Technology Refreshment Cost Estimating and Planning Model: User s Guide

NX001C1

June 2000

Virginia Stouffer
Robert Hemm

Technology Refreshment Cost Estimating and Planning Model: User s Guide

NX001C1

June 2000

Virginia Stouffer
Robert Hemm

The views, opinions, and findings contained in this report are those of LMI and should not be construed as an official agency position, policy, or decision, unless so designated by other official documentation.

LOGISTICS MANAGEMENT INSTITUTE
2000 CORPORATE RIDGE
MCLEAN, VIRGINIA 22102-7805

Executive Summary

The Federal Aviation Administration (FAA) has begun to use commercial off-the-shelf (COTS) components in its major air traffic control systems, but it lacks the means to predict COTS future years cost impacts, in terms of refreshment and maintenance on deferred refreshment. COTS computer-based systems are designed to be replaced every 3 years, but the FAA can t meet this cycle because of acquisition lead-time, year-to-year budgeting, and unexpected budget shortfalls. As COTS equipment ages, it becomes more difficult to maintain and replace. Commercial business cycles dictate when components and operating systems become obsolete.

We investigated the FAA s experience with COTS in several major systems, researched commercial firms COTS business practices, and looked at the Defense Department s experience with COTS. On the basis of FAA and commercial experience, we developed a model of COTS tech refresh and maintenance costs as they increase through time. We coded our findings into a Microsoft Excel interface for use by FAA cost and budget analysts.

The model will be delivered on compact disk to the FAA System Architecture and Investment Analysis Office and integrated product teams. FAA analysts will be able to use it to predict the future funding needs of COTS systems in any given year over a 30-year horizon. They can vary year of installation and the year of refreshment and use the model to derive optimal refresh cycles. Budget analysts can use it interactively in the budgeting process to determine the maintenance cost impact of delayed acquisition funding. We were also able to use the model to spontaneously derive a cost estimating relationship for use in predicting FAA s entire tech refresh cost burden.

This user s guide was written for all users, from the beginner having only passing familiarity with Excel to the advanced user, who can edit the model s macros to suit expanding needs. The guide begins with simple installation and operation instructions before going into greater technical detail about how the model actually works and how it can be expanded to cover other tasks. Finally, the guide describes the source of each estimating relation.

Chapter 1
Overview and Model Installation

OVERVIEW

The Federal Aviation Administration (FAA) has begun to use commercial off-the-shelf (COTS) components in its major air traffic control systems. But it lacks the means to predict COTS future years cost impacts, in terms of refreshment and maintenance on deferred refreshment. COTS computer-based systems are designed to be replaced every 3 years, but the FAA can t meet this cycle because of acquisition lead-time, year-to-year budgeting, and unexpected budget shortfalls. As COTS equipment ages, it becomes more difficult to maintain and replace. Commercial business cycles dictate when components and operating systems become obsolete.

We investigated the FAA s experience with COTS in several major systems, researched commercial firms COTS business practices and looked at the Defense Department s experience with COTS. On the basis of FAA and commercial experience, we developed a model of COTS refresh and maintenance costs as they increase through time. We coded our findings into a Microsoft Excel interface for use by FAA cost and budget analysts.

Model installation instructions are provided below in this chapter. Chapter 2 tells how to operate the model and describes the process flow of an analysis. Chapter 3 is a page-by-page discussion of the model spreadsheets. Chapter 4 addresses model expansion options. Chapter 5 describes the data sources.

MODEL INSTALLATION

System requirements: Windows 95 with Microsoft Excel 97 or better. 1MB (i.e., 1,000 KB) free space on your hard drive. Macros must be enabled when you load and use the model.

The Technology Refreshment Cost Estimating and Planning Model is originally packaged on a CD-ROM. It can be run from the CD if you don t want to make any permanent changes to the model. If you experience difficulty with the model running on CD-ROM (e.g., macros stop running and ask for debugging), you may have to install the model on your computer s hard drive.

To install the model on your computer, simply copy the file TechRef.xls to your preferred data directory on your computer. Open Excel, then use file—open to find TechRef.xls in your data directory. Select the filename and click on OK to

open the model. A warning screen may come up indicating that viruses can be stored in macros. The model as delivered on CD is free from viruses. Select Enable macros to continue.

Once the model opens, turn to Chapter 2 for operational instructions.

Chapter 2
Basic Operation

HOW TO OPERATE THE MODEL

The model was constructed to serve two functions: estimating time-sensitive COTS tech refresh and maintenance costs, and analyzing budget alternatives. The same parameters and functions are used in each, but since analysts using one function rather than the other will want specific data output, some elements of the model will be of greater interest than others, depending on the user.

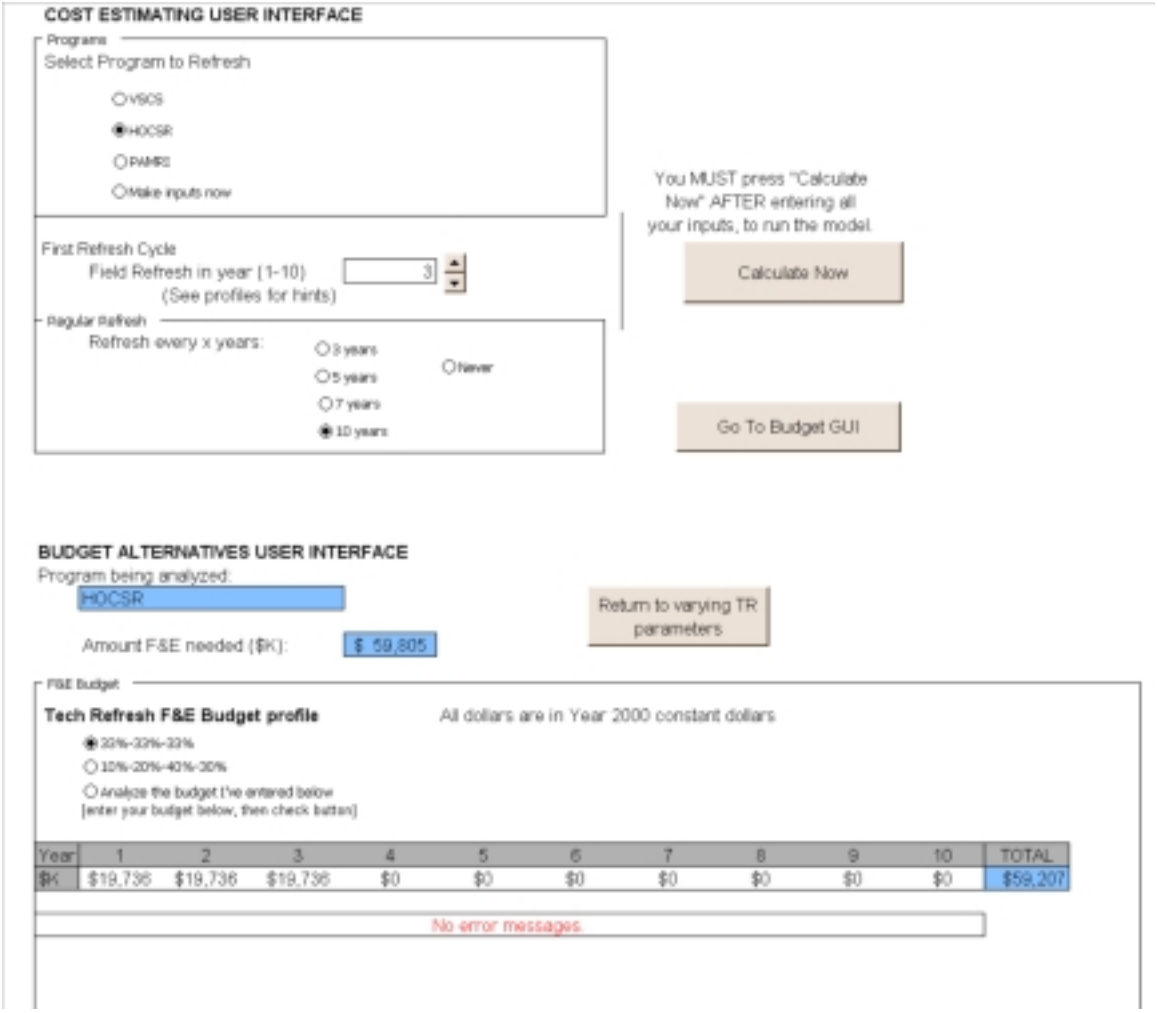
Figure 2-1 pictures the user interface; its use is described below. The full-page flow diagram (Figure 2-2) illustrates user inputs to operate the model.

Step-by-Step Guide

To begin using the model,

1. Turn to the User Interface worksheet, marked GUI on its tab. (See Figure 2-1.)
2. **Select a program to analyze**. Current choices are
   - VSCS,
   - HOCSR,
   - PAMRI, and
   - user-defined.

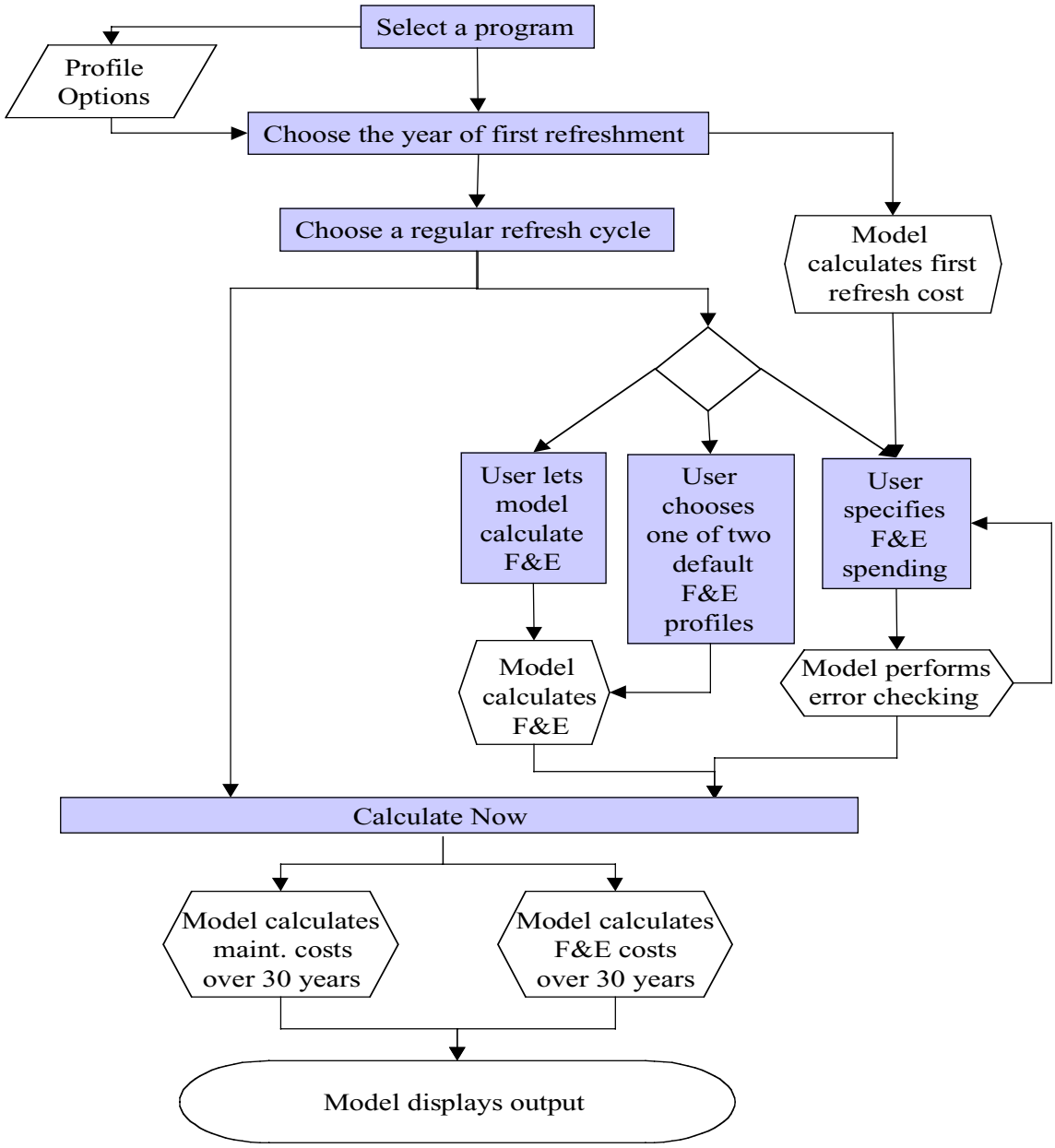
Figure 2-1. The GUI Worksheet



Select one of the pre-defined options by clicking on the radio button to the left of that program s text. If you are familiar with the model and wish to define a new program, please go to Chapter 4.

3. Use the spin button to **select a year to begin first tech refresh** for this program. This field determines the year in which the first refreshment will take place, relative to the current year.

Figure 2-2. User Flow Chart



A profile option chart can be accessed by clicking on the tab labeled Profile Option Chart. The profile chart illustrates combined default maintenance and acquisition expenditures associated with tech refresh for the program you have selected. Viewing the relative magnitudes of maintenance and acquisition expenditures on this chart may help the user choose a start year. The axis labeled Refresh in Year indicates a choice of first refresh year (i.e., refreshing a system when it is 1 year old, 2 years old, 3 years old, on up to 13 years). The axis labeled Spending merely lays out the years until the first refresh. Maintenance expenditures

grow over the years until first refresh. After first refresh, the chart is set to not depict any expenditures, though in fact maintenance costs will continue to be present. The final axis grades the magnitude of expenditures.

4. **Select a regular refreshment schedule** by pressing a radio button. This choice refers to how often you would like to plan to refresh your program after the first refresh. For example, you may schedule first refresh 5 years from now but wish to follow a regular 10-year cycle after that in that case you should select 5 in answer to step 3, and 10 in answer to step 4. Options supported by the model for regular refreshment are

   - every 3 years,
   - every 5 years,
   - every 7 years,
   - every 10 years, and
   - never.

   Your processor will process some short commands after you select any of these buttons. This is normal. Do not try to halt the processing. On most machines it will take a few milliseconds.

   The Never Refresh option will zero out the regular refresh cycle and the facilities and equipment (F&E)[1] budget; any inputs previously entered in those spaces will be overwritten with zeros.

5. If you would like to have the model automatically calculate an F&E budget for you, you can now go to step 9. If you would like to enter your own F&E expenditures, continue to step 6. If you chose Never Refresh, go to step 9.

