
then pl_pv_image = 2
 PROP Rule type
  Inference
 RULE PL_IMAGE_VL_PV
 PROP Rule text
if pl_p_image is 'Very Low'
then pl_pv_image = 1
 PROP Rule type
  Inference
 RULE PL_IMITATION_VH_PV
 PROP Rule text
if pl_p_imitation is 'Very Easy'
then pl_pv_imitation = 5
 PROP Rule type
  Inference
 RULE PL_IMITATION_H_PV
 PROP Rule text
if pl_p_imitation is 'Easy'
then pl_pv_imitation = 4
 PROP Rule type
  Inference
 RULE PL_IMITATION_MOD_PV
 PROP Rule text
if pl_p_imitation is 'Average'
then pl_pv_imitation = 3
 PROP Rule type
  Inference
```

```
 RULE PL_IMITATION_L_PV
 PROP Rule text
if pl_p_imitation is 'Hard'
then pl_pv_imitation = 2
 PROP Rule type
  Inference
 RULE PL_IMITATION_VL_PV
 PROP Rule text
if pl_p_imitation is 'Very Hard'
then pl_pv_imitation = 1
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_COMPD_EVAL
 PROP Rule text
if certainty of (pl_competitor_eval) < .0
then establish pl_msg_d_eval immediate and
pl_fac_int_flag = 4
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_CORPAM_EVAL
 PROP Rule text
if certainty of (pl_corporate_eval) <= .5 and
certainty of (pl_corporate_eval) > .2
then establish pl_msg_am_eval and
pl_fac_int_flag = 2
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_CORPRE_EVAL
 PROP Rule text
if certainty of (pl_corporate_eval) <= .2 and
certainty of (pl_corporate_eval) > .0
then establish pl_msg_re_eval and
pl_fac_int_flag = 3
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_CORPD_EVAL
 PROP Rule text
if certainty of (pl_corporate_eval) < .0
then establish pl_msg_d_eval and
pl_fac_int_flag = 4
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_PA_EVAL
 PROP Rule text
if certainty of (pl_product_eval) > .5
then establish pl_msg_a_eval and
pl_fac_int_flag = 1
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_CORPA_EVAL
 PROP Rule text
if certainty of (pl_corporate_eval) > .5
then establish pl_msg_a_eval and
pl_fac_int_flag = 1
 PROP Rule type
```

```
  Inference
 RULE MF_PL_FACTOR_COMPA_EVAL
 PROP Rule text
if certainty of (pl_competitor_eval) > .5
then establish pl_msg_a_eval immediate and
pl_fac_int_flag = 1
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_COMPAM_EVAL
 PROP Rule text
if certainty of (pl_competitor_eval) <= .5 and
certainty of (pl_competitor_eval) > .2
then establish pl_msg_am_eval immediate and
pl_fac_int_flag = 2
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_COMPRE_EVAL
 PROP Rule text
if certainty of (pl_competitor_eval) <= .2 and
certainty of (pl_competitor_eval) > .0
then establish pl_msg_re_eval immediate and
pl_fac_int_flag = 3
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_MA_EVAL
 PROP Rule text
if certainty of (pl_market_eval) > .5
then establish pl_msg_a_eval and
pl_fac_int_flag = 1
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_MAM_EVAL
 PROP Rule text
if certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .2
then establish pl_msg_am_eval and
pl_fac_int_flag = 2
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_MRE_EVAL
 PROP Rule text
if certainty of (pl_market_eval) <= .2 and
certainty of (pl_market_eval) > .0
then establish pl_msg_re_eval and
pl_fac_int_flag = 3
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_MD_EVAL
 PROP Rule text
if certainty of (pl_market_eval) < .0
then establish pl_msg_d_eval and
pl_fac_int_flag = 4
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_FAM_EVAL
```

```
 PROP Rule text
if certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .2
then establish pl_msg_am_eval and
pl_fac_int_flag = 2
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_FRE_EVAL
 PROP Rule text
if certainty of (pl_finance_eval) <= .2 and
certainty of (pl_finance_eval) > .0
then establish pl_msg_re_eval and
pl_fac_int_flag = 3
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_FD_EVAL
 PROP Rule text
if certainty of (pl_finance_eval) < .0
then establish pl_msg_d_eval and
pl_fac_int_flag = 4
 PROP Rule type
  Inference
 RULE PL_QUALITY_VH_COMPV
 PROP Rule text
if pl_comp_quality is 'Very High'
then pl_compv_quality = 5
 PROP Rule type
  Inference
 RULE PL_QUALITY_H_COMPV
 PROP Rule text
if pl_comp_quality is 'High'
then pl_compv_quality = 4
 PROP Rule type
  Inference
 RULE PL_QUALITY_MOD_COMPV
 PROP Rule text
if pl_comp_quality is 'Average'
then pl_compv_quality = 3
 PROP Rule type
  Inference
 RULE PL_QUALITY_L_COMPV
 PROP Rule text
if pl_comp_quality is 'Low'
then pl_compv_quality = 2
 PROP Rule type
  Inference
 RULE PL_QUALITY_VL_COMPV
 PROP Rule text
if pl_comp_quality is 'Very Low'
then pl_compv_quality = 1
 PROP Rule type
  Inference
 RULE PL_VULNER_VH_COMPV
 PROP Rule text
if pl_comp_vulnerability is 'Very High'
```

```
then pl_compv_vulnerability = 5
 PROP Rule type
  Inference
 RULE PL_VULNER_H_COMPV
 PROP Rule text
if pl_comp_vulnerability is 'High'
then pl_compv_vulnerability = 4
 PROP Rule type
  Inference
 RULE PL_VULNER_MOD_COMPV
 PROP Rule text
if pl_comp_vulnerability is 'Average'
then pl_compv_vulnerability = 3
 PROP Rule type
  Inference
 RULE PL_VULNER_L_COMPV
 PROP Rule text
if pl_comp_vulnerability is 'Low'
then pl_compv_vulnerability = 2
 PROP Rule type
  Inference
 RULE PL_VULNER_VL_COMPV
 PROP Rule text
if pl_comp_vulnerability is 'Very Low'
then pl_compv_vulnerability = 1
 PROP Rule type
  Inference
 RULE PL_COMP_VH_CORPV
 PROP Rule text
if pl_corp_comp is 'Very High'
then pl_corpv_comp = 5
 PROP Rule type
  Inference
 RULE PL_COMP_H_CORPV
 PROP Rule text
if pl_corp_comp is 'High'
then pl_corpv_comp = 4
 PROP Rule type
  Inference
 RULE PL_COMP_MOD_CORPV
 PROP Rule text
if pl_corp_comp is 'Average'
then pl_corpv_comp = 3
 PROP Rule type
  Inference
 RULE PL_COMP_L_CORPV
 PROP Rule text
if pl_corp_comp is 'Low'
then pl_corpv_comp = 2
 PROP Rule type
  Inference
 RULE PL_COMP_VL_CORPV
 PROP Rule text
if pl_corp_comp is 'Very Low'
then pl_corpv_comp = 1
```

