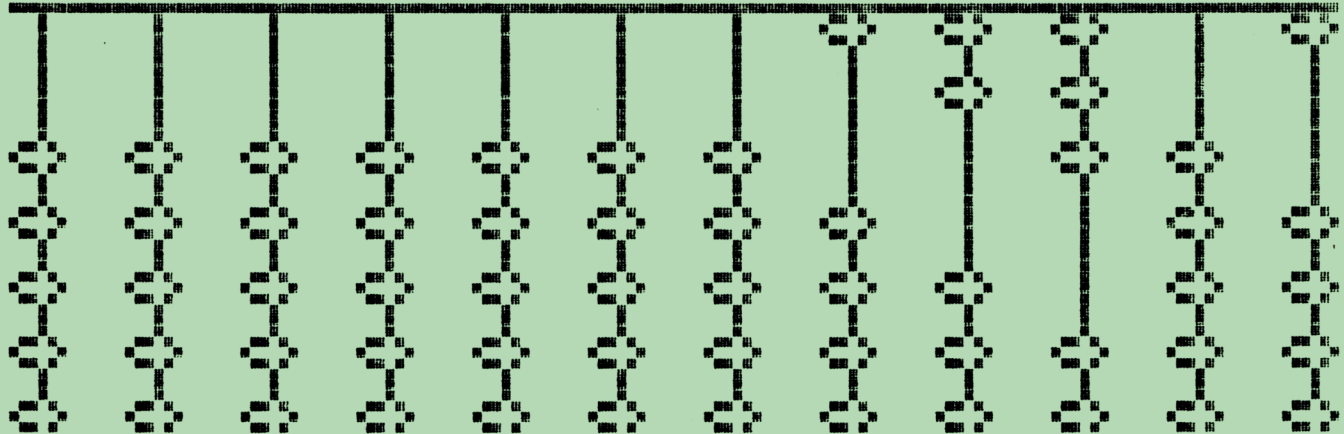
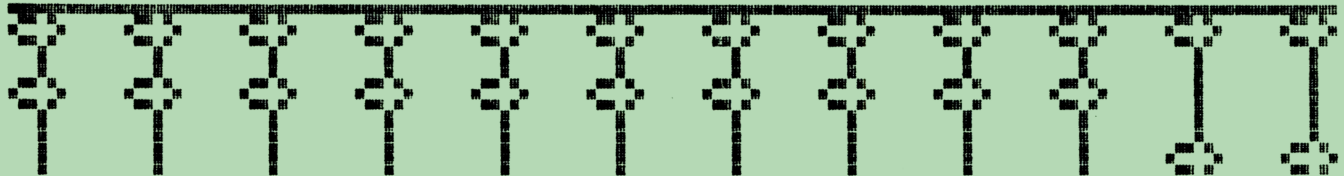


THE T/MAKER
USERS GROUP
NEWSLETTER



- JOB CANDIDATES EVALUATION
page 1
- MAILING LIST USED TO PERSONNALIZE LETTERS
page 6
- SOFTWARE COST PROJECTIONS
page 11

JOB CANDIDATES EVALUATION

Consider the situation where a 100 or more job candidates are to be ranked/ordered according to their experience and job related qualifications. Each candidate will not be evaluated in term of the job requirements only, but also in relation to other candidates as well. This is undoubtedly a tedious and time consuming task, if it is to be handled manually. Using T/MAKER, however, a simple but very practical procedure could be set up in minutes to make the task much easier and more flexible. The example presented in this article is an actual one. Although it may be a representation of a special case, with minor modifications it can be made to apply to many other similar cases.