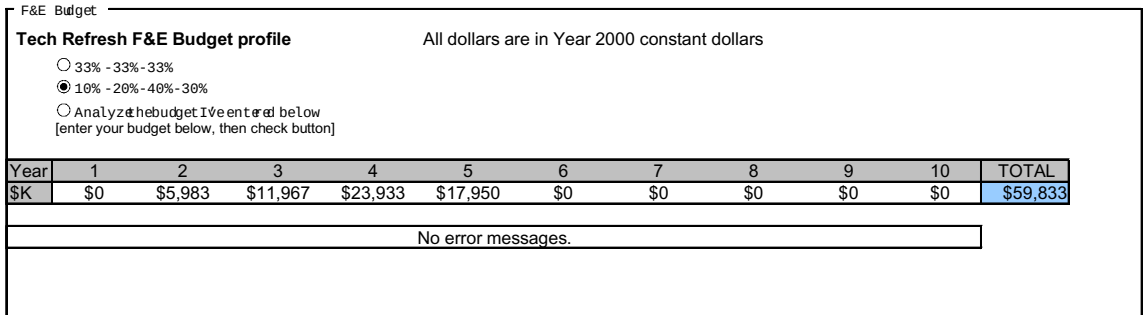
6. Your next choice is the F&E Budget profile. Figure 2-3 is an example. You can use the model s default budgets, which divide up the required F&E into 3- or 4-year spending profiles, or you can define your own F&E budget. The blue window above your profile tells you the amount of F&E you must spend to meet the model s projections for first refresh F&E. Note that the amount of money required to refresh your program increases as the year of first refresh moves farther into the future.

   To use the model defaults, click on a radio button next to the profile you prefer. At this time, the model supports two pre-defined budgets (33%—33%—33% and 10%—20%—40%—30%). Alternatively, to enter your own budget, click in the white boxes and begin entering numbers as you

[1] Facilities and equipment, or F&E, is used to refer to program acquisition costs.

would normally enter them in an Excel spreadsheet. Enter your budget in thousands ($K) of constant 2000 dollars. Each box should contain one year s worth of F&E funding. You can press Enter after each entry, but it is not necessary to do so. Use your cursor to advance to the next box if needed.

Figure 2-3. An Example of an F&E Budget Profile



The blue TOTAL box at the end of the budget line will automatically sum up your entries. You must enter an amount of funding equal to (or greater than) the displayed F&E requirement to avoid an error message.

You can enter expenditures in any year you like. The model will display an error message if your funding stream displays unusual characteristics, such as funding refresh after it occurs or expending funding more than two years before fielding. The error messages will appear in the text box below the funding entry boxes (See the No error messages box in Figure 2-3.)

7. If you receive an insufficient-funding message, you may re-enter your funding profile.

   Even if your funding stream causes an error message, the model will consider your inputs as the F&E stream for its calculations. See the section on Budgets in this chapter for further details.

8. When you are satisfied with your funding profile, click on the radio button associated with a user-defined budget profile. You may re-enter profiles and press this button as many times as necessary. Just be sure to press the user-defined (Analyze the budget I ve entered below) button after your final budget entry.

9. Page up and press the Calculate Now button.

Viewing Output

To view the output of your model run, click on the Output Chart tab. You will see a bar graph showing F&E and maintenance expenditures over a 30-year planning horizon for this program. If you would like to manipulate the output of this

model further, activate the show me the data button to access numerical output. This section can be copied and pasted onto another worksheet or document. Note that all numbers displayed on this sheet are constant 2000 dollars and that F&E is not in thousands as it is on the GUI sheet.

You can return to the graph by pressing the return to graph bu tton.

This graph is in a 3D format. To reverse the order of the bars displayed, press the switch rows in chart button, bringing the rearmost data series to the front of the chart.

If the chart seems to fail to reflect your tech refresh choices, return to the GUI tab, re-select your preferred options, and press the Calculate Now button. If the chart still fails to reflect your choices, see the section on debugging, below.

MODEL PURPOSE AND LIMITS

It is important that users understand the model s limits. This model is one tool that may be useful in certain contexts, and it should not be mistaken for something more than it is.

What the Model Does

The model estimates the life-cycle costs of certain COTS hardware and related operating system software. Its main usefulness derives from its ability to forecast relative costs for different refresh cycles and to help analysts plan and forecast the expenditures related to COTS refresh cycles.

The model takes a predefined set of COTS hardware and estimates

- its initial acquisition cost;
- its market life;
- its service life;
- its physical life;
- the cost of maintaining the hardware and operating system in each stage of those life periods;
- the cost of refreshing the hardware in any given year, at any stage of its lifecycle; and
- the cost of performing a customized rebuild for a failing subcomponent of the hardware, also known as a Service Life Extension Program, or SLEP.

Using these parameters, the model will tell you how much it will cost to refresh and maintain a given set of COTS hardware over the next 30 years. The model allows the user to vary refresh cycles and view their effects on costs. In addition, the model allows the budget analyst to do what if scenarios and project the impact of delayed F&E funding.

Underlying Algorithm

This section describes how the model relates hardware age to cost. The model deals primarily with computing and telecommunication equipment, such as computer processors, track balls, monitors, memory stacks, local area networks (LANs), switches, hubs, and routers. The model applies derived rules of thumb to classes of computers and telecommunication equipment to forecast their useful lives. In our model, hardware goes through distinct phases as it ages. The age of a piece of hardware is a determinant of its maintainability. The following definitions and examples will illustrate those phases. The examples should not be taken too literally; at this point the model is rather general in nature.

- Hardware—a physical component such as a computer, (e.g., a 386 computer), or a fiber optic LAN, or a TCP/IP dual-stack 4-port router.
- Operating system—the basic instruction set or set of protocols that runs the hardware (e.g., OS/2, MS-DOS IEEE 802.3, or IP version 4).
- Application—any software running on top of the operating system, often proprietary or unique, such as Windows 95, radar display update system, flight data.

Table 2-1 illustrates some examples.

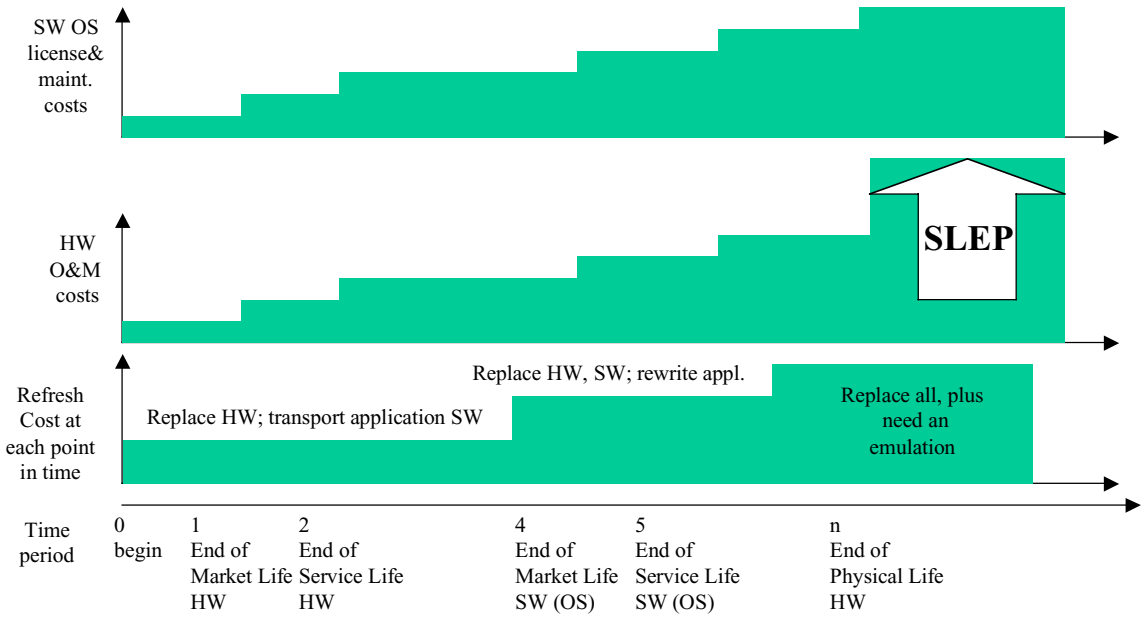
Table 2-1. Examples of Hardware, Operating Systems, and Applications

	Example 1	Example 2	Example 3
Hardware	386 computer	Fiber optic LAN	TCP/IP dual-stack 4-port router
Operating system	MS-DOS	IEEE 802.3	IP v.4
Application	Windows 3.0	Radar display update system	Encoded information and data

A piece of hardware goes through several periods in its life. The length of each varies by hardware type and is driven by commercial product cycles rather than by reliability. Figure 2-4 depicts these phases. When a piece of hardware (HW) is first introduced, it enjoys full product support from its manufacturer until its replacement is introduced. During the period in which hardware version x is the state of the art, it is generally one of only two generations of that hardware available for sale as new equipment. The period in which the hardware is offered

for sale by the original equipment manufacturer is the market life of the equipment. Market life ends when hardware x is no longer for sale from the manufacturer. The end of market life corresponds to the introduction of hardware version x+2. The end of market life also corresponds to the end of free manufacturer service; however, service can often be obtained from contractors.

Figure 2-4. How Maintenance, Licensing, and Refresh Costs Increase over Time for COTS Computer Equipment



There comes a time when maintenance cannot be obtained for the hardware on the open market. This is the end of the service life. From this point on, maintenance on the hardware can be obtained only from previously arranged long-term contracts and internal staff. Generally the end of service life occurs at a point twice the age of end of market life. Contracts for maintaining equipment after the service life has ended are exorbitantly expensive, if available at all. There is no guarantee that service is available at all during this period.

This period of exorbitant support is ended by the end of physical life. The end of physical life is generally reached when one subcomponent of the hardware begins failing en masse. An example is the power supply of a particular processor. For that generation of hardware, a particular subcomponent is the weak link and fails in most or all equipment. The owner of the hardware must backward engineer that subcomponent in order to continue to maintain the hardware. This backward engineering, part of a SLEP, extends the life of the hardware for a few more years but is very expensive.

The idea behind the algorithm is illustrated in a general way in Figure 2-4. Note that Figure 2-4 encompasses the possibility that support providers continuously raise prices through the life of the hardware and operating system.

The operating system of the hardware, referred to as software (SW) in the model, follows a similar lifecycle, differing only in the lengths of time and the pricing of maintenance and acquisition. The software generally is supported longer, since a relatively mature operating system does not need replacement parts. In software maintenance we also include any licensing fees. Annual l i-cense fees are often charged on operating systems, compilers, and proprietary middleware. The license fee is merely the annual rent for keeping the software operating. Software with a license fee usually includes a license-checking feature and disables itself if proper codes from the manufacturer are not entered before the software s due date.

Software can also reach the end of its physical life, when it cannot be supported any longer. A number of factors can trigger this: support for the operating system cannot be obtained for any amount of money; the legal owner of the operating system refuses to license the operating system further; the software was buggy to begin with and cumulative bug fixes have made it less reliable or too complicated to modify further; or the application s demands outgrow the operating system s operating parameters.

A combined hardware and software system reaches the end of its operating life when either the hardware or the software reaches the end of its physical life.

These life periods and their associated cost magnitudes have been determined for a finite set of computer and telecommunications equipment and programmed into the model as parameters. Costs and duration of the periods vary by equipment: that is, since a personal computer and a fiber optic cable have different life spans, parameters have to be defined individually for each class of hardware.

The model takes a defined set of input hardware and forecasts the maintenance expense of each component in each period of its life. Then, using the analyst s inputs on refreshment, the model projects maintenance costs until and after re-freshment. The model also predicts refreshment costs on the basis of the period-age of the equipment at the time of refreshment.

EXAMPLE

A new multi-processor computer costs approximately $35,000; its operating sys-tem, $1,000. The hardware is on the market for 3 years and the operating system is on the market for 5 years. During market life, hardware and software support contracts cost approximately $1,000 per year.[2] After the end of its market life, bridge contracts are entered into to provide maintenance and support for the hardware and operating system; they increase in price at the rate of 20 percent per

[2] Generally a manufacturer will offer free limited support during market life. However, to se-cure maintainability after the end of market life, or to receive 24 hour per day, 7 days per week support, a long-term support contract must be procured up front; thus an agency is usually unable to take advantage of the free support period.

year. After 5 years, the processor reaches its end of service life. After 10 years, the hardware and software together are no longer supportable.

To refresh this system before the end of its (software) market life, an agency would need only to procure new hardware and transport the application software. After the end of the software market life, new hardware will not run the old operating system. New software must be procured and a gap-filling program written to ease the transport of the application software to the next-generation operating system. After the end of physical life, generally the operating system has evolved substantially and cannot run the application at all, so new hardware and software must be accompanied by re-writing the application software or writing an emulation environment. These increasing requirements on technical refreshment are reflected in increasing costs to refresh as the COTS systems age.

These parameters do not hold for every piece of hardware, or even for all processors. Some bridge contracts begin at the end of market life, some at the end of service life. Some bridge contracts end at the end of physical life; others can be continued indefinitely. Every piece of hardware must be considered separately to determine its lifetime. An undisturbed LAN is projected to have a 30-year service life and a 60-year physical life [see Reference 102] and to need only minimal maintenance throughout, though user demands will probably render the LAN obsolete before it is physically inoperable. Some hardware does not have an operating system, simplifying the analysis of its maintenance costs and its replacement.

Additional Estimation

The remaining costs of a procurement, including program management, logistics, testing, training, spares, deployment, etc., are estimated as a percentage of Prime Mission Equipment.

Assumptions

The following assumptions apply:

1. The FAA will always procure maintenance agreements on COTS hardware, even if the manufacturer supplies free support for the first market years.

2. Contractor and FAA maintenance have comparable costs.

3. If the FAA chooses to not refresh a COTS system, the FAA will follow the path of lowest cost as the system reaches obsolescence; that is, it will choose to run a SLEP rather than face system outages, if at all possible.

4. Stretching the F&E funding of a tech refresh over many years does not in itself increase program costs, and there is no minimum program level in any funding year.[3]

5. A tech refresh is not considered fielded until 90 percent of its required F&E has been expended.

6. The FAA will procure rather than lease hardware. F&E funding always precedes fielding.

7. F&E funding always falls within 2 years of fielding

8. For simplicity, all fielding takes place in a single year, the year in which F&E funding equals 90 percent of the required tech refresh budget.

9. For maintenance costing purposes, it is assumed that a manufacturer will never refuse to license software but instead will continue to charge ever-escalating license fees.

10. In order to project a hypothetical 20-year cost profile for the Never Refresh case, it is assumed that a system will undergo SLEP every 3 to 5˚years (depending on the hardware) after its end of physical life.