```
 PROP Rule type
  Inference
 RULE PL_ORG_VH_CORPV
 PROP Rule text
if pl_corp_org is 'Very High'
then pl_corpv_org = 5
 PROP Rule type
  Inference
 RULE PL_ORG_H_CORPV
 PROP Rule text
if pl_corp_org is 'High'
then pl_corpv_org = 4
 PROP Rule type
  Inference
 RULE PL_ORG_MOD_CORPV
 PROP Rule text
if pl_corp_org is 'Average'
then pl_corpv_org = 3
 PROP Rule type
  Inference
 RULE PL_ORG_L_CORPV
 PROP Rule text
if pl_corp_org is 'Low'
then pl_corpv_org = 2
 PROP Rule type
  Inference
 RULE PL_ORG_VL_CORPV
 PROP Rule text
if pl_corp_org is 'Very Low'
then pl_corpv_org = 1
 PROP Rule type
  Inference
 RULE PL_PER_M_VH_CORPV
 PROP Rule text
if pl_corp_per_marketing is 'Very High'
then pl_corpv_per_marketing = 5
 PROP Rule type
  Inference
 RULE PL_PER_M_H_CORPV
 PROP Rule text
if pl_corp_per_marketing is 'High'
then pl_corpv_per_marketing = 4
 PROP Rule type
  Inference
 RULE PL_PER_M_MOD_CORPV
 PROP Rule text
if pl_corp_per_marketing is 'Average'
then pl_corpv_per_marketing = 3
 PROP Rule type
  Inference
 RULE PL_PER_M_L_CORPV
 PROP Rule text
if pl_corp_per_marketing is 'Low'
then pl_corpv_per_marketing = 2
 PROP Rule type
```

```
  Inference
 RULE PL_PER_M_VL_CORPV
 PROP Rule text
if pl_corp_per_marketing is 'Very Low'
then pl_corpv_per_marketing = 1
 PROP Rule type
  Inference
 RULE PL_PER_T_VH_CORPV
 PROP Rule text
if pl_corp_per_tech is 'Very High'
then pl_corpv_per_tech = 5
 PROP Rule type
  Inference
 RULE PL_PER_T_H_CORPV
 PROP Rule text
if pl_corp_per_tech is 'High'
then pl_corpv_per_tech = 4
 PROP Rule type
  Inference
 RULE PL_PER_T_MOD_CORPV
 PROP Rule text
if pl_corp_per_tech is 'Average'
then pl_corpv_per_tech = 3
 PROP Rule type
  Inference
 RULE PL_PER_T_L_CORPV
 PROP Rule text
if pl_corp_per_tech is 'Low'
then pl_corpv_per_tech = 2
 PROP Rule type
  Inference
 RULE PL_PER_T_VL_CORPV
 PROP Rule text
if pl_corp_per_tech is 'Very Low'
then pl_corpv_per_tech = 1
 PROP Rule type
  Inference
 RULE MF_PL_CATCH_EVAL
 PROP Rule text
if pl_final_flag = 0
then establish pl_fin_msg_re_eval
 PROP Rule type
  Inference
 RULE PL_EVAL_3
 PROP Rule text
fif certainty of (pl_finance_eval) > .5 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .0 or
certainty of (pl_finance_eval) > .5 and
```

```
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) > .5 and
certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .0 and
certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) > .5 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .0 and
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .0 and
certainty of (pl_competitor_eval) < .5 and

certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .0 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_corporate_eval) > .5 and

certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5
then establish pl_fin_msg_a_eval immediate and
pl_final_flag = 2
 PROP Rule type
  Inference
 RULE PL_EVAL_2B
 PROP Rule text
fif certainty of (pl_finance_eval) > .5 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .0 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .0 or
certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) > .5 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .0 and
```

```
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .0 or
certainty of (pl_finance_eval) > .5 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .0 and
certainty of (pl_market_eval) > .5 and
certainty of (pl_market_eval) < .0 and
certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .0 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) > .5 and
certainty of (pl_product_eval) < .5 and
certainty of (pl_product_eval) > .0 or
certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .0 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) < .5 and
certainty of (pl_market_eval) > .0 and
certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) > .5 and

certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .0 and
certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .0 and
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .0 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) < .5 or
certainty of (pl_product_eval) > .0 or
certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .0 and
certainty of (pl_corporate_eval) > .5 and
```

```
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) > .5 and
certainty of (pl_market_eval) < .0 and
certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .0 and
certainty of (pl_corporate_eval) > .5 and

certainty of (pl_competitor_eval) < .5 and
certainty of (pl_competitor_eval) > .0 and
certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5 or

certainty of (pl_finance_eval) < .5 and
certainty of (pl_finance_eval) > .0 and
certainty of (pl_corporate_eval) < .5 and
certainty of (pl_corporate_eval) > .0 and
certainty of (pl_competitor_eval) > .5 and

certainty of (pl_market_eval) > .5 and

certainty of (pl_product_eval) > .5

then establish pl_fin_msg_a_eval immediate and
pl_final_flag = 2
 PROP Rule type
  Inference
 RULE PL_P_APPEAL_A_EVAL
 PROP Rule text
if pl_pv_appeal >= rpl_pv_appeal
then there is some evidence that pl_product_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_P_APPEAL_D_EVAL
 PROP Rule text
if pl_pv_appeal < rpl_pv_appeal
then there is some negative evidence that
pl_product_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_P_DIST_A_EVAL
 PROP Rule text
if pl_pv_dist >= rpl_pv_dist
then there is some evidence that pl_product_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_P_DIST_D_EVAL
 PROP Rule text
if pl_pv_dist < rpl_pv_dist
then there is some negative evidence that
```