A Senior Programmer/Analyst position was advertised at the Doit Right Organization with the following requirements;

- A college degree
- Good command of the English language
- Working knowledge of the IBM/DOS systems
- At least 5 years programming experience in COBOL or PL/I
- At least 2 years experience with a DBMS, preferably ADABAS

In addition to the above requirements candidates must have experience in the following areas:

- Project leadership
- System analysis and design of large systems
- Technical experience
- Interact with users at the professional level
- Users training and advisement

Applications logging

As each application is recieved for the above position, three types of information are compiled and logged in the T/MAKER table listed on the next page;

1- Applicants information:

- a- applicant's name
- b- nationality
- c- year of birth
- d- college degree

2- Minimum requirements. A value of "1" is entered in the appropriate column for each requirement met by the applicant, otherwise "0". As can be seen in the example table, a column is reserved for each requirement specified in the job description.

3- Experience factors. An applicant is evaluated on a scale from 1 to 5 in each of the following experience categories

- a- his/her experience as a project leader
- b- his/her experience as a system analyst/designer
- c- his/her experience as a technical expert
- d- his/her experience at the professional capacity
- e- his/her experience as a trainer/advisor

The above categories are further weighted according to their relative importance to the posted job. See formula #5 for example.

Candidates Evaluation for the Senior Programmer/Analyst Position

	9	9	9	9	9	9	9	9	9	999	999	999	999	999	999	999999999
EX																ZZ
uc1	+	+	+	+	+	=										
uc2	+	+	+	+		=										
uc3		+	+	+	+	=										
uc4		+	+	+		=										
uc5*										30	30	10	10	20		
uc6										+	+	+	+	+	=	
uc9															+	=
uc10-																365
uc11															+ply	
I	Applicant Information	I	Minimum Requirements	I	Experience Factors (1-5)	S	I									
I		I		I		c	I									
I	:Nat'lty/	I	ExperienceI	I		O	I									
I	: Birth yr/	I Edu : Sys : Lang: Prog: DBMSI	I Proj.:Anal.:Tech.:Prof.:TrainI r : Adj I													
I Name :	Degree I[Col]:[IBM]:[Eng]:[Yr]:[2yr]I A B C D	ILeadr:Dsign: Exp :Contr:Adv'rI e :ScoreI														
I-----:	--- -- --- I --- : --- : --- : --- : --- I - - - -	I --- : --- : --- : --- : --- I ----:----I														
I# :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :	I :
+	I Beaty, Niel : Bra 51 BA I 1 :	1 : 1 : 1 : 1 : 1 I	I 4 : 3 : 2 : 5 : 4 I	:	I :	I :										
+	I Lowery, Thomas : Ven 40 -- I 0 :	1 : 1 : 1 : 1 : 1 I	I 3 : 3 : 2 : 5 : 3 I	:	I :	I :										
+	I Gross, Joe : Bra 46 ?? I 0 :	0 : 1 : 1 : 0 : 0 I	I 5 : 2 : 5 : 3 : 3 I	:	I :	I :										
+	I Mayes, David : ??? 30 -- I 0 :	0 : 1 : 1 : 0 : 0 I	I 5 : 4 : 2 : 5 : 3 I	:	I :	I :										
+	I Hetzel, Mark : USA 59 CS I 1 :	0 : 1 : 0 : 0 : 0 I	I 2 : 2 : 3 : 5 : 3 I	:	I :	I :										
+	I Evans, Ann : USA 32 Ma I 1 :	1 : 1 : 1 : 1 : 1 I	I 4 : 5 : 3 : 3 : 5 I	:	I :	I :										
+	I Griffin, Virginia : USA 34 CS I 1 :	1 : 1 : 1 : 1 : 0 I	I 4 : 2 : 3 : 4 : 5 I	:	I :	I :										
+	I Matthew, Frank : Bra 45 CS I 1 :	1 : 1 : 1 : 1 : 1 I	I 3 : 5 : 3 : 4 : 5 I	:	I :	I :										
+	I Stewart, John : USA 32 -- I 0 :	1 : 1 : 1 : 0 : 0 I	I 5 : 3 : 4 : 4 : 4 I	:	I :	I :										
+	I Thomas, David : Col 46 BA I 1 :	0 : 1 : 0 : 0 : 0 I	I 1 : 2 : 3 : 4 : 2 I	:	I :	I :										
+	I Murphy, Howard : Arg 47 -- I 0 :	0 : 1 : 1 : 0 : 0 I	I 1 : 4 : 3 : 3 : 4 I	:	I :	I :										
+	I Janus, Margaret : Bra 23 En I 1 :	0 : 1 : 1 : 0 : 0 I	I 1 : 3 : 3 : 4 : 3 I	:	I :	I :										
+	I Hammonds, Peter : USA 31 CS I 1 :	1 : 1 : 1 : 1 : 1 I	I 4 : 5 : 4 : 5 : 4 I	:	I :	I :										
+	I Johnson, Paul : USA 50 -- I 0 :	1 : 1 : 1 : 0 : 0 I	I 3 : 4 : 4 : 5 : 4 I	:	I :	I :										
+	I Powell, David : Arg 38 -- I 0 :	1 : 1 : 1 : 0 : 0 I	I 3 : 4 : 4 : 3 : 4 I	:	I :	I :										
+	I Short, James : Arg 34 Ar I 1 :	1 : 0 : 0 : 0 : 1 I	I 3 : 3 : 2 : 4 : 2 I	:	I :	I :										
+	I Marks, Ellie : Ven 45 Sc I 1 :	1 : 1 : 0 : 0 : 1 I	I 3 : 3 : 2 : 4 : 2 I	:	I :	I :										
+	I Bryant, George : USA 45 Sc I 1 :	1 : 1 : 0 : 0 : 0 I	I 3 : 3 : 3 : 4 : 2 I	:	I :	I :										
+	I Drake, Jannie : Fr 51 Ec I 1 :	0 : 1 : 1 : 0 : 0 I	I 5 : 2 : 3 : 4 : 4 I	:	I :	I :										
+	I Little, Edwin : Mex 53 BA I 1 :	1 : 1 : 1 : 1 : 0 I	I 5 : 4 : 2 : 5 : 5 I	:	I :	I :										
+	I Sampson, Michael : Mex 36 CS I 1 :	1 : 0 : 1 : 0 : 0 I	I 4 : 4 : 3 : 3 : 3 I	:												