Budget Analysis

User-Inputted Budgets and Error Checking

The model performs error checking on a user-inputted budget. The model notifies the user if the inputted budget does not conform to the model s requirements for funding. The model checks for

- F&E less than required refresh costs,
- A majority of F&E expended more than 2 years in advance of first refresh, and
- Less than 90 percent of required F&E expended before first refresh.

However, the model will allow most erroneous budget inputs to be used as the preferred first refresh budget profile. This was done in case the user wishes to input a nonstandard budget for a particular reason not encompassed in the planning of the model.

[3] We realize that many products require a critical mass, a period in which funding is sufficient to support staffing of several specialized tasks simultaneously; for instance, funding a logistician and system developer while the program is still being coded and built. Also, that keeping a project office open throughout the project is necessary, so that funding should never drop to zero but to some maintenance level. These requirements are not reflected in the model at this time.

The exception is insufficient funding. The model will not schedule a tech refresh and maintenance costs will not decrease over the first 10 years unless sufficient refresh money is included in the F&E budget. The model was programmed with this feature to show how immediate cuts in the F&E budget will cause increases in maintenance costs. Note that a 3 percent error margin is built into the comparison feature so that rounding errors will not cause the budget to return an error of insufficient funding.

The following examples (Figure 2-5 through 2-12) will illustrate the model s capabilities:

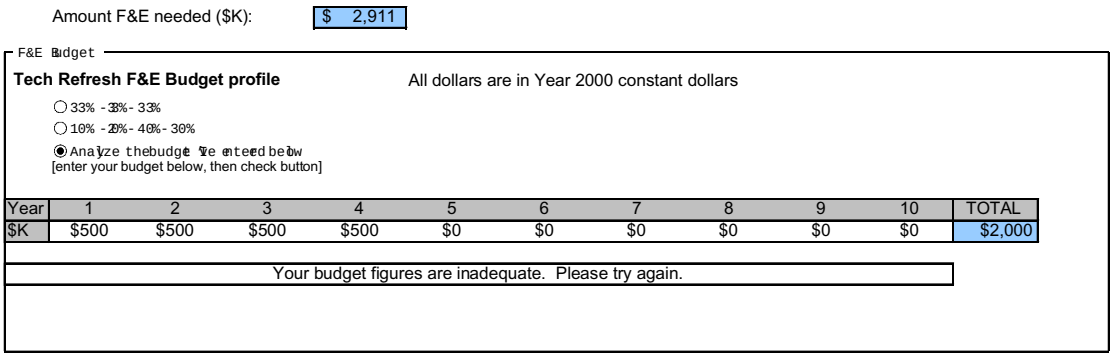
EXAMPLE 1

First Refresh: year 5

Required F&E: $2,911 K

You enter $2,000 K.

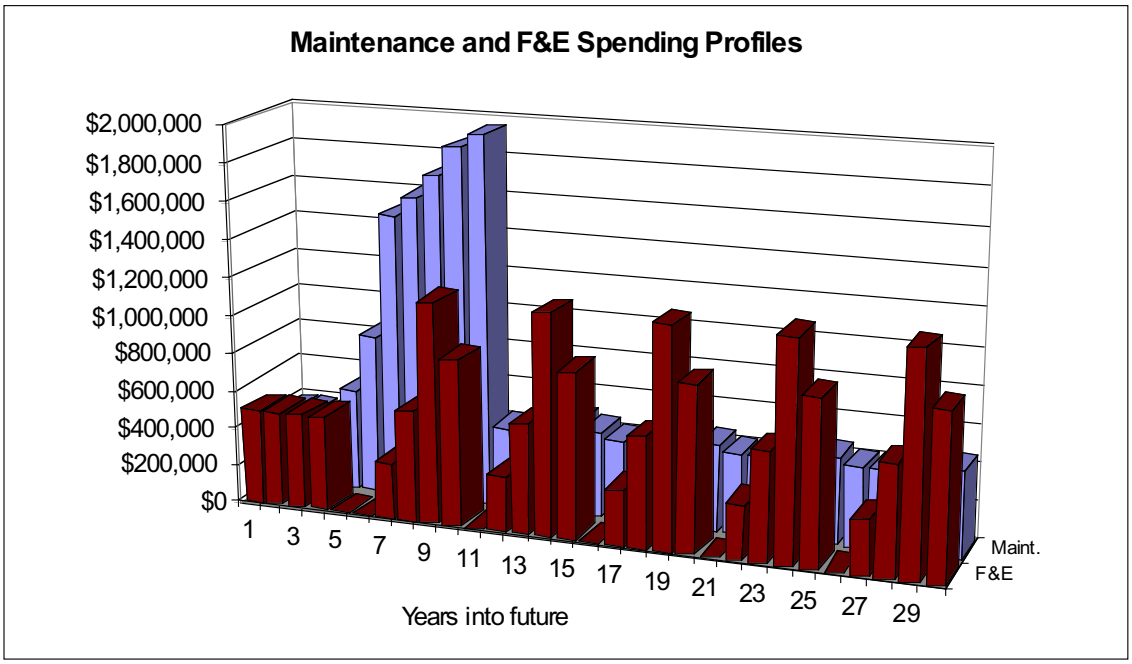
Figure 2-5. Example 1 Budget Profile



Your error: insufficient F&E funding to meet tech refresh requirements.

Model output: The model uses the (insufficient) user-input budget for the first refresh cycle; but refresh is never achieved because it is not fully funded. However, the repeat refresh cycles are automatically budgeted for and the model executes refresh cycles every 5 years, with automatic 4-year spending profiles.

Figure 2-6. Spending Streams Resulting from Example 1 Inputs



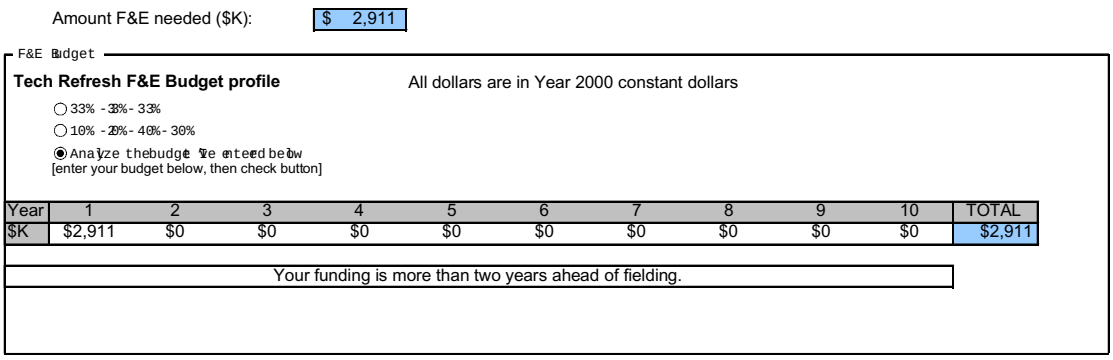
EXAMPLE 2

First Refresh: year 5

Required F&E: $2,911 K

You enter all funding in the first year.

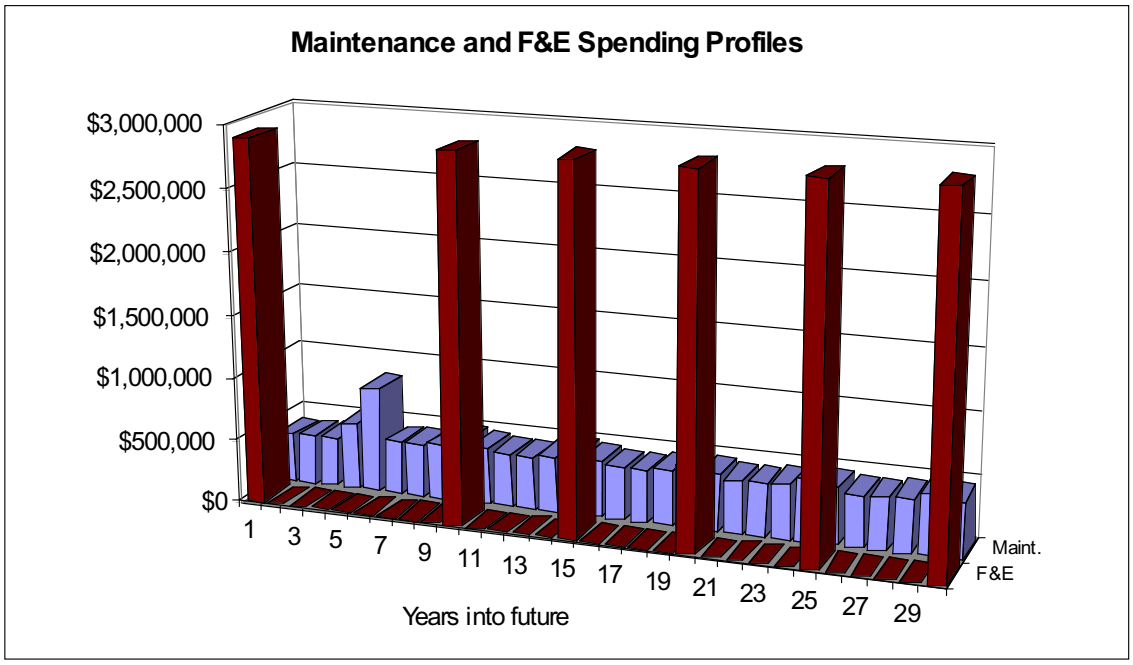
Figure 2-7. Example 2 Budget Profile



Your error: funding is considerably ahead of fielding.

Model output: Since your funding was sufficient, the model uses your funding profile for the first refresh. However, for regular refresh, the model moves your funding to the fielding year.

Figure 2-8. Spending Streams Resulting from Example 2 Inputs



If you want to move funding to a later year, you may do so and re-run the model, which was done in Example 3, as depicted in Figure 2-9 and Figure 2-10.

EXAMPLE 3

First Refresh: year 5

Required F&E: $2,911 K

You enter all funding in the fifth year.

Figure 2-9. Example 3 Budget Profile

Amount F&E needed ($K): $ 2,911

F&E Budget

Tech Refresh F&E Budget profile

All dollars are in Year 2000 constant dollars

○ 33% - 33% - 33%

○ 10% - 20% - 40% - 30%

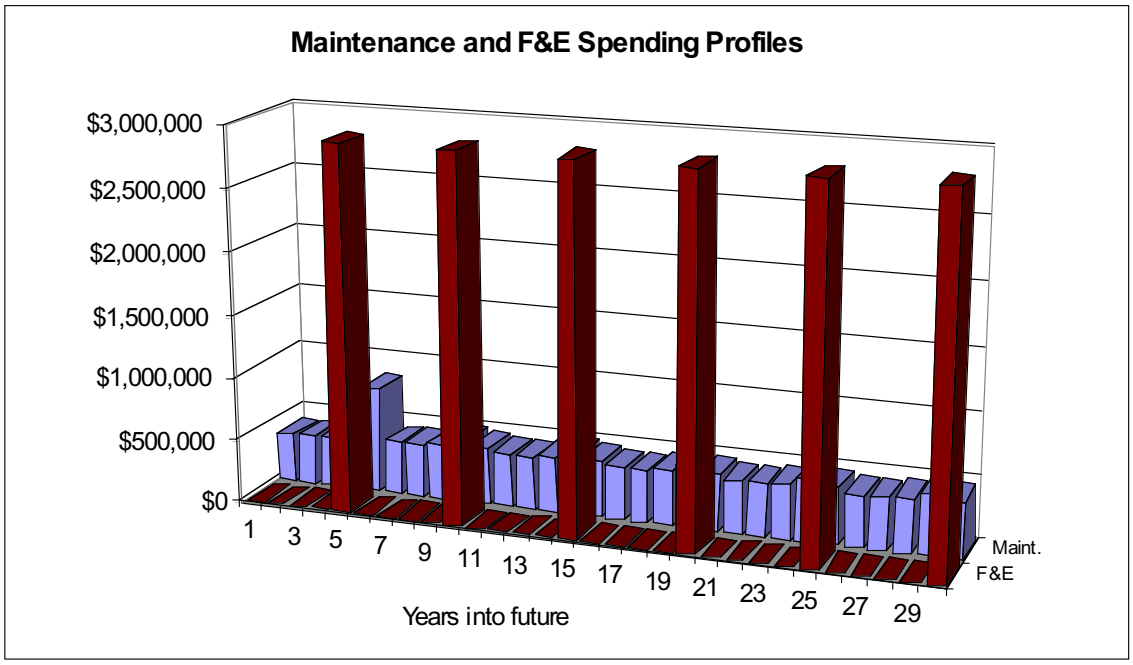
◉ Analyze the budget I've entered below
[enter your budget below, then check button]

Year	1	2	3	4	5	6	7	8	9	10	TOTAL
$K	$0	$0	$0	$0	$2,911	$0	$0	$0	$0	$0	$2,911

No error messages.

There are no errors. The model considers this an optimum spending profile and uses it again for future tech refresh cycles, as can be seen in Figure 2-10.

Figure 2-10. Spending Streams Resulting from Example 3 Budget Inputs



EXAMPLE 4

First Refresh: year 5

Required F&E: $2,911 K

You enter an actual funding profile that has been stopped and re-started.

Figure 2-11. Example 4 Budget Profile

Amount F&E needed ($K): $ 2,911

F&E Budget

Tech Refresh F&E Budget profile

All dollars are in Year 2000 constant dollars

○ 33% - 33% - 33%
○ 10% - 20% - 40% - 30%
◉ Analyze the budget profile entered below
[enter your budget below, then check button]

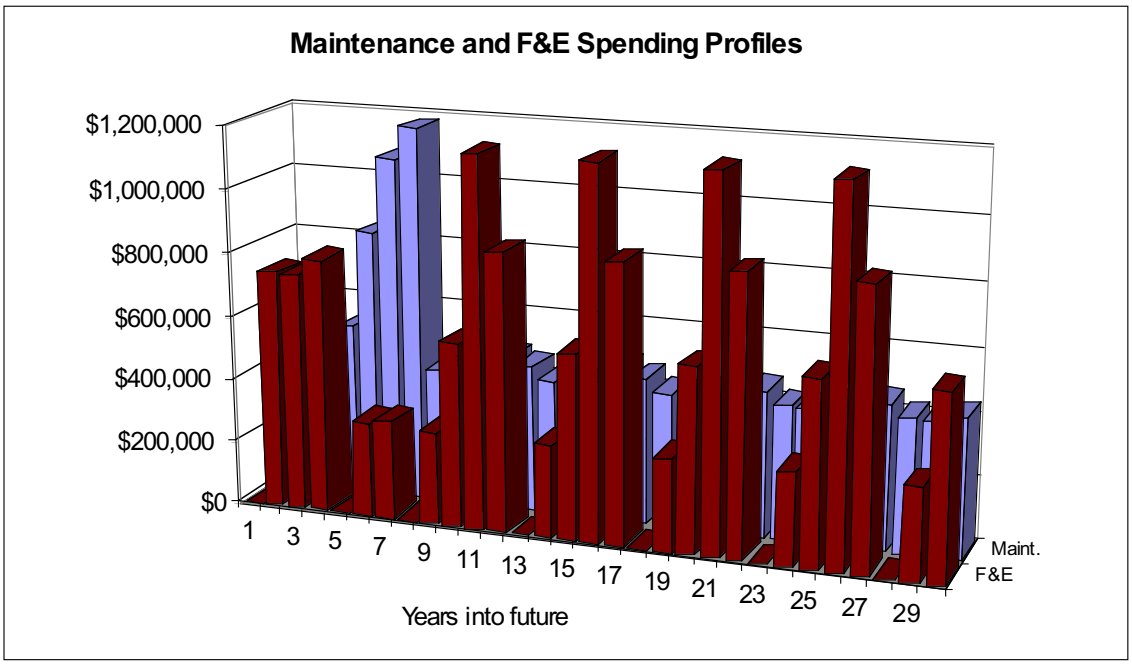
Year	1	2	3	4	5	6	7	8	9	10	TOTAL
$K	$0	$750	$750	$800	$0	$300	$311	$0	$0	$0	$2,911

No error messages.