```
pl_product_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_P_IMAGE_A_EVAL
 PROP Rule text
if pl_pv_image >= rpl_pv_image
then there is some evidence that pl_product_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_P_IMAGE_D_EVAL
 PROP Rule text
if pl_pv_image < rpl_pv_image
then there is some negative evidence that
pl_product_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_P_IMITATION_A_EVAL
 PROP Rule text
if pl_pv_imitation >= rpl_pv_imitation
then there is some evidence that pl_product_eval is
'approve'
 PROP Rule type
  Inference
 RULE PL_P_IMITATION_D_EVAL
 PROP Rule text
if pl_pv_imitation < rpl_pv_imitation
then there is some negative evidence that
pl_product_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_COMP_A_EVAL
 PROP Rule text
if pl_corpv_comp >= rpl_corpv_comp
then there is some evidence that pl_corporate_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_COMP_D_EVAL
 PROP Rule text
if pl_corpv_comp < rpl_corpv_comp
then there is some negative evidence that
pl_corporate_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_ORG_A_EVAL
 PROP Rule text
if pl_corpv_org >= rpl_corpv_org
then there is some evidence that pl_corporate_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_ORG_D_EVAL
 PROP Rule text
if pl_corpv_org < rpl_corpv_org
```

```
then there is some negative evidence that
pl_corporate_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_PER_M_A_EVAL
 PROP Rule text
if pl_corpv_per_marketing < rpl_corpv_per_marketing
then there is some evidence that pl_corporate_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_PER_M_D_EVAL
 PROP Rule text
if pl_corpv_per_marketing >= rpl_corpv_per_marketing
then there is some negative evidence that
pl_corporate_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_PER_T_A_EVAL
 PROP Rule text
if pl_corpv_per_tech < rpl_corpv_per_tech
then there is some evidence that pl_corporate_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_PER_T_D_EVAL
 PROP Rule text
if pl_corpv_per_tech >= rpl_corpv_per_tech
then there is some negative evidence that
pl_corporate_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_OUTLAY_A_EVAL
 PROP Rule text
if pl_corp_outlay <= rpl_corp_outlay
then there is some evidence that pl_corporate_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_CORP_OUTLAY_D_EVAL
 PROP Rule text
if pl_corp_outlay > rpl_corp_outlay
then there is some negative evidence that
pl_corporate_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_COMP_QUALITY_A_EVAL
 PROP Rule text
if pl_compv_quality <= rpl_compv_quality
then there is some evidence that pl_competitor_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_COMP_QUALITY_D_EVAL
 PROP Rule text
```

```
if pl_compv_quality > rpl_compv_quality
then there is some negative evidence that
pl_competitor_eval is 'approve'
 PROP Rule type
  Inference
 RULE PL_COMP_VULNER_A_EVAL
 PROP Rule text
if pl_compv_vulnerability >= rpl_compv_vulnerability
then there is some evidence that pl_competitor_eval
is 'approve'
 PROP Rule type
  Inference
 RULE PL_COMP_VULNER_D_EVAL
 PROP Rule text
if pl_compv_vulnerability < rpl_compv_vulnerability
then there is some negative evidence that
pl_competitor_eval is 'approve'
 PROP Rule type
  Inference
 RULE NAME_PL_STAT
 PROP Rule text
if ask_type = 'Product Line'
then namehold = rpl_name
 PROP Rule type
  Inference
 RULE NAME_B_STAT
 PROP Rule text
if ask_type = 'Brand'
then namehold = db_name
 PROP Rule type
  Inference
 RULE NAME_P_STAT
 PROP Rule text
if ask_type = 'Product'
then namehold = dp_name
 PROP Rule type
  Inference
 RULE MF_PL_FACTOR_FA_EVAL
 PROP Rule text
if certainty of (pl_finance_eval) > .5
then establish pl_msg_a_eval and
pl_fac_int_flag = 1
 PROP Rule type
  Inference
 RULE RB_TURNOVER_H_STV
 PROP Rule text
if rb_st_turnover is 'Important'
then rb_stv_turnover = 4
 PROP Rule type
  Inference
 RULE B_RELATIONSHIP_E_QUALV
 PROP Rule text
if b_qual_relationship is 'excellent'
then b_qualv_relationship = 5
 PROP Rule type
```

```
  Inference
 RULE B_RELATIONSHIP_G_QUALV
 PROP Rule text
if b_qual_relationship is 'good'
then b_qualv_relationship = 4
 PROP Rule type
  Inference
 RULE B_RELATIONSHIP_F_QUALV
 PROP Rule text
if b_qual_relationship is 'fair'
then b_qualv_relationship = 3
 PROP Rule type
  Inference
 RULE B_RELATIONSHIP_P_QUALV
 PROP Rule text
if b_qual_relationship is 'poor'
then b_qualv_relationship = 2
 PROP Rule type
  Inference
 RULE B_RELATIONSHIP_VP_QUALV
 PROP Rule text
if b_qual_relationship is 'very poor'
then b_qualv_relationship = 1
 PROP Rule type
  Inference
 RULE RB_RELATIONSHIP_E_QUALV
 PROP Rule text
if rb_qual_relationship is 'excellent'
then rb_qualv_relationship = 5
 PROP Rule type
  Inference
 RULE RB_RELATIONSHIP_G_QUALV
 PROP Rule text
if rb_qual_relationship is 'good'
then rb_qualv_relationship = 4
 PROP Rule type
  Inference
 RULE RB_RELATIONSHIP_F_QUALV
 PROP Rule text
if rb_qual_relationship is 'fair'
then rb_qualv_relationship = 3
 PROP Rule type
  Inference
 RULE RB_RELATIONSHIP_P_QUALV
 PROP Rule text
if rb_qual_relationship is 'poor'
then rb_qualv_relationship = 2
 PROP Rule type
  Inference
 RULE RB_RELATIONSHIP_VP_QUALV
 PROP Rule text
if rb_qual_relationship is 'very poor'
then rb_qualv_relationship = 1
 PROP Rule type
  Inference
```

```
 RULE B_REPUTATION_E_STV
 PROP Rule text
if b_st_reputation is 'excellent'
then b_stv_reputation = 5
 PROP Rule type
  Inference
 RULE B_REPUTATION_G_STV
 PROP Rule text
if b_st_reputation is 'good'
then b_stv_reputation = 4
 PROP Rule type
  Inference
 RULE B_REPUTATION_F_STV
 PROP Rule text
if b_st_reputation is 'fair'
then b_stv_reputation = 3
 PROP Rule type
  Inference
 RULE B_REPUTATION_P_STV
 PROP Rule text
if b_st_reputation is 'poor'
then b_stv_reputation = 2
 PROP Rule type
  Inference
 RULE B_REPUTATION_VP_STV
 PROP Rule text
if b_st_reputation is 'very poor'
then b_stv_reputation = 1
 PROP Rule type
  Inference
 RULE MF_B_VH_FIND
 PROP Rule text
if 1 of the following (mf_b_earn_value = 'Very Importan
t', mf_b_past_value = 'Very Important', mf_b_qual_value
= 'Very Important', mf_b_size_value = 'Very Important',
 mf_b_st_value = 'Very Important')
then establish mf_b_vh_figure immediate
 PROP Rule type
  Inference
 RULE W_MF_PL
 PROP Rule text
if ask_type = 'product line'
then establish product_line
 PROP Rule type
  Inference
 RULE W_MF_B
 PROP Rule text
if ask_type = 'brand'
then establish brand_name
 PROP Rule type
  Inference
 RULE W_MF_P
 PROP Rule text
if ask_type = 'product'
then establish product
```