Grading the candidates

Once all applications have been logged in the proposed T/MAKER table/file the next step would be to execute the COMPUTE command against the table to generate the following values:

- a- Requirement A. This is the sum of all minimum requirements. A value of 5 indicates that a candidate meets all minimum requirements.
- b- Requirement B. The sum of requirements 1, 2, 3, & 4.
- c- Requirement C. The sum of requirements 2, 3, 4, & 5.
- d- Requirement D. The sum of erquirements 2, 3, & 4.

Requirements B, C, and D above are designed to enable waving one or two of the minimum requirements in favor of stronger experience factors.

e- Score. A score is computed for each candidate by multiplying each experience factor by its corresponding weight and then aggregating the result for all five experience categories.

f- Adjusted score. This value simply Zeros out all candidates below a certain score -in this case 365. Zero values are converted to "%%" to facilitate their elimination from the ranking operation.

Ranking the candidates

After having graded the candidates the next and final step would be to determine the criteria on which the candidates are to be ranked. Examples of possible criterias are:

1- Ranking the candidates according to their fulfillment of all requirements -Requirement A.

2- Ranking the candidates who have a degree in Computer Science (CS), according to their fulfillment of Requirement C.

3- Etc.

Taking the first criteria as an example, the following are the necessary commands to produce an ordered list of best qualified candidates.

```
GET table
COMPUTE
FIND EX
CLEAN
DROP %%
ARRANGE 200 220 1 35 66 68 105 116 END
FIND I*
SORT D 57 63
PRINT IT
```

The result of the above commands is a list in which candidates are ranked according to their qualifications to the Senior Programmer/Analyst position. The example listing on top of the next page shows Peter Hammond to be the most qualified candidate. Next best would be Ann Evans, and so on. Note, however, that waving the 2 years DBMS experience would result in Edwin Little ranking first along with Peter Hammond.