Result: Your funding profile is sufficient, so it is used. The 90 percent benchmark for fielding refresh is reached in year 7, so the model fields refresh in year 7, even though the user had indicated first refresh in year 5. (The budget sufficiency checking overrides all else.) Because your budget profile stops and starts and ex-

ceeds your planned refresh date, the model generates an automatic budget profile for the regular refreshment years.

Figure 2-12. Spending Streams Resulting from Example 4 Inputs



Important Limitations of the Model

The model does not yet estimate application software acquisition, testing, or development, which may easily constitute the majority of costs of any procurement project.

The model does not have any utility for the amount of developmental items a procurement will need to field a particular system. These costs are not modeled.

The model is not capable of independently producing a cost estimate of any FAA program for which it has not been programmed. The FAA programs that the model is capable of addressing are listed with radio buttons on the GUI sheet.

This model is rule-based and as such represents a coarse level of measurement. It cannot help a procurement officer differentiate between competing COTS offerors.

This model does not currently support a leasing scenario, or power by the hour payments. Future versions could be re-programmed to include a leasing option under F&E options. It is possible to *emulate* a leasing option by entering an F&E budget profile of equally divided payments. For example, if the refresh cycle is 5˚years and the F&E cost is $100,000 K, enter 5 years of funding of $20,000 K each, and run the model as usual.

TROUBLESHOOTING

Macros Stopping

Most of the computation problems in the model will be caused by incomplete transference of data from one location to another. The first step when any anomalous outcomes are spotted is to return to the GUI sheet and re-activate Calculate Now. If needed, re-click your desired inputs and click on Calculate Now again.

If the macro continues to fail to run, re-install the model from the CD onto your hard drive. Before doing so, save any additions you have made to the model on a separate spreadsheet, so that you can copy them to the new model.

All the macros in the model refer to particular ranges and particular model sheets. If you have renamed any sheets or inserted rows, columns, or cells, the macros will function incorrectly. Re-installing the model will cure this problem.

It is possible to add sheets to the model when it is installed on your hard drive. As long as they do not have the same name as the original model s sheet tabs, they should not cause errors.

Illogical Results

If the user finds the results defying logic, the error is probably in the algorithm or input of key parameters. For example, if maintenance costs decrease as the system ages, it is likely that a value on the Parts worksheet is causing that effect. It may be the case that the bridge support growth rate was accidentally entered with a negative number, or that initial maintenance costs were overstated by an order of magnitude. To find the source of illogical results, first examine the maintenance cost streams on the Calc2 worksheet. It should be possible to identify one stream as causing the strange result. Follow that cost stream back to its source worksheet, e.g., HW maint, SW maint, or Refresh. Follow the reasoning of the algorithm on this worksheet to determine which year the anomaly begins to occur, then consult the Parts worksheet, looking for the parameters that describe that period.

Contents

Chapter 1 Overview and Model Installation 1-1

OVERVIEW 1-1

MODEL INSTALLATION 1-1

Chapter 2 Basic Operation 2-1

HOW TO OPERATE THE MODEL 2-1

Step-by-Step Guide 2-1

Viewing Output 2-5

MODEL PURPOSE AND LIMITS 2-6

What the Model Does 2-6

Underlying Algorithm 2-7

Additional Estimation 2-10

Assumptions 2-10

Budget Analysis 2-11

Important Limitations of the Model 2-16

TROUBLESHOOTING 2-17

Macros Stopping 2-17

Illogical Results 2-17

Chapter 3 Page-by-Page Explanation of the Model 3-1

INTRODUCTION AND DEFINITIONS 3-1

GUI 3-1

Purpose of GUI Worksheet 3-1

FAA Program Options 3-2

First Refresh 3-2

Regular Refresh 3-3

Buttons 3-4

Budget GUI 3-4

Calculate Now 3-4

PROFILE OPTION CHART 3-5

OUTPUT CHART 3-5
CALC1 3-5
CALC2 3-6
FAA SYSTEMS 3-9
FAA SYSTEM OUTAGES 3-9
PARTS 3-10
HW MAINT 3-12
SW MAINT 3-13
REFRESH 3-14
BUDGET 3-15
MODULES AND MACROS 3-18
MACROS OF THE BUDGET MODULE 3-19
userbudsel1() 3-19
userbudsel2() 3-20
userbudsel3() 3-20
choose_budget(userbudvar) 3-20
Default_budget1() 3-20
Default_budget2() 3-20
paste_input_budget() 3-20
MACROS OF THE FAA PROGRAMS MODULE 3-21
VSCS_prgm_data() 3-21
HOCSR_pgm_data() 3-21
PAMRI_pgm_data() 3-21
clear_old_prgmdata() 3-21
new_prgm_data1() 3-22
MACROS OF THE FINISHING MODULE 3-22
output() 3-22
show_data() 3-22
return_to_graph() 3-22
gotobudgetGUI() 3-23
return2CEGUI() 3-23
switch_data_series() 3-23
MACROS OF THE HOUSEKEEPING MODULE 3-23

MACROS OF THE MAIN ENGINE MODULE 3-23
Define_cycles_3() 3-23
Define_cycles_5() 3-24
Define_cycles_7() 3-24
Define_cycles_10() 3-24
Horizons() 3-24
choose(styear1, styear2, styear) 3-25
select_budget_horizon(budyrs, trcycle, funnyfunding, NSF, styear, budhor) 3-26
Pull1 Through Pul13 3-26
Start_cycles_right(styear) 3-27
Push1 through Push10 3-27
MACROS OF THE NVRREFR MODULE 3-27

Chapter 4 Expanding the Model 4-1
ADDING FAA PROGRAMS 4-1
Temporary Data Input 4-1
Permanent Data Input 4-2
ADDING TO THE PARTS DATABASE 4-3
Crane Surface Naval Warfare Center COTS Refresh Model 4-3
ADDING OR CHANGING ESTIMATING RELATIONSHIPS 4-4
Example 1. Change the Lifetime of a Component 4-4
Example 2. Change Several Parameters of a Component 4-4
ADDING SOCIAL COST 4-5
System Outage Disruption Model 4-6
Aviation System Analysis Capability 4-6
SOFTWARE ESTIMATION 4-8
GRADUATED REFRESHMENT 4-8
AUTOMATIC OPTIMIZER 4-12
FAA-WIDE TECH REFRESH ESTIMATING 4-13

Chapter 5 Data Sources 5-1
BASIC ESTIMATING RELATION 5-1
FAA PROGRAM STORIES 5-1
VSCS 5-2

HOCSR 5-3
PAMRI 5-5
STARS 5-7
ASR-9 5-7
Lessons Learned 5-8
PARTS DATABASE TABLE OF SOURCES 5-9

References

Appendix A Abbreviations

FIGURES

Figure 2-1. The GUI Worksheet 2-2
Figure 2-2. User Flow Chart 2-3
Figure 2-3. An Example of an F&E Budget Profile 2-5
Figure 2-4. How Maintenance, Licensing, and Refresh Costs Increase over Time for COTS Computer Equipment 2-8
Figure 2-5. Example 1 Budget Profile 2-12
Figure 2-6. Spending Streams Resulting from Example 1 Inputs 2-13
Figure 2-7. Example 2 Budget Profile 2-13
Figure 2-8. Spending Streams Resulting from Example 2 Inputs 2-14
Figure 2-9. Example 3 Budget Profile 2-14
Figure 2-10. Spending Streams Resulting from Example 3 Budget Inputs 2-15
Figure 2-11. Example 4 Budget Profile 2-15
Figure 2-12. Spending Streams Resulting from Example 4 Inputs 2-16
Figure 3-1. Spinner Box 3-3
Figure 3-2. Auditing Variables 3-9
Figure 4-1. Flow Diagram of ASAC and the Integrated System Safety Model 4-7
Figure 4-2. Radar Outage Example in System Safety Integrated Model 4-7
Figure 4-3. Step 1 Enter Sub-programs on FAA Systems Worksheet 4-9
Figure 4-4. Step 2 Enter Desired Parameters for the First Sub-program on the GUI Worksheet 4-10
Figure 4-5. Step 3 Copy the Output Numbers from the First Program Run to Another Worksheet 4-10
Figure 4-6. Step 4 Run the Second Sub-program 4-11

Figure 4-7. Step 5 Copy the Output Numbers from the Second Program Run to Another Worksheet 4-11

Figure 4-8. Step 6 When You Are Done Running Sub-programs, Combine Your Data on the New Worksheet to Tell the Proper Story 4-12

Figure 4-9. Comparison of Algorithm Predictions with Data 4-14

Figure 5-1. Graphic from a PowerPoint Presentation Showing HOCSR Phase 1 Components Replacement 5-5

Figure 5-2. PAMRI in Relation to Radar Input and the Host Computer System 5-6

TABLES

Table 2-1. Examples of Hardware, Operating Systems, and Applications 2-7

Table 3-1. List of Macros and Their Module Locations 3-18

Table 3-2. Key Variables Functions and Locations 3-25

Table 4-1. Refresh Cost as a Ratio of Acquisition Cost 4-13

Table 5-1. Sources for Parts Worksheet Entries: Reliability and Initial Hardware Cost 5-10

Table 5-2. Sources for Parts Worksheet Entries: Initial Hardware Support Cost 5-11

Table 5-3. Sources for Parts Worksheet Entries: Initial Bridge Hardware Support Cost 5-12

Table 5-4. Sources for Parts Worksheet Entries: Bridge Hardware Support Cost Growth Rate 5-13

Table 5-5. Sources for Parts Worksheet Entries: Initial Software Cost 5-13

Table 5-6. Sources for Parts Worksheet Entries: Initial Software Support Cost 5-14

Table 5-7. Sources for Parts Worksheet Entries: Second Software Support Cost 5-15

Table 5-8. Sources for Parts Worksheet Entries: Second Software Support Cost Growth Rate 5-15

Table 5-9. Sources for Parts Worksheet Entries: Program Cost Percentage 5-16

Table 5-10. Sources for Parts Worksheet Entries: End of Hardware Market Life 5-16

Table 5-11. Sources for Parts Worksheet Entries: End of Hardware Service Life 5-17

Table 5-12. Sources for Parts Worksheet Entries: End of Hardware Phys ical Life 5-18

Table 5-13. Sources for Parts Worksheet Entries: End of Software (Operating System) Market Life 5-18

Table 5-14. Sources for Parts Worksheet Entries: End of Software (Operating System) Service Life 5-19

Table 5-15. Sources for Parts Worksheet Entries: End of Software Physical Life 5-20
Table 5-16. Sources for Parts Worksheet Entries: SLEP Cost 5-20
Table 5-17. Sources for Parts Worksheet Entries: End of Life and Failure Rate After EOL 5-21

Chapter 1
Overview and Model Installation

OVERVIEW

The Federal Aviation Administration (FAA) has begun to use commercial off-the-shelf (COTS) components in its major air traffic control systems. But it lacks the means to predict COTS future years cost impacts, in terms of refreshment and maintenance on deferred refreshment. COTS computer-based systems are designed to be replaced every 3 years, but the FAA can t meet this cycle because of acquisition lead-time, year-to-year budgeting, and unexpected budget shortfalls. As COTS equipment ages, it becomes more difficult to maintain and replace. Commercial business cycles dictate when components and operating systems become obsolete.

We investigated the FAA s experience with COTS in several major systems, researched commercial firms COTS business practices and looked at the Defense Department s experience with COTS. On the basis of FAA and commercial experience, we developed a model of COTS refresh and maintenance costs as they increase through time. We coded our findings into a Microsoft Excel interface for use by FAA cost and budget analysts.

Model installation instructions are provided below in this chapter. Chapter 2 tells how to operate the model and describes the process flow of an analysis. Chapter 3 is a page-by-page discussion of the model spreadsheets. Chapter 4 addresses model expansion options. Chapter 5 describes the data sources.

MODEL INSTALLATION

System requirements: Windows 95 with Microsoft Excel 97 or better. 1MB (i.e., 1,000 KB) free space on your hard drive. Macros must be enabled when you load and use the model.

The Technology Refreshment Cost Estimating and Planning Model is originally packaged on a CD-ROM. It can be run from the CD if you don t want to make any permanent changes to the model. If you experience difficulty with the model running on CD-ROM (e.g., macros stop running and ask for debugging), you may have to install the model on your computer s hard drive.

To install the model on your computer, simply copy the file TechRef.xls to your preferred data directory on your computer. Open Excel, then use file—open to find TechRef.xls in your data directory. Select the filename and click on OK to

open the model. A warning screen may come up indicating that viruses can be stored in macros. The model as delivered on CD is free from viruses. Select Enable macros to continue.

Once the model opens, turn to Chapter 2 for operational instructions.

Chapter 2
Basic Operation

HOW TO OPERATE THE MODEL

The model was constructed to serve two functions: estimating time-sensitive COTS tech refresh and maintenance costs, and analyzing budget alternatives. The same parameters and functions are used in each, but since analysts using one function rather than the other will want specific data output, some elements of the model will be of greater interest than others, depending on the user.