```
 PROP Rule type
  Inference
 RULE MF_B_EARN_H
 PROP Rule text
if mf_b_earn_value = 'Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_EARN_VH
 PROP Rule text
if mf_b_earn_value = 'Very Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_EARN_MOD
 PROP Rule text
if mf_b_earn_value = 'Average'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_EARN_L
 PROP Rule text
if mf_b_earn_value = 'Not Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_EARN_VL
 PROP Rule text
if mf_b_earn_value = 'Not At All Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_PAST_VH
 PROP Rule text
if mf_b_past_value = 'Very Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_PAST_H
 PROP Rule text
if mf_b_past_value = 'Important'
```

```
then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_PAST_MOD
 PROP Rule text
if mf_b_past_value = 'Average'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_PAST_L
 PROP Rule text
if mf_b_past_value = 'Not Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_PAST_VL
 PROP Rule text
if mf_b_past_value = 'Not At All Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_QUAL_VH
 PROP Rule text
if mf_b_qual_value = 'Very Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_QUAL_H
 PROP Rule text
if mf_b_past_value = 'Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_QUAL_MOD
 PROP Rule text
if mf_b_past_value = 'Average'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_QUAL_L
 PROP Rule text
if mf_b_past_value = 'Not Important'
```

```
then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_QUAL_VL
 PROP Rule text
if mf_b_qual_value = 'Very Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_SIZE_VH
 PROP Rule text
if mf_b_size_value = 'Very Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_SIZE_H
 PROP Rule text
if mf_b_size_value = 'Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_SIZE_MOD
 PROP Rule text
if mf_b_size_value = 'Average'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_SIZE_L
 PROP Rule text
if mf_b_size_value = 'Not Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_SIZE_VL
 PROP Rule text
if mf_b_size_value = 'Very Important'

then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_ST_VH
 PROP Rule text
if mf_b_st_value = 'Very Important'
```

```
then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_ST_H
 PROP Rule text
if mf_b_st_value = 'Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_ST_MOD
 PROP Rule text
if mf_b_earn_value = 'Average'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_ST_L
 PROP Rule text
if mf_b_st_value = 'Not At All Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 RULE MF_B_ST_VL
 PROP Rule text
if mf_b_st_value = 'Not At All Important'


then establish plf_market_eval
 PROP Rule type
  Inference
 SCREEN PROFDISP
 PROP Screen Defs.
FIELD
UPPERLEFTROW 1
UPPERLEFTCOL 4
LOWERRIGHTROW 1
LOWERRIGHTCOL 74
STARTLITERAL
You are evaluating a :VL *ask_type against Profile :VL *namehold
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
```

```
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 3
UPPERLEFTCOL 13
LOWERRIGHTROW 6
LOWERRIGHTCOL 67
STARTLITERAL
 III   N   N   N   N  0000  V   V   AA   TTT  0000  RRRR
  I    NN  N   NN  N  0  0   V V   A  A   T   0  0  R  R
  I    N N N   N N N  0  0   V V   AAAA   T   0  0  RRR
 III   N   N   N   N  0000    V    A  A   T   0000  R  R
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT NO
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 8
UPPERLEFTCOL 5
LOWERRIGHTROW 19
LOWERRIGHTCOL 73
FIELDTYPE DISPLAY
INITFIELDTYPE DISPLAY
BORDER YES
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
```

```
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID YES
COLOR7 PINK
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 21
UPPERLEFTCOL 20
LOWERRIGHTROW 21
LOWERRIGHTCOL 54
STARTLITERAL
(Press ENTER if ready to continue)
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 22
UPPERLEFTCOL 5
LOWERRIGHTROW 22
LOWERRIGHTCOL 76
STARTLITERAL
PF1 Help  PF2 Review  PF4 What  PF7 Up  PF8 Down  PF10 How  PF11 Why
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
```

```
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
 SCREEN TITLE
 PROP Screen Defs.
FIELD
UPPERLEFTROW 3
UPPERLEFTCOL 13
LOWERRIGHTROW 6
LOWERRIGHTCOL 67
STARTLITERAL
 III   N   N   N   N  0000  V   V   AA   TTT  0000  RRRR
  I    NN  N   NN  N  0  0   V V   A  A   T   0  0  R  R
  I    N NN   N NN  0  0   V V   AAAA   T   0  0  RRR
 III   N   N   N   N  0000    V    A  A   T   0000  R  R
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT NO
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 8
UPPERLEFTCOL 5
LOWERRIGHTROW 19
LOWERRIGHTCOL 73
FIELDTYPE DISPLAY
INITFIELDTYPE DISPLAY
BORDER YES
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
```

```
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID YES
COLOR7 PINK
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 21
UPPERLEFTCOL 19
LOWERRIGHTROW 21
LOWERRIGHTCOL 55
STARTLITERAL
[ Press ENTER if ready to continue ]
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 22
UPPERLEFTCOL 5
LOWERRIGHTROW 22
LOWERRIGHTCOL 76
STARTLITERAL
PF1 Help  PF2 Review  PF4 What  PF7 Up  PF8 Down  PF10 How  PF11 Why
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
```

```
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
 SCREEN CHOSE
 PROP Screen Defs.
FIELD
UPPERLEFTROW 1
UPPERLEFTCOL 4
LOWERRIGHTROW 1
LOWERRIGHTCOL 74
STARTLITERAL
You are evaluating a :VL *ask_type against Profile :VL *namehold
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 3
UPPERLEFTCOL 13
LOWERRIGHTROW 6
LOWERRIGHTCOL 67
STARTLITERAL
 III  N  N  N  N  0000  V   V   AA   TTT  0000  RRRR
  I   NN N  NN N  0  0   V V   A  A   T   0  0  R  R
  I   N NN  N NN  0  0   V V   AAAA   T   0  0  RRR
 III  N  N  N  N  0000    V    A  A   T   0000  R  R
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
```