Applicant Information				S	I
				c	I
:Nat'lty/				I	I
: Birth yr/				I	I
Name	:	Degree	I	A	I
				e	:ScoreI
Hammonds, Peter	:	USA 31 CS	I 5 I	440:	75I
Evans, Ann	:	USA 32 Ma	I 5 I	430:	65I
Matthew, Frank	:	Bra 45 CS	I 5 I	410:	45I
Heywood, Blair	:	Bra 44 Ma	I 5 I	410:	45I
Little, Edwin	:	Mex 53 BA	I 4 I	440:	75I
Gusler, Melvin	:	USA 29 Ed	I 4 I	420:	55I
Stewart, John	:	USA 32 --	I 3 I	400:	35I
Johnson, Paul	:	USA 50 --	I 3 I	380:	15I
Gustely, Elmer	:	Pan 32 CS	I 3 I	370:	5I
Mayes, David	:	??? 30 --	I 2 I	400:	35I
I*	:		I	I	:
I	:		I	I	:

MAILING LIST USED TO PERSONALIZE LETTERS AND TYPE THE ENVELOPES

The advertising manager of the Stanford GSB newspaper wanted to send a personalized letter to every executive recruiter in the area, inviting them to advertise in the paper.

--SETTING UP THE ADDRESS LIST--

To begin with, she created a file called "hedhntrs" (headhunters) in which she typed every company name and address she could find in the yellow pages (this list contained 57 names and addresses, though a file could contain hundreds more). The file looked like this:

Headhunters International
 1 Progress Parkway
 Palo Alto, CA 94304
 Execu-find
 10 Tally-Ho Road
 Cupertino, CA 95014
 College Recruiting Service
 57 Campus Drive
 Stanford, CA 94305

(CONTINUED ON NEXT PAGE . . .)

Using the keystroke macro, she then turned every entry into one suitable to later use the load function:

Keystrokes were:	Line looked like:
BEGIN MACRO	Execu-find
INSERT CHARACTER (6 times)	Execu-find
01 = '	01 = 'Execu-find
END OF LINE '	01 = 'Execu-find'
RETURN	
EXECUTE MACRO 171 (57 times 3) times	

(then type over numbers as shown to get this:)

```
01 = 'Headhunters International'
01 = '1 Progress Parkway'
01 = 'Palo Alto, CA 94304'
02 = 'Execu-find'
02 = '10 Tally-Ho Road'
02 = 'Cupertino, CA 95014'
03 = 'College Recruiting Service'
03 = '57 Campus Drive'
03 = 'Stanford, CA 94305'
```

(Then use the following keystrokes:)

```
BEGIN MACRO
CHARACTER INSERT (4 TIMES)
NAME
RETURN
CHARACTER INSERT (4 TIMES)
ADRA
RETURN
CHARACTER INSERT (4 TIMES)
ADRB
RETURN
LINE INSERT
RETURN
EXECUTE MACRO 56 TIMES
```

(the list will then look like this)

```
NAME01 = 'Headhunters International'
ADRA01 = '1 Progress Parkway'
ADRB01 = 'Palo Alto, CA 94304'

NAME02 = 'Execu-find'
ADRA02 = '10 Tally-Ho Road'
ADRB02 = 'Cupertino, CA 95014'

NAME03 = 'College Recruiting Service'
ADRA03 = '57 Campus Drive'
ADRB03 = 'Stanford, CA 94305'
```

MAILING LISTS, CONTINUED

--SETTING UP THE LETTER--

At this point, her "hedhntsr" file is ready for use. Next, she set up the following file, named "headletr", using "load" formats as found in the T/Maker manual:

September 1, 1982

The Reporter
Graduate School of Business
Stanford University
Stanford, CA 94305

{<NAME01 }
{<ADRA01 }
{<ADRBO1 }

Dear Executive Placement Manager:

How would you like to find a few good Stanford MBAs with 5 years of impressive work experience and undergraduate degrees in engineering or business from top notch schools? How about a hundred of them, and all of them looking for work?