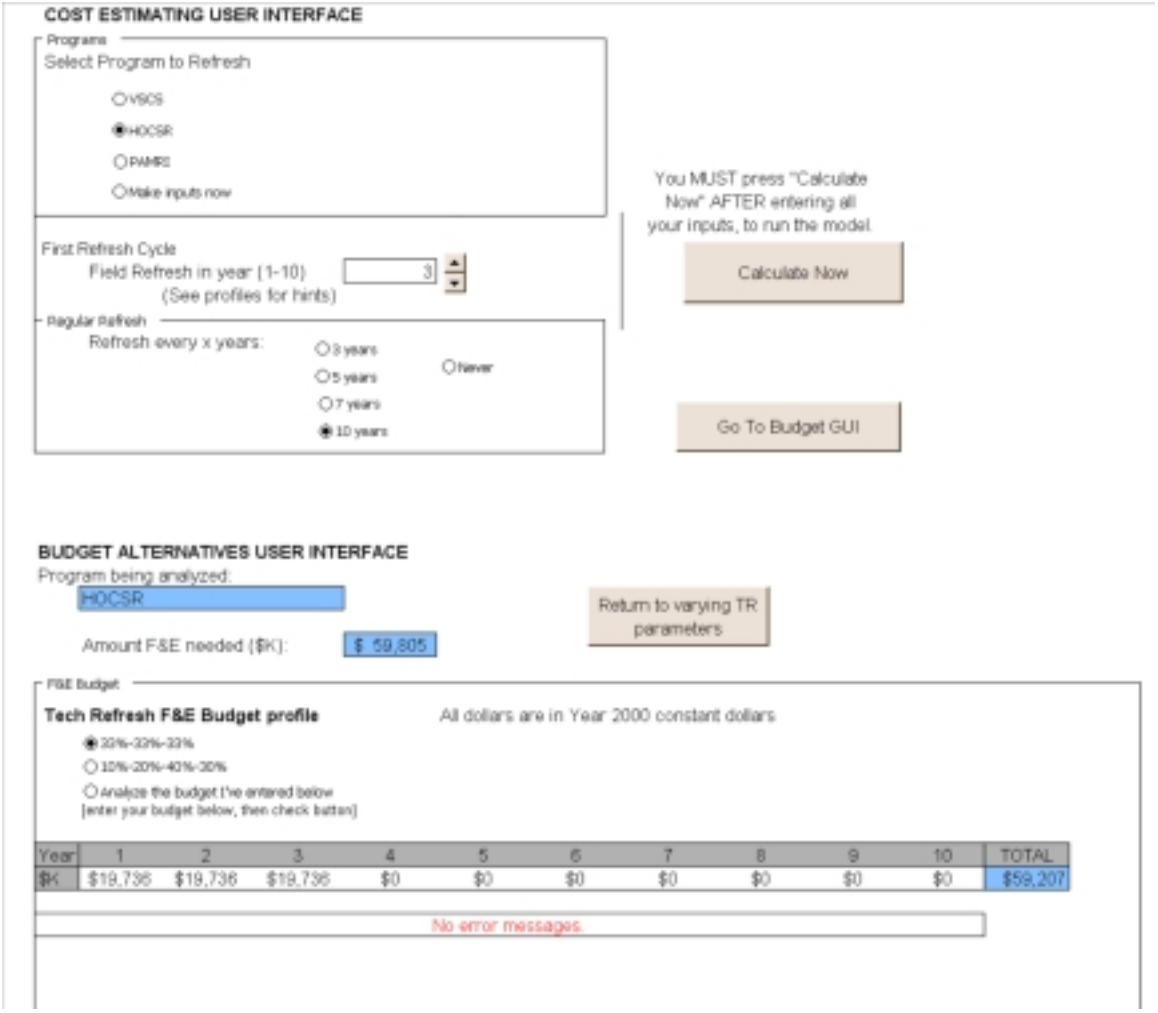
Figure 2-1 pictures the user interface; its use is described below. The full-page flow diagram (Figure 2-2) illustrates user inputs to operate the model.

Step-by-Step Guide

To begin using the model,

1. Turn to the User Interface worksheet, marked GUI on its tab. (See Figure 2-1.)
2. **Select a program to analyze**. Current choices are
   - VSCS,
   - HOCSR,
   - PAMRI, and
   - user-defined.

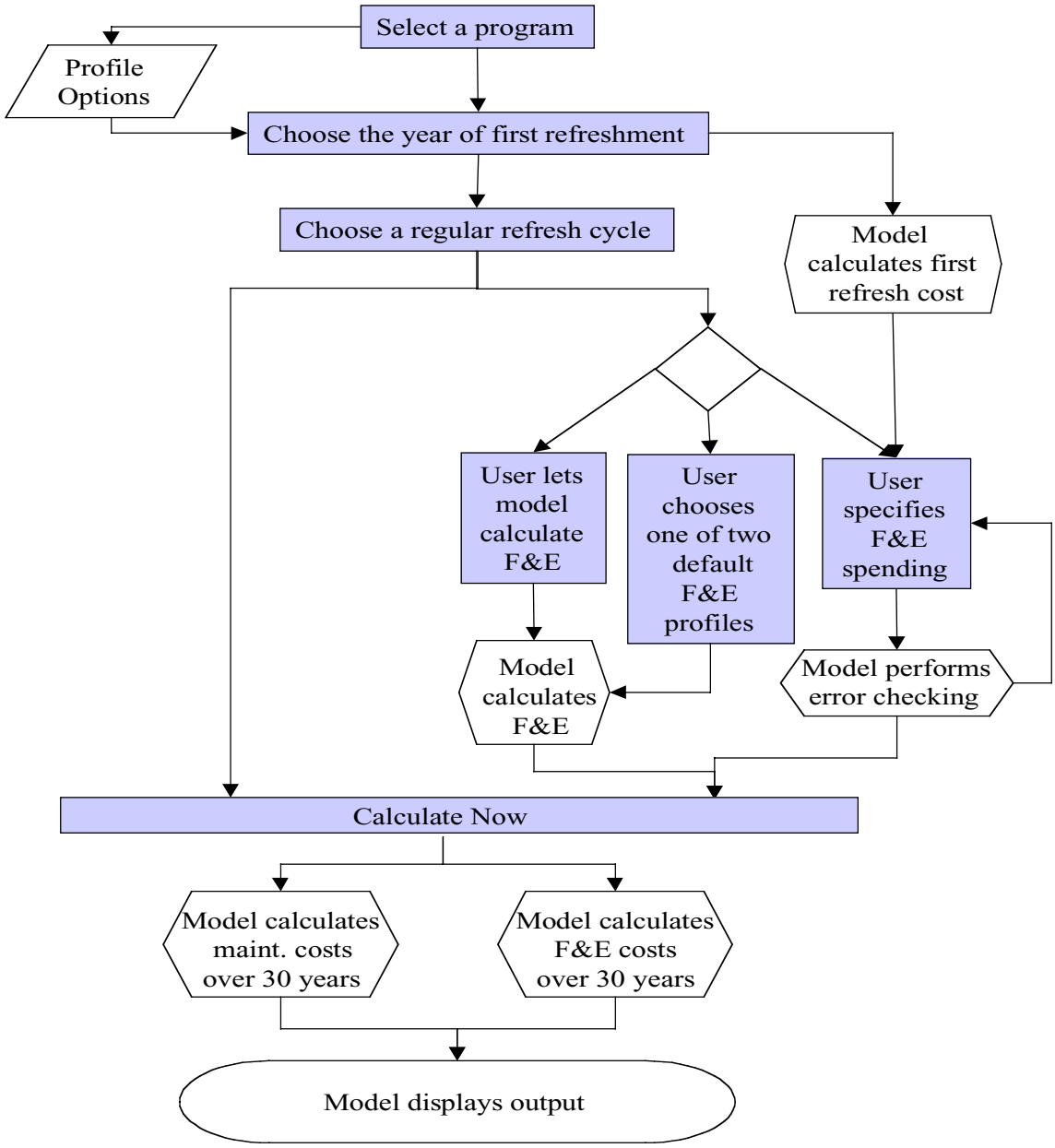
Figure 2-1. The GUI Worksheet



Select one of the pre-defined options by clicking on the radio button to the left of that program s text. If you are familiar with the model and wish to define a new program, please go to Chapter 4.

3. Use the spin button to **select a year to begin first tech refresh** for this program. This field determines the year in which the first refreshment will take place, relative to the current year.

Figure 2-2. User Flow Chart



A profile option chart can be accessed by clicking on the tab labeled Profile Option Chart. The profile chart illustrates combined default maintenance and acquisition expenditures associated with tech refresh for the program you have selected. Viewing the relative magnitudes of maintenance and acquisition expenditures on this chart may help the user choose a start year. The axis labeled Refresh in Year indicates a choice of first refresh year (i.e., refreshing a system when it is 1 year old, 2 years old, 3 years old, on up to 13 years). The axis labeled Spending merely lays out the years until the first refresh. Maintenance expenditures

grow over the years until first refresh. After first refresh, the chart is set to not depict any expenditures, though in fact maintenance costs will continue to be present. The final axis grades the magnitude of expenditures.

4. **Select a regular refreshment schedule** by pressing a radio button. This choice refers to how often you would like to plan to refresh your program after the first refresh. For example, you may schedule first refresh 5 years from now but wish to follow a regular 10-year cycle after that in that case you should select 5 in answer to step 3, and 10 in answer to step 4. Options supported by the model for regular refreshment are

   - every 3 years,
   - every 5 years,
   - every 7 years,
   - every 10 years, and
   - never.

   Your processor will process some short commands after you select any of these buttons. This is normal. Do not try to halt the processing. On most machines it will take a few milliseconds.

   The Never Refresh option will zero out the regular refresh cycle and the facilities and equipment (F&E)[1] budget; any inputs previously entered in those spaces will be overwritten with zeros.

5. If you would like to have the model automatically calculate an F&E budget for you, you can now go to step 9. If you would like to enter your own F&E expenditures, continue to step 6. If you chose Never Refresh, go to step 9.

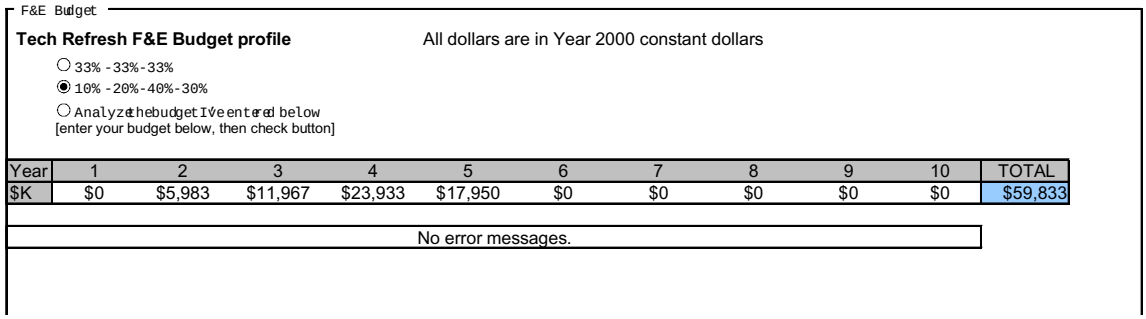
6. Your next choice is the F&E Budget profile. Figure 2-3 is an example. You can use the model s default budgets, which divide up the required F&E into 3- or 4-year spending profiles, or you can define your own F&E budget. The blue window above your profile tells you the amount of F&E you must spend to meet the model s projections for first refresh F&E. Note that the amount of money required to refresh your program increases as the year of first refresh moves farther into the future.

   To use the model defaults, click on a radio button next to the profile you prefer. At this time, the model supports two pre-defined budgets (33%—33%—33% and 10%—20%—40%—30%). Alternatively, to enter your own budget, click in the white boxes and begin entering numbers as you

[1] Facilities and equipment, or F&E, is used to refer to program acquisition costs.

would normally enter them in an Excel spreadsheet. Enter your budget in thousands ($K) of constant 2000 dollars. Each box should contain one year s worth of F&E funding. You can press Enter after each entry, but it is not necessary to do so. Use your cursor to advance to the next box if needed.

Figure 2-3. An Example of an F&E Budget Profile



The blue TOTAL box at the end of the budget line will automatically sum up your entries. You must enter an amount of funding equal to (or greater than) the displayed F&E requirement to avoid an error message.

You can enter expenditures in any year you like. The model will display an error message if your funding stream displays unusual characteristics, such as funding refresh after it occurs or expending funding more than two years before fielding. The error messages will appear in the text box below the funding entry boxes (See the No error messages box in Figure 2-3.)

7. If you receive an insufficient-funding message, you may re-enter your funding profile.

   Even if your funding stream causes an error message, the model will consider your inputs as the F&E stream for its calculations. See the section on Budgets in this chapter for further details.

8. When you are satisfied with your funding profile, click on the radio button associated with a user-defined budget profile. You may re-enter profiles and press this button as many times as necessary. Just be sure to press the user-defined (Analyze the budget I ve entered below) button after your final budget entry.

9. Page up and press the Calculate Now button.

Viewing Output

To view the output of your model run, click on the Output Chart tab. You will see a bar graph showing F&E and maintenance expenditures over a 30-year planning horizon for this program. If you would like to manipulate the output of this

model further, activate the show me the data button to access numerical output. This section can be copied and pasted onto another worksheet or document. Note that all numbers displayed on this sheet are constant 2000 dollars and that F&E is not in thousands as it is on the GUI sheet.

You can return to the graph by pressing the return to graph bu tton.

This graph is in a 3D format. To reverse the order of the bars displayed, press the switch rows in chart button, bringing the rearmost data series to the front of the chart.

If the chart seems to fail to reflect your tech refresh choices, return to the GUI tab, re-select your preferred options, and press the Calculate Now button. If the chart still fails to reflect your choices, see the section on debugging, below.

MODEL PURPOSE AND LIMITS

It is important that users understand the model s limits. This model is one tool that may be useful in certain contexts, and it should not be mistaken for something more than it is.

What the Model Does

The model estimates the life-cycle costs of certain COTS hardware and related operating system software. Its main usefulness derives from its ability to forecast relative costs for different refresh cycles and to help analysts plan and forecast the expenditures related to COTS refresh cycles.

The model takes a predefined set of COTS hardware and estimates

- its initial acquisition cost;
- its market life;
- its service life;
- its physical life;
- the cost of maintaining the hardware and operating system in each stage of those life periods;
- the cost of refreshing the hardware in any given year, at any stage of its lifecycle; and
- the cost of performing a customized rebuild for a failing subcomponent of the hardware, also known as a Service Life Extension Program, or SLEP.

Using these parameters, the model will tell you how much it will cost to refresh and maintain a given set of COTS hardware over the next 30 years. The model allows the user to vary refresh cycles and view their effects on costs. In addition, the model allows the budget analyst to do what if scenarios and project the impact of delayed F&E funding.