```
FORMAT NO
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 8
UPPERLEFTCOL 5
LOWERRIGHTROW 10
LOWERRIGHTCOL 73
FIELDTYPE QUESTION
FIELDTYPE HOW
FIELDTYPE WHAT
FIELDTYPE WHY
INITFIELDTYPE QUESTION
BORDER YES
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT YES
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID YES
COLOR7 WHITE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 12
UPPERLEFTCOL 5
LOWERRIGHTROW 20
LOWERRIGHTCOL 37
FIELDTYPE ANSWER
INITFIELDTYPE ANSWER
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 2
FORMAT YES
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
```

```
REQUIRED YES
DISPLAYONLY NO
LISTVALID YES
COLOR7 TURQUOISE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 YELLOW
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 22
UPPERLEFTCOL 5
LOWERRIGHTROW 22
LOWERRIGHTCOL 76
STARTLITERAL
PF1 Help  PF2 Review  PF4 What  PF7 Up  PF8 Down  PF10 How  PF11 Why
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
 SCREEN STRENT
 PROP Screen Defs.
FIELD
UPPERLEFTROW 1
UPPERLEFTCOL 4
LOWERRIGHTROW 1
LOWERRIGHTCOL 74
STARTLITERAL
You are evaluating a :VL *ask_type against Profile :VL *namehold
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
```

```
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 3
UPPERLEFTCOL 13
LOWERRIGHTROW 6
LOWERRIGHTCOL 67
STARTLITERAL
 III   N   N   N   N  0000  V   V   AA   TTT  0000  RRRR
  I    NN  N   NN  N  0  0   V V   A  A   T   0  0  R  R
  I    N N N   N N N  0  0   V V   AAAA   T   0  0  RRR
 III   N   N   N   N  0000    V    A  A   T   0000  R  R
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT NO
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 8
UPPERLEFTCOL 5
LOWERRIGHTROW 10
LOWERRIGHTCOL 73
FIELDTYPE QUESTION
FIELDTYPE HOW
FIELDTYPE WHAT
FIELDTYPE WHY
INITFIELDTYPE QUESTION
BORDER YES
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT YES
```

```
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID YES
COLOR7 WHITE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 13
UPPERLEFTCOL 5
LOWERRIGHTROW 15
LOWERRIGHTCOL 73
FIELDTYPE ANSWER
INITFIELDTYPE ANSWER
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT YES
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED YES
DISPLAYONLY NO
LISTVALID YES
COLOR7 TURQUOISE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 YELLOW
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 22
UPPERLEFTCOL 5
LOWERRIGHTROW 22
LOWERRIGHTCOL 76
STARTLITERAL
PF1 Help  PF2 Review  PF4 What  PF7 Up  PF8 Down  PF10 How  PF11 Why
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
```

```
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
 SCREEN CHOSEI
 PROP Screen Defs.
FIELD
UPPERLEFTROW 3
UPPERLEFTCOL 13
LOWERRIGHTROW 6
LOWERRIGHTCOL 67
STARTLITERAL
 III   N  N   N  N   0000   V    V    AA    TTT   0000   RRR
  I    NN N   NN N   0  0    V V    A  A     T    0  0   R  R
  I    N NN   N NN   0  0    V V    AAAA     T    0  0   RRR
 III   N  N   N  N   0000     V     A  A     T    0000   R  R
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT NO
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 8
UPPERLEFTCOL 5
LOWERRIGHTROW 12
LOWERRIGHTCOL 73
FIELDTYPE QUESTION
FIELDTYPE HOW
FIELDTYPE WHAT
FIELDTYPE WHY
INITFIELDTYPE QUESTION
BORDER YES
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT YES
```

```
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID YES
COLOR7 WHITE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 14
UPPERLEFTCOL 5
LOWERRIGHTROW 20
LOWERRIGHTCOL 37
FIELDTYPE ANSWER
INITFIELDTYPE ANSWER
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 2
FORMAT YES
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED YES
DISPLAYONLY NO
LISTVALID YES
COLOR7 TURQUOISE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 YELLOW
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 22
UPPERLEFTCOL 5
LOWERRIGHTROW 22
LOWERRIGHTCOL 76
STARTLITERAL
PF1 Help  PF2 Review  PF4 What  PF7 Up  PF8 Down  PF10 How  PF11 Why
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
```

```
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
 SCREEN STRENTI
 PROP Screen Defs.
FIELD
UPPERLEFTROW 3
UPPERLEFTCOL 13
LOWERRIGHTROW 6
LOWERRIGHTCOL 67
STARTLITERAL
 III  N  N  N  N  0000  V   V   AA   TTT  0000  RRRR
  I   NN N  NN N  O  O   V V   A  A   T   O  O  R  R
  I   N NN  N NN  O  O   V V   AAAA   T   O  O  RRR
 III  N  N  N  N  0000    V    A  A   T   0000  R  R
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW O
ARROWCOL O
CERTAINTYWIDTH O
FORMAT NO
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY BRIGHT
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 8
UPPERLEFTCOL 5
LOWERRIGHTROW 10
LOWERRIGHTCOL 73
FIELDTYPE QUESTION
FIELDTYPE HOW
FIELDTYPE WHAT
FIELDTYPE WHY
INITFIELDTYPE QUESTION
BORDER YES
ARROWROW O
ARROWCOL O
CERTAINTYWIDTH O
FORMAT YES
```

```
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
DISPLAYONLY YES
LISTVALID YES
COLOR7 WHITE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 13
UPPERLEFTCOL 5
LOWERRIGHTROW 15
LOWERRIGHTCOL 73
FIELDTYPE ANSWER
INITFIELDTYPE ANSWER
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT YES
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED YES
DISPLAYONLY NO
LISTVALID YES
COLOR7 TURQUOISE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 YELLOW
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
FIELD
UPPERLEFTROW 22
UPPERLEFTCOL 5
LOWERRIGHTROW 22
LOWERRIGHTCOL 76
STARTLITERAL
PF1 Help  PF2 Review  PF4 What  PF7 Up  PF8 Down  PF10 How  PF11 Why
ENDLITERAL
FIELDTYPE LITERAL
INITFIELDTYPE LITERAL
BORDER NO
ARROWROW 0
ARROWCOL 0
CERTAINTYWIDTH 0
FORMAT DEFAULT
HEADER DEFAULT
VALUES DEFAULT
KNOWN DEFAULT
REQUIRED NO
```