Greetings to you at {!NAME01 } from the Stanford Graduate School of Business student newspaper, the REPORTER.

Over two hundred organizations requested on-campus interviews last year, which began in October for the June graduates. Yet many of the students wishing to find employment in this area conduct searches on their own. These are the kind of people that {!NAME01 } can place now-- or place in the years to come, once the relationship is established.

The REPORTER is the only student publication distributed directly to each MBA student mailbox. As such, it is the most effective medium available to you to reach this community. Enclosed you will find all the information you need to place ads in the REPORTER. If you would like more information, please give me a call at the numbers listed on the brochure.

I look forward to hearing from you. Happy hunting . . .

Advertising Manager,
THE REPORTER

MAILING LISTS, CONTINUED

On the first line of the headletr file, she added the line:

```
LOAD HEDHNTRS ALIGN PRINT IT GET HEADLETR 2 REPLACE 01
```

--PRINTING THE LETTERS--

To print the letters, she types in :

```
WHAT NEXT?  Get HEADLETR 1 DO 02 1 DO 03 1 DO 04 1 DO
```

And so on, for as many letters as she wants. Address numbers can also be skipped, if not all the names on the list are to get letters. As T/Maker works through this command, it will get the letter, load the "01" name and address, align the letter, print it, then get the original letter, move to the line below the do line, change all 01s to the new number given, move back to the first line (the do line), and execute the commands to load, align, print, and move to the next one. You can see a resulting letter on the following page.

--PRINTING THE ENVELOPES--

To print envelopes, she set up another file named "envelope", which consisted of the following:

```
LOAD HEDHNTRS  PRINT IT GET ENVELOPE 2 REPLACE 001
```

```
    {<NAME001  
    {<ADRA001  
    {<ADRB001
```

To print envelopes, she used the same command as above:

```
WHAT NEXT?  get envelope DO 02 1 DO 03 1 DO 04 1 DO
```

--OTHER POSSIBILITIES--

By using other variables in the address list, you could include personal names, ages, children's names, account balances, and the like. Remember--you don't have to use every piece of information for every letter. For the numbering scheme, you could use customer numbers instead of just 1,2,3, etc. Other information in the address list could be used for sorts, (for example, to choose your biggest volume accounts), then you could send letters to only those accounts--you can selectively choose who should receive the letters, since the command need not run in numerical order.

Resulting letter:

September 1, 1982

The Reporter
Graduate School of Business
Stanford University
Stanford, CA 94305

Headhunters International
1 Progress Parkway
Palo Alto, CA 94304

Dear Executive Placement Manager:

How would you like to find a few good Stanford MBAs with 5 years of impressive work experience and undergraduate degrees in engineering or business from top notch schools? How about a hundred of them, and all of them looking for work?

Greetings to you at Headhunters International from the Stanford Graduate School of Business student newspaper, the REPORTER.

Over two hundred organizations requested on-campus interviews last year, which began in October for the June graduates. Yet many of the students wishing to find employment in this area conduct searches on their own. These are the kind of people that Headhunters International can place now--or place in the years to come, once the relationship is established.

The REPORTER is the only student publication distributed directly to each MBA student mailbox. As such, it is the most effective medium available to you to reach this community. Enclosed you will find all the information you need to place ads in the REPORTER. If you would like more information, please give me a call at the numbers listed on the brochure.

I look forward to hearing from you. Happy hunting . . .

Advertising Manager,
THE REPORTER

SOFTWARE COST PROJECTIONS

In this article T/MAKER is used in preparing and projecting the Biannual Computer Software Budget for the periods 1982-1987.

For each existing and planned piece of software, the following information was collected and/or estimated.