Underlying Algorithm

This section describes how the model relates hardware age to cost. The model deals primarily with computing and telecommunication equipment, such as computer processors, track balls, monitors, memory stacks, local area networks (LANs), switches, hubs, and routers. The model applies derived rules of thumb to classes of computers and telecommunication equipment to forecast their useful lives. In our model, hardware goes through distinct phases as it ages. The age of a piece of hardware is a determinant of its maintainability. The following definitions and examples will illustrate those phases. The examples should not be taken too literally; at this point the model is rather general in nature.

- Hardware—a physical component such as a computer, (e.g., a 386 computer), or a fiber optic LAN, or a TCP/IP dual-stack 4-port router.
- Operating system—the basic instruction set or set of protocols that runs the hardware (e.g., OS/2, MS-DOS IEEE 802.3, or IP version 4).
- Application—any software running on top of the operating system, often proprietary or unique, such as Windows 95, radar display update system, flight data.

Table 2-1 illustrates some examples.

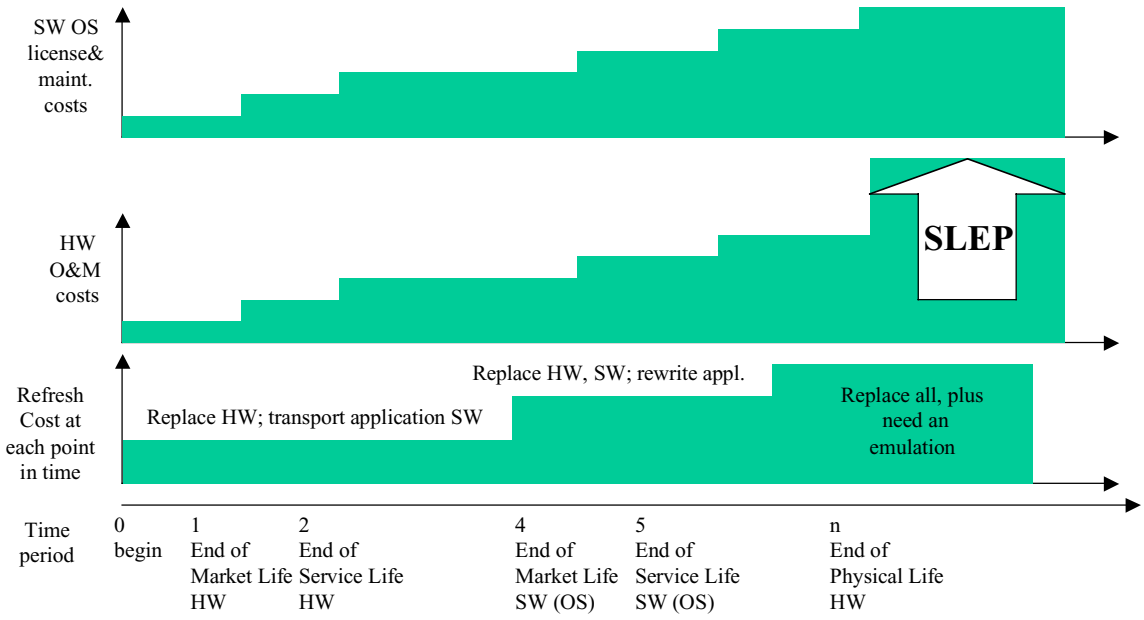
Table 2-1. Examples of Hardware, Operating Systems, and Applications

	Example 1	Example 2	Example 3
Hardware	386 computer	Fiber optic LAN	TCP/IP dual-stack 4-port router
Operating system	MS-DOS	IEEE 802.3	IP v.4
Application	Windows 3.0	Radar display update system	Encoded information and data

A piece of hardware goes through several periods in its life. The length of each varies by hardware type and is driven by commercial product cycles rather than by reliability. Figure 2-4 depicts these phases. When a piece of hardware (HW) is first introduced, it enjoys full product support from its manufacturer until its replacement is introduced. During the period in which hardware version x is the state of the art, it is generally one of only two generations of that hardware available for sale as new equipment. The period in which the hardware is offered

for sale by the original equipment manufacturer is the market life of the equipment. Market life ends when hardware x is no longer for sale from the manufacturer. The end of market life corresponds to the introduction of hardware version x+2. The end of market life also corresponds to the end of free manufacturer service; however, service can often be obtained from contractors.

Figure 2-4. How Maintenance, Licensing, and Refresh Costs Increase over Time for COTS Computer Equipment



There comes a time when maintenance cannot be obtained for the hardware on the open market. This is the end of the service life. From this point on, maintenance on the hardware can be obtained only from previously arranged long-term contracts and internal staff. Generally the end of service life occurs at a point twice the age of end of market life. Contracts for maintaining equipment after the service life has ended are exorbitantly expensive, if available at all. There is no guarantee that service is available at all during this period.

This period of exorbitant support is ended by the end of physical life. The end of physical life is generally reached when one subcomponent of the hardware begins failing en masse. An example is the power supply of a particular processor. For that generation of hardware, a particular subcomponent is the weak link and fails in most or all equipment. The owner of the hardware must backward engineer that subcomponent in order to continue to maintain the hardware. This backward engineering, part of a SLEP, extends the life of the hardware for a few more years but is very expensive.

The idea behind the algorithm is illustrated in a general way in Figure 2-4. Note that Figure 2-4 encompasses the possibility that support providers continuously raise prices through the life of the hardware and operating system.

The operating system of the hardware, referred to as software (SW) in the model, follows a similar lifecycle, differing only in the lengths of time and the pricing of maintenance and acquisition. The software generally is supported longer, since a relatively mature operating system does not need replacement parts. In software maintenance we also include any licensing fees. Annual l i-cense fees are often charged on operating systems, compilers, and proprietary middleware. The license fee is merely the annual rent for keeping the software operating. Software with a license fee usually includes a license-checking feature and disables itself if proper codes from the manufacturer are not entered before the software s due date.

Software can also reach the end of its physical life, when it cannot be supported any longer. A number of factors can trigger this: support for the operating system cannot be obtained for any amount of money; the legal owner of the operating system refuses to license the operating system further; the software was buggy to begin with and cumulative bug fixes have made it less reliable or too complicated to modify further; or the application s demands outgrow the operating system s operating parameters.

A combined hardware and software system reaches the end of its operating life when either the hardware or the software reaches the end of its physical life.

These life periods and their associated cost magnitudes have been determined for a finite set of computer and telecommunications equipment and programmed into the model as parameters. Costs and duration of the periods vary by equipment: that is, since a personal computer and a fiber optic cable have different life spans, parameters have to be defined individually for each class of hardware.

The model takes a defined set of input hardware and forecasts the maintenance expense of each component in each period of its life. Then, using the analyst s inputs on refreshment, the model projects maintenance costs until and after re-freshment. The model also predicts refreshment costs on the basis of the period-age of the equipment at the time of refreshment.

EXAMPLE

A new multi-processor computer costs approximately $35,000; its operating sys-tem, $1,000. The hardware is on the market for 3 years and the operating system is on the market for 5 years. During market life, hardware and software support contracts cost approximately $1,000 per year.[2] After the end of its market life, bridge contracts are entered into to provide maintenance and support for the hardware and operating system; they increase in price at the rate of 20 percent per

[2] Generally a manufacturer will offer free limited support during market life. However, to se-cure maintainability after the end of market life, or to receive 24 hour per day, 7 days per week support, a long-term support contract must be procured up front; thus an agency is usually unable to take advantage of the free support period.

year. After 5 years, the processor reaches its end of service life. After 10 years, the hardware and software together are no longer supportable.

To refresh this system before the end of its (software) market life, an agency would need only to procure new hardware and transport the application software. After the end of the software market life, new hardware will not run the old operating system. New software must be procured and a gap-filling program written to ease the transport of the application software to the next-generation operating system. After the end of physical life, generally the operating system has evolved substantially and cannot run the application at all, so new hardware and software must be accompanied by re-writing the application software or writing an emulation environment. These increasing requirements on technical refreshment are reflected in increasing costs to refresh as the COTS systems age.

These parameters do not hold for every piece of hardware, or even for all processors. Some bridge contracts begin at the end of market life, some at the end of service life. Some bridge contracts end at the end of physical life; others can be continued indefinitely. Every piece of hardware must be considered separately to determine its lifetime. An undisturbed LAN is projected to have a 30-year service life and a 60-year physical life [see Reference 102] and to need only minimal maintenance throughout, though user demands will probably render the LAN obsolete before it is physically inoperable. Some hardware does not have an operating system, simplifying the analysis of its maintenance costs and its replacement.

Additional Estimation

The remaining costs of a procurement, including program management, logistics, testing, training, spares, deployment, etc., are estimated as a percentage of Prime Mission Equipment.

Assumptions

The following assumptions apply:

1. The FAA will always procure maintenance agreements on COTS hardware, even if the manufacturer supplies free support for the first market years.

2. Contractor and FAA maintenance have comparable costs.

3. If the FAA chooses to not refresh a COTS system, the FAA will follow the path of lowest cost as the system reaches obsolescence; that is, it will choose to run a SLEP rather than face system outages, if at all possible.

4. Stretching the F&E funding of a tech refresh over many years does not in itself increase program costs, and there is no minimum program level in any funding year.[3]

5. A tech refresh is not considered fielded until 90 percent of its required F&E has been expended.

6. The FAA will procure rather than lease hardware. F&E funding always precedes fielding.

7. F&E funding always falls within 2 years of fielding

8. For simplicity, all fielding takes place in a single year, the year in which F&E funding equals 90 percent of the required tech refresh budget.

9. For maintenance costing purposes, it is assumed that a manufacturer will never refuse to license software but instead will continue to charge ever-escalating license fees.

10. In order to project a hypothetical 20-year cost profile for the Never Refresh case, it is assumed that a system will undergo SLEP every 3 to 5˚years (depending on the hardware) after its end of physical life.

Budget Analysis

User-Inputted Budgets and Error Checking

The model performs error checking on a user-inputted budget. The model notifies the user if the inputted budget does not conform to the model s requirements for funding. The model checks for

- F&E less than required refresh costs,
- A majority of F&E expended more than 2 years in advance of first refresh, and
- Less than 90 percent of required F&E expended before first refresh.

However, the model will allow most erroneous budget inputs to be used as the preferred first refresh budget profile. This was done in case the user wishes to input a nonstandard budget for a particular reason not encompassed in the planning of the model.

[3] We realize that many products require a critical mass, a period in which funding is sufficient to support staffing of several specialized tasks simultaneously; for instance, funding a logistician and system developer while the program is still being coded and built. Also, that keeping a project office open throughout the project is necessary, so that funding should never drop to zero but to some maintenance level. These requirements are not reflected in the model at this time.

The exception is insufficient funding. The model will not schedule a tech refresh and maintenance costs will not decrease over the first 10 years unless sufficient refresh money is included in the F&E budget. The model was programmed with this feature to show how immediate cuts in the F&E budget will cause increases in maintenance costs. Note that a 3 percent error margin is built into the comparison feature so that rounding errors will not cause the budget to return an error of insufficient funding.

The following examples (Figure 2-5 through 2-12) will illustrate the model s capabilities:

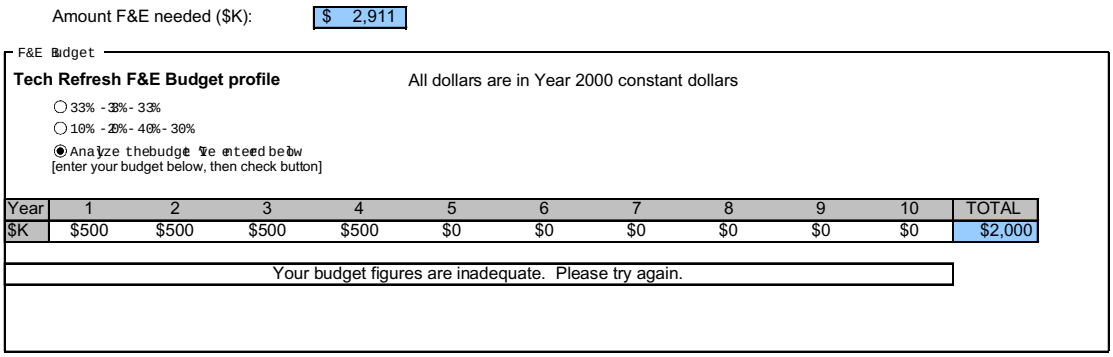
EXAMPLE 1

First Refresh: year 5

Required F&E: $2,911 K

You enter $2,000 K.

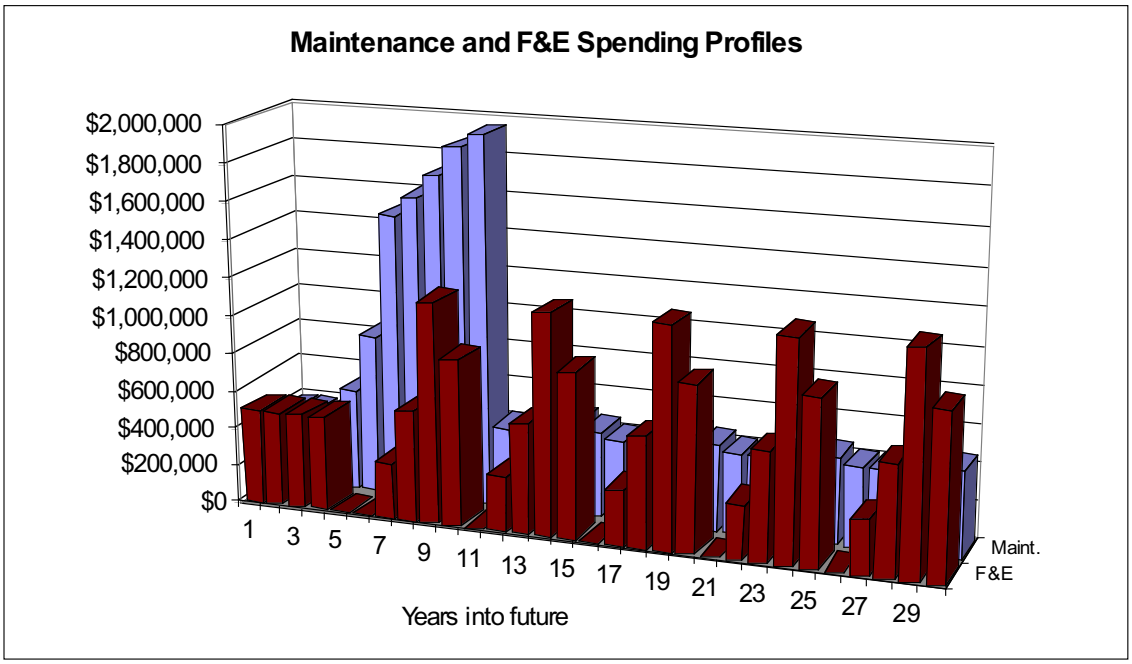
Figure 2-5. Example 1 Budget Profile



Your error: insufficient F&E funding to meet tech refresh requirements.