```
DISPLAYONLY YES
LISTVALID NO
COLOR7 BLUE
INTENSITY NORMAL
HIGHLIGHT NO HIGHLIGHT
CERTCOLOR7 GREEN
CERTINTENSITY NORMAL
CERTHIGHLIGHT NO HIGHLIGHT
 GROUP LOG_BUILD
 PROP Member list
PARAMETER:FUSERCO
PARAMETER:FUSERNAME
PARAMETER:USERTITLE
PARAMETER:ASK_TYPE
PARAMETER:NAMEHOLD
 GROUP INTRO
 PROP Member list
PARAMETER:USERNAME
PARAMETER:USERCO
PARAMETER:USERTITLE
 GROUP DPL_QUEST
 PROP Member list
PARAMETER:DPL_CORP_OUTLAY
PARAMETER:DPL_F_GMARGIN
PARAMETER:DPL_F_PAYBACK
PARAMETER:DPL_F_RETURN
PARAMETER:DPL_F_VOL
PARAMETER:DPL_M_GRW_RATE
PARAMETER:DPL_M_SIZE
 GROUP RPLG
 PROP Member list
PARAMETER:RPL_CORP_OUTLAY
PARAMETER:RPL_M_SIZE
PARAMETER:RPL_F_GMARGIN
PARAMETER:RPL_F_PAYBACK
PARAMETER:RPL_F_RETURN
PARAMETER:RPL_F_VOL
PARAMETER:RPL_M_GRW_RATE
 GROUP PROF_GET
 PROP Member list
RULE:GET_NEW_PROFILE
RULE:GET_OLD_PROFILE
 GROUP PICK_DB_ROW
 PROP Member list
RULE:NEW_ROW
RULE:OLD_ROW
 GROUP DPLG
 PROP Member list
PARAMETER:DPL_CORP_OUTLAY
PARAMETER:DPL_F_GMARGIN
PARAMETER:DPL_F_PAYBACK
PARAMETER:DPL_F_RETURN
PARAMETER:DPL_F_VOL
PARAMETER:DPL_M_GRW_RATE
PARAMETER:DPL_M_SIZE
```

```
 GROUP PLG
 PROP Member list
PARAMETER:PL_COMP_QUALITY
PARAMETER:PL_COMP_VULNERABILITY
PARAMETER:PL_CORP_COMP
PARAMETER:PL_CORP_ORG
PARAMETER:PL_CORP_OUTLAY
PARAMETER:PL_CORP_PER_MARKETING
PARAMETER:PL_CORP_PER_TECH
PARAMETER:PL_F_GMARGIN
PARAMETER:PL_F_PAYBACK
PARAMETER:PL_F_RETURN
PARAMETER:PL_F_VOL
PARAMETER:PL_M_GRW_RATE
PARAMETER:PL_M_SEASON
PARAMETER:PL_M_SIZE
PARAMETER:PL_M_S_ECON
PARAMETER:PL_P_APPEAL
PARAMETER:PL_P_DIST
PARAMETER:PL_P_IMAGE
PARAMETER:PL_P_IMITATION
 GROUP PL_VALUES
 PROP Member list
PARAMETER:PL_M_VALUE
PARAMETER:PL_F_VALUE
PARAMETER:PL_P_VALUE
PARAMETER:PL_COMP_VALUE
PARAMETER:PL_CORP_VALUE
 GROUP PL_FIGURE
 PROP Member list
RULE:PL_VH_FIND
RULE:PL_H_FIND
RULE:PL_MOD_FIND
RULE:PL_L_FIND
RULE:PL_VL_FIND
 GROUP PL_MARKET_Q
 PROP Member list
PARAMETER:PL_M_SIZE
PARAMETER:PL_M_GRW_RATE
PARAMETER:PL_M_S_ECON
PARAMETER:PL_M_SEASON
 GROUP PL_FINANCE_Q
 PROP Member list
PARAMETER:PL_F_GMARGIN
PARAMETER:PL_F_PAYBACK
PARAMETER:PL_F_RETURN
PARAMETER:PL_F_VOL
 GROUP PL_PRODUCT_Q
 PROP Member list
PARAMETER:PL_P_APPEAL
PARAMETER:PL_P_DIST
PARAMETER:PL_P_IMAGE
PARAMETER:PL_P_IMITATION
 GROUP PL_COMPETITORS_Q
 PROP Member list
```

```
PARAMETER:PL_COMP_QUALITY
PARAMETER:PL_COMP_VULNERABILITY
 GROUP PL_CORPORATE_Q
 PROP Member list
PARAMETER:PL_CORP_COMP
PARAMETER:PL_CORP_ORG
PARAMETER:PL_CORP_OUTLAY
PARAMETER:PL_CORP_PER_MARKETING
PARAMETER:PL_CORP_PER_TECH
 GROUP PL_VH_PICK
 PROP Member list
RULE:PL_M_VH
RULE:PL_F_VH
RULE:PL_P_VH
RULE:PL_COMP_VH
RULE:PL_CORP_VH
 GROUP PL_H_PICK
 PROP Member list
RULE:PL_M_H
RULE:PL_F_H
RULE:PL_P_H
RULE:PL_COMP_H
RULE:PL_CORP_H
 GROUP PL_MOD_PICK
 PROP Member list
RULE:PL_M_MOD
RULE:PL_F_MOD
RULE:PL_P_MOD
RULE:PL_COMP_MOD
RULE:PL_CORP_MOD
 GROUP PL_L_PICK
 PROP Member list
RULE:PL_M_L
RULE:PL_F_L
RULE:PL_P_L
RULE:PL_COMP_L
RULE:PL_CORP_L
 GROUP PL_VL_PICK
 PROP Member list
RULE:PL_M_VL
RULE:PL_F_VL
RULE:PL_P_VL
RULE:PL_COMP_VL
RULE:PL_CORP_VL
 GROUP COMP_FAC_EVAL
 PROP Member list
RULE:MF_PL_FACTOR_COMPA_EVAL
RULE:MF_PL_FACTOR_COMPAM_EVAL
RULE:MF_PL_FACTOR_COMPRE_EVAL
RULE:MF_PL_FACTOR_COMPD_EVAL
 GROUP CORP_FAC_EVAL
 PROP Member list
RULE:MF_PL_FACTOR_CORPA_EVAL
RULE:MF_PL_FACTOR_CORPAM_EVAL
RULE:MF_PL_FACTOR_CORPRE_EVAL
```