- Software description
- year installed
- type of financing (rent, lease, or purchase)
- cost of purchase
- monthly cost of rent/lease
- monthly cost of maintenance
- total months installed 1982-83
- total months installed 1984-85
- total months installed 1986-87

The objective here is to calculate the total cost of software in the biannual periods 82-83, 84-85, and 86-87. Given a 12% annual inflation rate, the T/MAKER procedure below is used to compute the expected cost of each piece of computer software in each of the biannual periods, and provide a summary of total software cost by installation date.

(CONTINUED ON NEXT PAGE . . .)

HOW ARE YOU USING T/MAKER?

DO YOU USE T/MAKER TO HELP YOU IN YOUR BUSINESS, YOUR HOBBY, OR YOUR HOME? WE ARE LOOKING FOR APPLICATION STORIES ABOUT HOW PEOPLE ARE USING T/MAKER. PLEASE SUBMIT YOUR IDEAS TO THE USER'S GROUP ADDRESS.

Software Cost Projections for periods 1982-1987

ex						999,999	99,999	9,999	999	999	999	999,999	999,999	999,999
zv							-	-	-	-	-	0	0	0
uc1							+	+	+			=		
uc2							+	+		+		=		
uc3							+	+		+			=	
uc4*												1.12	1.4	1.7
	Mfg	Description	Qty	Year	Install'n Terms	Cost of Purch	Monthly Rental	Cost Maint	Months 82-3	Inst'd 84-5	86-7	-Software 82-83	Rent/Leas 84-85	Cost- 86-87
+	IBM	System Control Program	1	1980	Rnt		-	450	24	24	24	12,096	15,120	18,360
+	IBM	VM/SP	1	1982	Rnt		250	50	18	24	24	6,048	10,080	12,240
+	IBM	VM IPCS Extensions	1	1980	Rnt		40	5	24	24	24	1,210	1,512	1,836
+	IBM	SCP - DOS/VS(E)	1	1981	Rnt		-	-	24	24	24	0	0	0
+	IBM	DOS/VS(E) Advanced func	1	1981	Rnt		165	50	24	24	24	5,779	7,224	8,772
+	IBM	DOS/VS(E) IPCS	1	1981	Rnt		30	5	24	24	24	941	1,176	1,428
+	IBM	DOS/VS(E) POWER	1	1981	Rnt		45	15	24	24	24	1,613	2,016	2,448
+	IBM	DOS/VS(E) VSAM	1	1981	Rnt		45	20	24	24	24	1,747	2,184	2,652
+	IBM	DOS/VS(E) IPF	1	1981	Rnt		35	5	24	24	24	1,075	1,344	1,632
+	IBM	DOS/VS(E) ICCF	1	1981	Rnt		73	25	24	24	24	2,634	3,293	3,998
+	IBM	DOS/VS(E) BTAM SCP	1	1981	Rnt		-	-	24	24	24	0	0	0
+	IBM	DOS/VS(E) Dev Supt FacI	1	1981	Rnt		-	-	24	24	24	0	0	0
+	IBM	DOS/VS(E) BTAM Extended	1	1981	Rnt		25	5	24	24	24	806	1,008	1,224
+	IBM	DOS/VS(E) CICS	1	1981	Rnt		360	120	24	24	24	12,902	16,128	19,584
+	IBM	DOS/VS(E) DITTO	1	1977	Rnt		35	5	24	24	24	1,075	1,344	1,632
+	IBM	DOS/VS(E) FastCopy	1	1977	Rnt		330	5	24	24	24	9,005	11,256	13,668
+	IBM	PVM Facility	1	1982	Rnt		125	10	12	24	24	1,814	4,536	5,508
+	IBM	DOS/VS(E) COBOL	1	1977	Rnt		140	15	24	24	24	4,166	5,208	6,324
+	IBM	DOS/VS(E) DMS	1	1977	Rnt		300	80	24	24	24	10,214	12,768	15,504
+	IBM	DOS/VS(E) FORTRAN IV	1	1977	Rnt		35	5	24	24	24	1,075	1,344	1,632
+	IBM	DOS/VS(E) PL/I	1	1971	Rnt		285	40	24	24	24	8,736	10,920	13,260
+	IBM	DOS/VS(E) RPG II	1	1979	Rnt		115	5	24	24	24	3,226	4,032	4,896
+	IBM	CMS/DS/COBOL	1	1983	Rnt		290	-	-	24	24	0	9,744	11,832
+	IBM	VIDEO/370 KEY ENTRY	1	1977	Rnt		250	95	24	-	-	9,274	0	0
+	IBM	CMS Doc Comp Facility	1	1983	Rnt		600	70	9	24	24	6,754	22,512	27,336
+	IBM	CMS Online Memb System	1	1983	Rnt		500	-	6	24	24	3,360	16,800	20,400
+	SAG	ADABAS/DBMS	1	1980	Pur	90,000	-	550	24	24	24	14,784	18,480	22,440
+	SAG	NATURAL	1	1980	Pur	30,000	-	300	24	24	24	8,064	10,080	12,240
+	SAG	COMPLETE	1	1984	Pur	40,000	-	150	-	18	24	0	3,780	6,120
+	CA	CA/SORT	1	1980	Lea		250	-	24	24	24	6,720	8,400	10,200
+	CA	Dynam/T	1	1980	Lea		450	-	24	24	24	12,096	15,120	18,360
+	CA	Dynam/D	1	1980	Lea		350	-	24	24	24	9,408	11,760	14,280
+	CA	Dynam/FI	1	1981	Lea		220	-	24	24	24	5,914	7,392	8,976