Model output: The model uses the (insufficient) user-input budget for the first refresh cycle; but refresh is never achieved because it is not fully funded. However, the repeat refresh cycles are automatically budgeted for and the model executes refresh cycles every 5 years, with automatic 4-year spending profiles.

Figure 2-6. Spending Streams Resulting from Example 1 Inputs



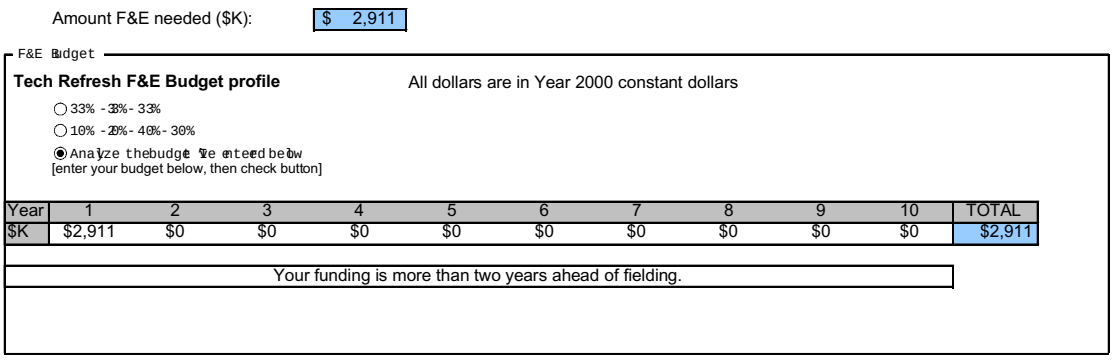
EXAMPLE 2

First Refresh: year 5

Required F&E: $2,911 K

You enter all funding in the first year.

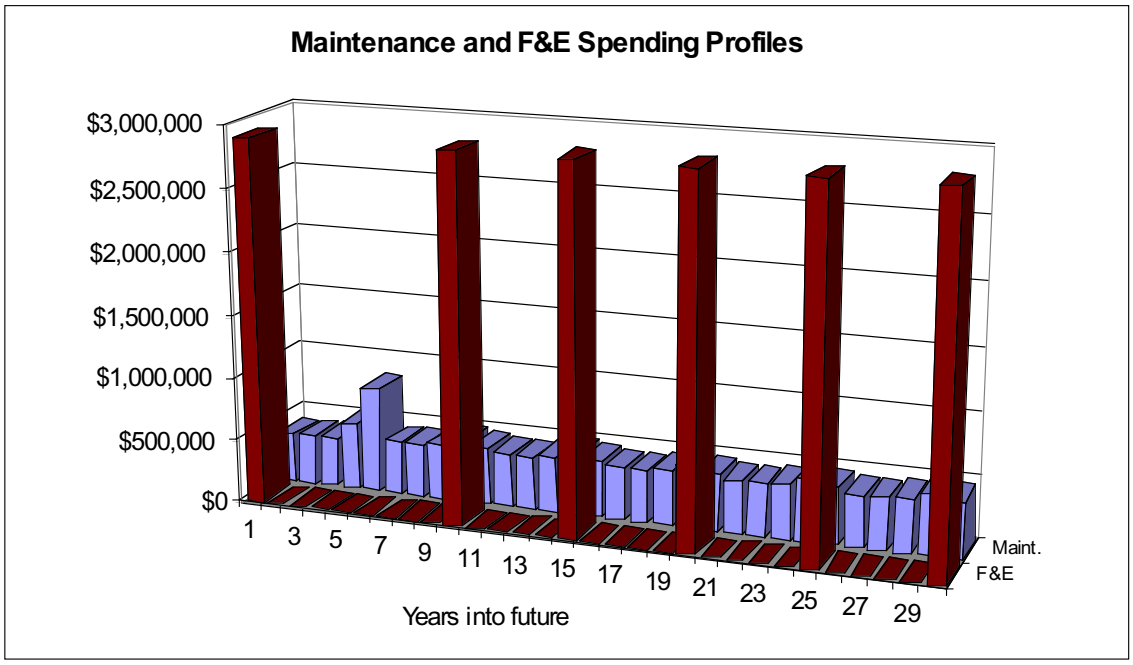
Figure 2-7. Example 2 Budget Profile



Your error: funding is considerably ahead of fielding.

Model output: Since your funding was sufficient, the model uses your funding profile for the first refresh. However, for regular refresh, the model moves your funding to the fielding year.

Figure 2-8. Spending Streams Resulting from Example 2 Inputs



If you want to move funding to a later year, you may do so and re-run the model, which was done in Example 3, as depicted in Figure 2-9 and Figure 2-10.

EXAMPLE 3

First Refresh: year 5

Required F&E: $2,911 K

You enter all funding in the fifth year.

Figure 2-9. Example 3 Budget Profile

Amount F&E needed ($K): $ 2,911

F&E Budget

Tech Refresh F&E Budget profile

All dollars are in Year 2000 constant dollars

○ 33% - 33% - 33%

○ 10% - 20% - 40% - 30%

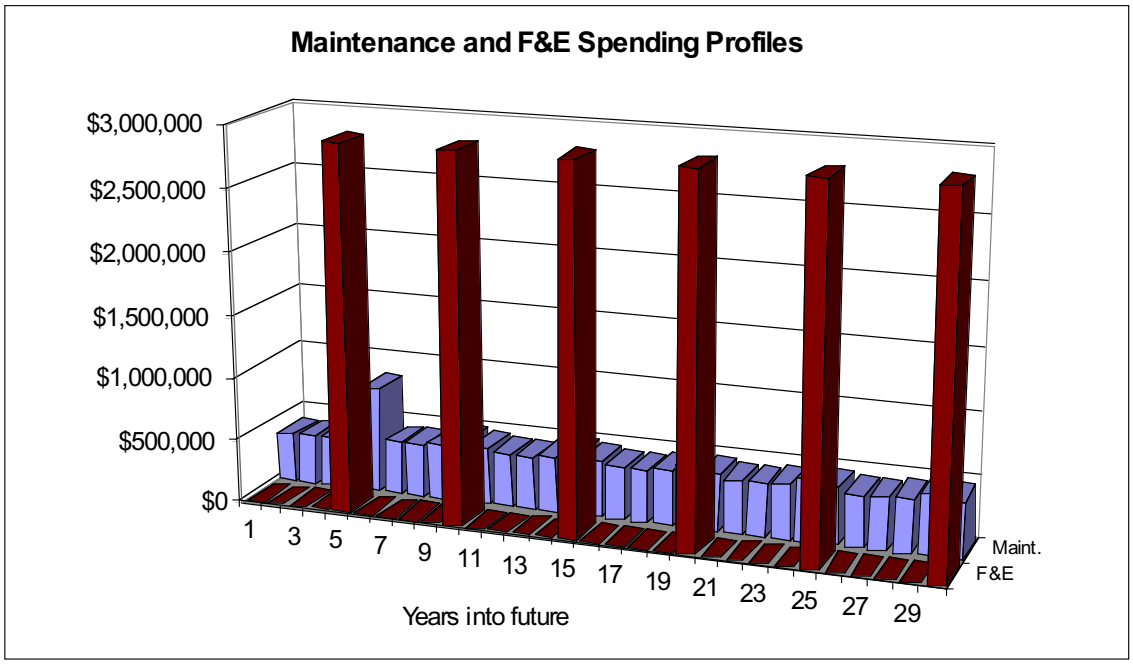
◉ Analyze the budget I've entered below
[enter your budget below, then check button]

Year	1	2	3	4	5	6	7	8	9	10	TOTAL
$K	$0	$0	$0	$0	$2,911	$0	$0	$0	$0	$0	$2,911

No error messages.

There are no errors. The model considers this an optimum spending profile and uses it again for future tech refresh cycles, as can be seen in Figure 2-10.

Figure 2-10. Spending Streams Resulting from Example 3 Budget Inputs



EXAMPLE 4

First Refresh: year 5

Required F&E: $2,911 K

You enter an actual funding profile that has been stopped and re-started.

Figure 2-11. Example 4 Budget Profile

Amount F&E needed ($K): $ 2,911

F&E Budget

Tech Refresh F&E Budget profile

All dollars are in Year 2000 constant dollars

○ 33% - 33% - 33%
○ 10% - 20% - 40% - 30%
◉ Analyze the budget profile entered below
[enter your budget below, then check button]

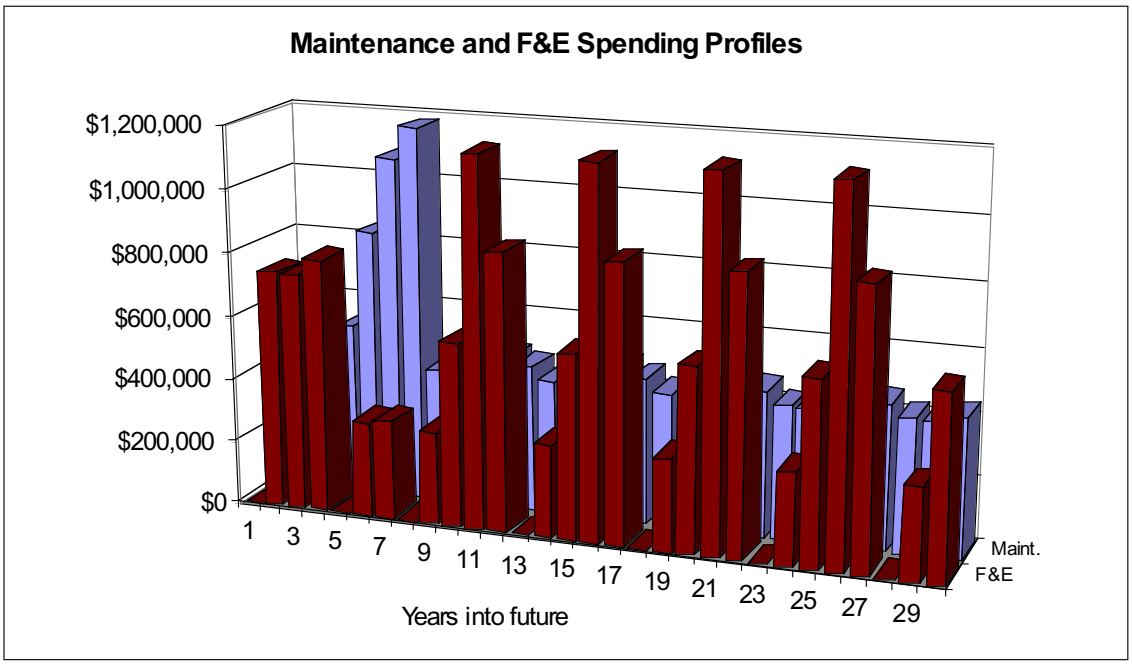
Year	1	2	3	4	5	6	7	8	9	10	TOTAL
$K	$0	$750	$750	$800	$0	$300	$311	$0	$0	$0	$2,911

No error messages.

Result: Your funding profile is sufficient, so it is used. The 90 percent benchmark for fielding refresh is reached in year 7, so the model fields refresh in year 7, even though the user had indicated first refresh in year 5. (The budget sufficiency checking overrides all else.) Because your budget profile stops and starts and ex-

ceeds your planned refresh date, the model generates an automatic budget profile for the regular refreshment years.

Figure 2-12. Spending Streams Resulting from Example 4 Inputs



Important Limitations of the Model

The model does not yet estimate application software acquisition, testing, or development, which may easily constitute the majority of costs of any procurement project.

The model does not have any utility for the amount of developmental items a procurement will need to field a particular system. These costs are not modeled.

The model is not capable of independently producing a cost estimate of any FAA program for which it has not been programmed. The FAA programs that the model is capable of addressing are listed with radio buttons on the GUI sheet.

This model is rule-based and as such represents a coarse level of measurement. It cannot help a procurement officer differentiate between competing COTS offerors.

This model does not currently support a leasing scenario, or power by the hour payments. Future versions could be re-programmed to include a leasing option under F&E options. It is possible to *emulate* a leasing option by entering an F&E budget profile of equally divided payments. For example, if the refresh cycle is 5˚years and the F&E cost is $100,000 K, enter 5 years of funding of $20,000 K each, and run the model as usual.

TROUBLESHOOTING

Macros Stopping

Most of the computation problems in the model will be caused by incomplete transference of data from one location to another. The first step when any anomalous outcomes are spotted is to return to the GUI sheet and re-activate Calculate Now. If needed, re-click your desired inputs and click on Calculate Now again.

If the macro continues to fail to run, re-install the model from the CD onto your hard drive. Before doing so, save any additions you have made to the model on a separate spreadsheet, so that you can copy them to the new model.

All the macros in the model refer to particular ranges and particular model sheets. If you have renamed any sheets or inserted rows, columns, or cells, the macros will function incorrectly. Re-installing the model will cure this problem.

It is possible to add sheets to the model when it is installed on your hard drive. As long as they do not have the same name as the original model s sheet tabs, they should not cause errors.

Illogical Results

If the user finds the results defying logic, the error is probably in the algorithm or input of key parameters. For example, if maintenance costs decrease as the system ages, it is likely that a value on the Parts worksheet is causing that effect. It may be the case that the bridge support growth rate was accidentally entered with a negative number, or that initial maintenance costs were overstated by an order of magnitude. To find the source of illogical results, first examine the maintenance cost streams on the Calc2 worksheet. It should be possible to identify one stream as causing the strange result. Follow that cost stream back to its source worksheet, e.g., HW maint, SW maint, or Refresh. Follow the reasoning of the algorithm on this worksheet to determine which year the anomaly begins to occur, then consult the Parts worksheet, looking for the parameters that describe that period.

Chapter 3

Page-by-Page Explanation of the Model

INTRODUCTION AND DEFINITIONS

By examining each worksheet and its associated macros in turn, this chapter explains in detail how the model works. It is not necessary (or even recommended) to read this chapter to operate the model. When reading this chapter, it is vitally important to have the model open on your computer at the same time. This chapter exists to assist in future modification of the model.

Some definitions are in order. In this report and in the model itself, automatic means that the cell under discussion is a function of or a variable of a function that ordinarily runs in regular Excel. The simplest example of this relation is:

In cell C1, the formula,

= A1 + B1

Here, we would say that cell C1 automatically updates from cells A1 and B1; or that cells A1 and B1 automatically update C1. No macro needs to run for C1 to change its value.