```
RULE:MF_PL_FACTOR_CORPD_EVAL
 GROUP PROD_FAC_EVAL
 PROP Member list
RULE:MF_PL_FACTOR_PA_EVAL
RULE:MF_PL_FACTOR_PAM_EVAL
RULE:MF_PL_FACTOR_PRE_EVAL
RULE:MF_PL_FACTOR_PD_EVAL
 GROUP MAR_FAC_EVAL
 PROP Member list
RULE:MF_PL_FACTOR_MA_EVAL
RULE:MF_PL_FACTOR_MAM_EVAL
RULE:MF_PL_FACTOR_MRE_EVAL
RULE:MF_PL_FACTOR_MD_EVAL
 GROUP FIN_FAC_EVAL
 PROP Member list
RULE:MF_PL_FACTOR_FA_EVAL
RULE:MF_PL_FACTOR_FAM_EVAL
RULE:MF_PL_FACTOR_FRE_EVAL
RULE:MF_PL_FACTOR_FD_EVAL
 GROUP MF_PL_FIN_RULES_EVAL
 PROP Member list
RULE:PL_EVAL_1
RULE:PL_EVAL_2
RULE:PL_EVAL_2A
RULE:PL_EVAL_4
RULE:PL_EVAL_6
RULE:PL_EVAL_3
RULE:PL_EVAL_2B
 GROUP NAMESTAT
 PROP Member list
RULE:NAME_PL_STAT
RULE:NAME_P_STAT
RULE:NAME_B_STAT
 GROUP DB_BUILD
 PROP Member list
PARAMETER:USERCO
PARAMETER:USERNAME
PARAMETER:DPL_NAME
 GROUP MF_B_VH_PICK
 PROP Member list
RULE:MF_B_EARN_VH
RULE:MF_B_PAST_VH
RULE:MF_B_QUAL_VH
RULE:MF_B_SIZE_VH
RULE:MF_B_ST_VH
 GROUP MF_WHICH
 PROP Member list
RULE:W_MF_PL
RULE:W_MF_B
RULE:W_MF_P
 GROUP MF_BRAND_VALUES
 PROP Member list
PARAMETER:MF_B_EARN_VALUE
PARAMETER:MF_B_PAST_VALUE
PARAMETER:MF_B_QUAL_VALUE
```

```
PARAMETER:MF_B_SIZE_VALUE
PARAMETER:MF_B_ST_VALUE
 GROUP MF_B_H_PICK
 PROP Member list
RULE:MF_B_EARN_H
RULE:MF_B_PAST_H
RULE:MF_B_QUAL_H
RULE:MF_B_SIZE_H
RULE:MF_B_ST_H
 GROUP MF_B_MOD_PICK
 PROP Member list
RULE:MF_B_EARN_MOD
RULE:MF_B_PAST_MOD
RULE:MF_B_QUAL_MOD
RULE:MF_B_SIZE_MOD
RULE:MF_B_ST_MOD
 GROUP MF_B_L_PICK
 PROP Member list
RULE:MF_B_EARN_L
RULE:MF_B_PAST_L
RULE:MF_B_QUAL_L
RULE:MF_B_SIZE_L
RULE:MF_B_ST_L
 GROUP MF_B_VL_PICK
 PROP Member list
RULE:MF_B_EARN_VL
RULE:MF_B_PAST_VL
RULE:MF_B_QUAL_VL
RULE:MF_B_SIZE_VL
RULE:MF_B_ST_VL
 GROUP MF_B_EARN_Q
 PROP Member list
PARAMETER:B_EARN_COMMISSION
PARAMETER:B_EARN_MIN_AMOUNT
 GROUP MF_B_PAST_Q
 PROP Member list
PARAMETER:B_PAST_GROWTH
PARAMETER:B_PAST_RETURN
PARAMETER:B_PAST_TOP20
 GROUP MF_B_QUAL_Q
 PROP Member list
PARAMETER:B_QUAL_AVAILABILITY
PARAMETER:B_QUAL_RELATIONSHIP
PARAMETER:B_QUAL_SWITCH
 GROUP MF_B_SIZE_Q
 PROP Member list
PARAMETER:B_SIZE_ASSETS
PARAMETER:B_SIZE_NUMBER
PARAMETER:B_SIZE_RANGE
 GROUP MF_B_ST_Q
 PROP Member list
PARAMETER:B_ST_AGE
PARAMETER:B_ST_REPUTATION
PARAMETER:B_ST_TURNOVER
 FCB OVERMODULE
```

```
 PROP Parent
ROOT
 PROP Control text
ask intro;

establish which;
 PROP Announce
:CE ON
WELCOME TO INNOVATOR !


The Financial Services Expert System.
:CE OFF




 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Mult Choice Scr
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB WHICH
 PROP Parent
OVERMODULE
 PROP Control text
ask ask_type;
ask ask_pl;
discover
-- use mf_which;
determine fresult;
display (log_build,datetime);
access(insert(fuserco,fusername,usertitle,ask_type,
namehold));
 PROP Announce
INNOVATOR will assist you in evaluating any of the foll
owing types of Financial Services:
:CE ON
_
    1) A Product Line (e.g. Stocks, Mutual Funds, etc.)
    2) A Product      (e.g. Overseas Funds, etc.)
```

```
    3) A Brand  (e.g. Fidelity, Pioneer, Destiny, etc.)

:CE OFF
INNOVATOR will conduct the evaluation by asking you que
stions concerning the Financial Service of your choice.
 You can ask for clarifications by using the appropriat
e function keys.
 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Mult Choice Scr
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PRODUCT_LINE
 PROP Parent
WHICH
 PROP Control text

access (select profl_lst);
determine profile_old;
determine rpl_name;
discover
--use (get_new_profile, get_old_profile);
determine namehold;
establish mf_pl_factor_text;
ask pl_values;
discover
-- use pl_vh_find;
discover
-- use pl_h_find;
discover
-- use pl_mod_find;
discover
-- use pl_l_find;
discover
-- use pl_vl_find;
determine pl_final_eval;
discover
-- use mf_pl_fin_rules_eval;
determine pl_final_flag;

discover -- use mf_pl_catch_eval;
 PROP Max instances
1
 PROP Display Screen
```

```
SCREEN:TITLE
 PROP Mult Choice Scr
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB NEW_PROFILE
 PROP Parent
PRODUCT_LINE
 PROP Control text
ask dpl_name;
discover
-- use namestat;
ask dpl_quest;


access (insert (db_build, dpl_corp_outlay_n,
dpl_corp_outlay));
access (insert(db_build, dpl_f_gmargin_n, dpl_f_gmargin
));
access (insert(db_build, dpl_f_payback_n, dpl_f_payback
));
access(insert(db_build, dpl_f_return_n, dpl_f_return));
access(insert(db_build, dpl_f_vol_n, dpl_f_vol));
access(insert(db_build, dpl_m_grw_rate_n,
dpl_m_grw_rate));


establish use_old;
 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Mult Choice Scr
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB USE_OLD
 PROP Parent
PRODUCT_LINE
```