(CONTINUED ON NEXT PAGE . . .)

Software Cost Projections for periods 1982-1987

+	ADR Librarian (batch)	1	1979	Pur	10,000	-	70	24	24	24	1,882	2,352	2,856
+	ADR Librarian/VS	1	1982	Pur	4,000	-	120	24	24	24	3,226	4,032	4,896
+	ADR Vollie On-line	1	1980	Pur	5,000	-	200	24	24	24	5,376	6,720	8,160
+	JSI JARS Batch job acct	1	1981	Pur	7,000	-	70	24	24	24	1,882	2,352	2,856
+	JSI UMAX Online job stats	1	1981	Pur	4,000	-	40	24	24	24	1,075	1,344	1,632
+	JSI GOLD Gra Online Disp	1	1982	Pur	3,000	-	20	16	24	24	358	672	816
+	SAS SAS - Stat Anal System	1	1980	Pur	3,000	-	80	24	24	24	2,150	2,688	3,264
+	SAS SAS/GRAPH	1	1982	Pur	3,000	-	50	15	24	24	840	1,680	2,040
+	SAS SAS/FSP	1	1982	Pur	2,000	-	30	15	24	24	504	1,008	1,224
+	... Misc. future needs		1983	Rnt		500	-	6	24	24	3,360	16,800	20,400
+	... Misc. future needs		1985	Rnt		500	-	-	6	24	0	4,200	20,400
+	... Misc. future needs		1987	Rnt		500	-	-	-	6	0	0	5,100

ex
zv

9,999,999 , , , , 9,999,999 9,999,999 9,999,999

- - - -

Software Cost by Installation Date

		Purchase Cost	Biannual Software purchase & rental budget		
			82-83	84-85	86-87
		-----	-----	-----	-----
= +	Currently Installed	161,000	169,716	206,573	250,838
jc5		+	+=		
= +	1983 Additions	-	13,474	65,856	79,968
jc6		+		+=	
= +	1984 - 1985 Additions	40,000	-	47,980	26,520
jc7		+			+=
=+	1986 - 1987 Additions	-	-	-	5,100
		=====	=====	=====	=====
=	Total Software Costs	201,000	183,189	320,409	362,426