```
 PROP Control text

discover
--use pick_db_row;
access(select(rplg));


 PROP Max instances
1
 PROP Mult Choice Scr
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_VH_FIGURE
 PROP Parent
PRODUCT_LINE
 PROP Control text
discover
-- use pl_vh_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PLF_MARKET_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
ask pl_market_q;
determine pl_market_eval;
 PROP Announce
Now Evaluating the influence of MARKET  considerations
on this decision.
 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
```

```
ALL
 PROP Rule selection
ALL
 FCB PLF_PRODUCT_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
ask pl_product_q;
determine pl_product_eval;
 PROP Announce
Now Evaluating the influence of PRODUCT considerations
on this decision.
 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PLF_FINANCE_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
ask pl_finance_q;
determine pl_finance_eval;
 PROP Announce
Now Evaluating the influence of Financial consideration
s on this decision.
 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PLF_CORPORATE_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
ask pl_corporate_q;
determine pl_corporate_eval;
 PROP Announce
Now evaluating the influence of CORPORATE constraints f
rom your own organization on this decision.
 PROP Max instances
```

```
1
 PROP Display Screen
SCREEN:TITLE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PLF_COMPETITORS_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
ask pl_competitors_q;
determine pl_competitor_eval;
 PROP Announce
Now Evaluating the influence of COMPETITORS on this dec
ision.
 PROP Max instances
1
 PROP Display Screen
SCREEN:TITLE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_H_FIGURE
 PROP Parent
PRODUCT_LINE
 PROP Control text
discover
-- use pl_h_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_MOD_FIGURE
 PROP Parent
PRODUCT_LINE
 PROP Control text
discover
-- use pl_mod_pick;
 PROP Max instances
```

```
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_VL_FIGURE
 PROP Parent
PRODUCT_LINE
 PROP Control text
discover
-- use pl_vl_pick
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_L_FIGURE
 PROP Parent
PRODUCT_LINE
 PROP Control text
discover
-- use pl_l_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB ASK_OLD
 PROP Parent
PRODUCT_LINE
 PROP Control text
access (select prof1_lst);

determine ex_profile_pl;
determine rpl_name;


establish use_old;
 PROP Max instances
1
 PROP Mult Choice Scr
```

```
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_MSG_A_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The data for this factor has been APPROVED by INNOVATOR
.  This will contribute to the new product being apporv
ed.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_MSG_D_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The data for this factor has been DISAPPROVED by INNOVA
TOR.  This will contribute to the new product being dis
approved.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_MSG_RE_EVAL
 PROP Parent
```

```
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The data for this factor is too ambiguous for INNOVATOR
 to approve or disapprove.  Please re-evaluate the info
rmation this factor consists of and resubmit it to INNO
VATOR at your earliest opportunity.  This factor will n
ot contribute to the approval of the new product.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_MSG_AM_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The data for this factor has been MARGINALLY APPROVED b
y INNOVATOR.  This could contribute to the new product
being approved.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_FIN_MSG_A_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The new product has been APPROVED by INNOVATOR.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
```

```
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_FIN_MSG_AM_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The new product has been MARGINALLY APPROVED by INNOVAT
OR.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_FIN_MSG_RE_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The description of this new product is too ambiguous fo
r INNOVATOR to approve or disaprove.  Please re-evaluat
e the information and resubmit to INNOVATOR at your ear
liest opportunity.
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PL_FIN_MSG_D_EVAL
 PROP Parent
PRODUCT_LINE
 PROP Control text
determine pl_int_flag;
 PROP Announce
The new product has been DISAPPROVED by INNOVATOR.
```

```
 PROP Max instances
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_PL_FACTOR_TEXT
 PROP Parent
PRODUCT_LINE
 PROP Control text
establish mf_pl_abort_text;
 PROP Announce
You have entered profile :vl *rpl_name
the entries are:
:CE ON
FINANCE
Gross Margin is :vl *rpl_f_gmargin %
Payback Period is :vl *rpl_f_payback (years)
Rate of Return is :vl *rpl_f_return ($1,000,000s)
Volume of Business is :vl *rpl_f_vol ($1,000,000s)
MARKET FACTORS
Market Share is :vl *rpl_m_size %
Market Growth Rate is :vl *rpl_m_grw_rate %
CORPORATE FACTORS
Corporate Outlay is :vl *rpl_corp_outlay ($1,000,000s)
 PROP Max instances
1
 PROP Display Screen
SCREEN:PRO    P
 PROP Dyn   le Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_PL_ABORT_TEXT
 PROP Parent
MF_PL_FACTOR_TEXT
 PROP Control text
determine pl_final_flag;
 PROP Announce
At this time you may either abort the session by pressi
ng the PF3 key (if all you wanted to do was define a ne
w profile, it has been saved now), or you may continue
the session and compare a new product to an existing pr
ofile or one you have just defined.
 PROP Max instances
```

```
1
 PROP Display Screen
SCREEN:PROFDISP
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB BRAND_NAME
 PROP Parent
WHICH
 PROP Control text


access (select (mf_b_pr_lst));
determine mf_b_pr_old;

discover
-- use prof_get;
determine namehold;
ask mf_brand_values;
discover
-- use pl_vh_find;
discover
-- use pl_h_find;
discover
-- use pl_mod_find;
discover
-- use pl_l_find;
discover
-- use pl_vl_find;
determine pl_final_eval;
display pl_final_eval;
 PROP Max instances
1
 PROP Mult Choice Scr
SCREEN:CHOSE
 PROP Enter Value Scr
SCREEN:CHOSE
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_B_VH_FIGURE
 PROP Parent
BRAND_NAME
 PROP Control text
discover
```

```
-- use mf_b_vh_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_B_H_FIGURE
 PROP Parent
BRAND_NAME
 PROP Control text
discover
-- use mf_b_h_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_B_MOD_FIGURE
 PROP Parent
BRAND_NAME
 PROP Control text
discover
-- use mf_b_mod_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_B_L_FIGURE
 PROP Parent
BRAND_NAME
 PROP Control text
discover
-- use mf_b_l_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
```

```
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB MF_B_VL_FIGURE
 PROP Parent
BRAND_NAME
 PROP Control text
discover
-- use mf_b_vl_pick;
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 FCB PRODUCT
 PROP Parent
WHICH
 PROP Control text
display ask_type
 PROP Max instances
1
 PROP Dyn Rule Order
FALSE
 PROP DisposeWhenDone
FALSE
 PROP Param selection
ALL
 PROP Rule selection
ALL
 ENDKB
```