However, if cell A1 is a value pasted by a macro, then we would say that A1 is updated by a macro. For an example of a macro pasting several cells that are then summed automatically, see the discussion of the worksheet Calc2. It is the interdependency of macro-assigned cells and automatic cells that makes running Calc Now a must after the user assigns parameters to the model.

Also, we commonly say that a cell is called by a macro, or that a macro is called by a button. When a macro is called, that means it is run. When a cell is called, that means that the cell holds a value that is referenced by a macro, usually to guide the macro as to what to do in a select case choice list.

GUI

Purpose of GUI Worksheet

The GUI worksheet is the model s user-interface center. It provides a single location for adjusting inputs for the most basic use of the model. Because of that function, it is also the sheet in which key parameters are selected. These key parameters remain important for advanced users of the model.

The GUI worksheet provides model interfaces for the two anticipated users of the model: budget analysts and cost analysts. The cost analysts interface can be considered the first 20 rows of the worksheet; the budget analysts workspace comprises row 24 through 43.

Each user interface consists of several parameters to be set, most of which activate short macros that pre-set the model. It would be possible to centralize all these macros to run all at once, if one were inclined to do so; however, this was considered inefficient at the time the model was initially developed. Additionally, originally the GUI was designed to be an Excel Visual Basic UserForm, but bugs were discovered in Excel that caused incorrect operation of program buttons, as well as inability to link spinners to text boxes, so the UserForm was discarded.

FAA Program Options

The cost estimators section of the model includes selection of a pre-defined FAA program, selection of the first refresh year, and selection of a regular refresh cycle. The pre-defined FAA programs are VSCS, HOCSR, and PAMRI. These programs, along with Make inputs now are predefined within a Group Box labeled Programs, which makes selection of buttons mutually exclusive. Only one program can be run at a time. Selecting the VSCS button calls the VSCS_prgm_data macro. Selecting the HOCSR button calls the HOCSR_pgm_data macro. Selecting the PAMRI button calls the PAMRI_pgm_data macro. Each of these macros performs the same function: it goes to the FAA Systems worksheet, copies the row of data associated with that program, and pastes it onto the Calc1 worksheet. The macro clears the copied data from its clipboard, writes the name of the system being analyzed in the budget analysts section of the GUI, returns to GUI cell A5, and quits.

Selecting the Make inputs now button calls the clear_old_prgmdata macro. This macro writes user defined in the budget analysts GUI, erases old pr o-gram data in Calc1, and positions itself to record program inputs on the FAA Systems worksheet. It also leaves a single 1 in one of the Calc1 rows, preventing the entire workbook from being corrupted by #DIV/0! errors while new program data are being entered.

First Refresh

First Refresh Cycle is operated from a spinner box (see Figure 3-1) with associated text box.

Figure 3-1. Spinner Box

Operating the spinner buttons causes the spinner s value to be increased or decreased; the text box merely displays the value of the spinner. The spinner does not activate any macros, but cell A101 on the Calc2 worksheet is automatically updated from this text box, and the value in cell A101 is a key parameter called by the model s main engine macros. Additionally, this text box (GUI:F12) is automatically used in over 30 of the budget-setting and budget-checking functions on the Budget spreadsheet.

Regular Refresh

The regular refresh cycle options 3 years, 5 years, 7 years, 10 years, and Never —are mutually exclusive button-operated choices within the Group Box labeled Regular Refresh. The 3 years button calls the Define_cycles_3 macro, the 5 years button calls the Define_cycles_5 macro, the 7 years button calls the Define_cycles_7 macro, the 10 years button calls the Define_cycles_10 macro, and the Never button calls the Never macro. The first four macros, generically called the Define_cycles_n macros, operate in the same way: they basically set up a few variables for operating the model. All four paste the option number selected (either 3, 5, 7, or 10) into cell A102 on the Calc2 worksheet, then copy the a refresh cycle budget stream from either row 84, 85, 86, or 87 to row 106 on the Calc2 worksheet. Calc2 cell A102 is a key parameter called by main engine macros. The refresh budgets on rows 84, 85, 86, and 87, like much of Calc2, were generated automatically once the FAA program was selected.

The Never macro forces the first refresh spinner to zero, then places zeros in the user-defined budget (GUI cells C38-L38). It calls the paste_input_budget macro, which pastes the zero budget into the budget workspace on the Budget worksheet. Finally, it pastes 100 into the define cycles variable, found on the Calc2 worksheet, cell A102. This cell is pasted only by macros and is used to set the repeat refresh cycle. The value of 100 in this variable literally tells the model to follow a 100-year regular refresh cycle, well beyond the planning horizon, and so the model never schedules regular refreshment.

Buttons

The Go To Budget GUI button activates the gotobudgetGUI macro, which merely tries to re-position the active viewing screen so that budget choices are available.

We ll discuss the Calculate Now button at the end of the section on the GUI worksheet, below.

Budget GUI

The Budget Alternatives User Interface first displays the current program under analysis and the model s calculated tech refresh cost for the first refreshment of that program. The name of the program was pasted by an FAA program selection button, as discussed above. The tech refresh cost is automatically calculated on sheets Calc1, Calc2, Refresh, and Budget once the program and its first refresh year are known. The cost of the refresh is calculated by the model and cannot be changed by the user. Budget funding can be changed by the user.

The budget analysis consists of 3 choices, arranged in a Group Box labeled F&E Budget. The user can select a model-generated funding profile over 3˚years or one over 4 years, or can generate a new funding profile. The 33%—33%—33% button calls the userbudsel1 macro; the 10%—20%—40%—30% button calls the userbudsel2 macro; the Analyze the budget I ve entered below button calls the userbudsel3 macro. Each macro sets a variable, userbudvar, which is stored in cell D1 on the Budget worksheet. The userbudvar key vari able later tells the main engine macros which budget macro to call: Default_budget1, Default_budget2, or Paste_input_budget.

Cell B40 on the GUI worksheet automatically compares the sum of cells C38 through L38 with the requirement posted in cell F28 and generates an inadequate funding message if the user-defined budget does not sum to at least 97 percent of the model s calculated requirement. This cell also references cells O26 and O29 on the Budget worksheet and notifies the user if the user has entered a partic u-larly odd funding stream or has scheduled fielding before more than 50 percent of the funding is spent. If the user has selected an automatic budget, there will be no error messages.

The Return to varying TR parameters button activates the return2CEGUI macro, which merely re-positions the active viewscreen to the upper half of the GUI worksheet.

Calculate Now

The Calculate Now button activates the Horizons macro, which in turn calls many other macros that constitute the model s main engine. Horizons calls choose_ budget(userbudvar) and then sets variables trcycle (regular refresh

cycle years); budyrs (the number of years in which there is F&E funding); funnyfunding (indicates a stop and re-start in F&E); NSF (indicates that insuffi - cient F&E funding has been entered by the user); styear1 (year in which user wishes to do first refresh); and styear2 (year in which funding is sufficient to do first refresh). Horizons then calls the function choose(styear1, styear2, styear) and the macros select_budget_horizon, start_cycles_right and output. Each of these macros is discussed in the Modules and Macros section, below. S e- lect_budget_horizon and start_cycles_right are the most complex macros.

PROFILE OPTION CHART

This chart is automatically generated upon selection of the relevant FAA program, before any user inputs into first refresh year and refresh cycle. It shows the combined maintenance costs and tech refresh cost for the relevant FAA program over a 13-year horizon. It is drawn from data shown on Calc2 in cells E68 to R80, generated from the Calc1, HW maint, HW maint, and Refresh wor k- sheets.

OUTPUT CHART

The Output Chart is the final answer to all of the analysts inputs, showing first refresh costs, regular refresh cycles, and maintenance costs over a 30-year time horizon. It is automatically generated from Calc2, cells E110 through AI110. The values of cells E110 through AI110 are the sums of numbers in rows 104˚through 109. Cells in row 104 through 109 are pasted by macros. The chart automatically resizes its Y axis (dollar value) to fit the magnitude of dollars displayed, so an eye should be kept on the values listed when doing comparative analysis.

The display of maintenance or F&E can be pushed to the front for better viewing by clicking on the switch rows in chart button, which activates the switch_data_series macro. The show me the data button activates the show_data macro, which repositions the viewscreen to the numerical output corresponding to the table. The return_to_graph button activates the r e- turn_to_graph macro, which re-positions the viewing screen to view the graphi - cal display once again.

CALC1

The Calc1 worksheet is a calculation worksheet that takes the first step in transforming generic maintenance requirements for each component into an FAA system requirement. The next steps are taken in Calc2.

The Calc1 worksheet lists all the COTS components currently programmed into the model. Column D lists the number of each component used by the system currently being analyzed by the model. Column D is pasted by the

VSCS_prgm_data, HOCSR_pgm_data, or PAMRI_pgm_data macro when it is invoked by program selection on the GUI worksheet.

In the columns to the right of the component counts, Calc1 multiplies component count by its annual cost, for years 1 through 20, for the categories hardware maintenance, software maintenance, and refresh costs. Hardware maintenance for years 1 through 20 occupies columns F through Y; software maintenance for years 1 through 20 occupies columns AA through AT; the cost of refreshing in year 1 or 2 or 3, and so on through year 20, occupies columns AV through BO. The calculations are arranged side by side so that remaining rows can be used to expand the model s component database.

The cost of maintaining a single piece of hardware, of maintaining a single software license, or of refreshing a single hardware-software combination piece is calculated on the HW maint, SW maint, and Refresh worksheets, respectively. The reference from Calc1 back to single-component costs on the HW maint, SW maint, and Refresh worksheets is automatic.

The numbers calculated on Calc1 are automatically summed and displayed on Calc2.

CALC2

Calc2 uses the output of Calc1 to produce FAA system cost estimates.

At the top of Calc2, the cost of maintaining and refreshing the required number of each component in an FAA system is summed from Calc1. These sums are produced automatically and are shown over a 20-year time horizon. (See rows 1˚through 4.)

In rows 8 through 65, the costs shown in rows 1 through 4 are re-referenced and arrayed as the FAA would incur them if the FAA undertook first refresh in year 1 (rows 9—11), or first refresh in year 2 (rows 15-17), or first refresh in year 3 (rows 21-23), and so on. This is accomplished by referencing the age-specific maintenance cost for hardware and software under the year(s) before refreshment. In only the year of refreshment, the cost of refreshment is also listed. These cost profiles are labeled profiles in the spreadsheet. They are automatically generated from Calc2 rows 1 through 4. Careful reading of the other sections in this chapter reveals that rows 1 through 4 were automatically generated after selection of the FAA program; hence all sums in this section are correct without having invoked main engine macros.

The profiles generated in rows 8 through 65 are summed in rows 69 through 80 and are used to generate the Profile Option Chart as described above in the section Profile Option Chart. Rows 69 through 80 are generated automatically.

Rows 84 through 88 generate the maintenance costs over 30 years associated with a regular refresh cycle of 3, 5, 7, or 10 years. Maintenance costs tend to be lowest in the first 2 or 3 years after new installation or refreshment, then grow over time. Here maintenance costs recorded in rows 2 through 4 are reproduced starting from their lowest point, growing as the system ages, being reset to their low point in the year of refreshment, growing again, and so on. These flows are generated automatically.

Rows 104 through 109 contain complex formulas, interact with macros, produce the final maintenance cost numbers, and present the final budget streams over the next 30 years. Row 104 produces the sum of annual hardware and software maintenance costs until first refresh. In the first year, cell F104, maintenance cost is always incurred. Even if refresh is scheduled in year 1, in this model maintenance is assumed to be ongoing even during installation. In the rest of the row, cells G104 through AI104 show the maintenance costs from rows 2 and 3 until refresh is reached. This is accomplished with a MAX() function; the current year is compared to the user-inputted first refresh year (Calc2!A101), to the year of funding sufficiency (Budget!O31), and to a variable indicating that the Never Refresh option was chosen (neverRef variable, or Budget!O30). If any of these variables is greater than the current year, then the cell reproduces the annual maintenance cost. If all these variables are less than the current year, the cell sets itself to zero. For example, a user might have selected first refresh in year 5, but entered a budget profile in which required funding is delayed until year 6. Never Refresh in this case would be zero, since it was not selected. In cell G104, the function compares the current year, 2, to 5, 6, and zero and returns the maintenance cost. Cell H104 compares year 3 to the set 5, 6, and zero and returns the maintenance cost. This continues until year 7, when the year is greater than all three comparison variables, and so cell L104 returns a zero, as do cells M104 through AI104.

Row 105 produces the current budget stream for first refresh, drawn automatically from the working space on the Budget worksheet.

Row 106 depicts the repeating rising and falling maintenance costs associated with regular refresh every 3, 5, 7, or 10 years. This row is updated during the running of the main engine macros; one of rows 84 through 87 is pasted (paste special value only) by a Define_cycles_ *n* macro. This row s function is to temporarily hold the correct repeating maintenance cycle. During main engine running, Define_cycles_*n* pastes the repeating maintenance cycle here; then the model performs some calculations and sends another macro to copy the cycle to the correct spot in row 107.

Row 107 will hold the correct repeating maintenance cycle, starting *after* first refresh is reached. The purpose of storing this cost stream in row 106 and then copying it to row 107 is to enable the model to push the repeating maintenance cycle costs out so that they begin after the first refresh. The model pushes the

costs out during the copy process, using a set of macros called push*n*, where *n* is a number between zero and 10.

The model s main engine compares two variables, styear1 and styear2, selects the maximum value, and names it styear. Styear is shorthand for start year. Styear1 is the variable holding the year the user has designated for first refresh. Styear2 is set by functions on the Budget worksheet indicating the year in which first refresh has sufficient funding. If the budget is determined by the model, styear1 and styear2 will be the same. If the user enters a budget sufficient to field refresh in the year he has chosen, styear1 and styear2 will be the same. When styear1 and styear2 are unequal, styear is the greater of the two.

The value in styear is used in the macro Start_cycles_right(styear), a select-case macro that calls one of the push*n* macros corresponding to the proper start year. So, if styear is equal to 7, Start_cycles_right(styear) calls the macro push7, which pastes the refresh cycle from row 106 into row 107, and pastes it to begin in year 7.

The push*n* macros also clear row 107 of old values (from the last model run) before they paste new values. In the case of Never Refresh, styear is set to zero, which causes Start_cycles_right(styear) to do nothing and end. In the case of Never Refresh, row 107 is zeroed out by the pul13 macro, which is called by the select_budget_ horizon( ) macro.

Row 108 is a temporary holding place for the regular refresh F&E expenditures, much as row 106 is for maintenance expenditures. The F&E expenditure stream comes from one of 12 automatically generated regular refresh F&E streams. The 12 F&E streams are created in rows 25 through 48 on the Budget worksheet. The macro select_budget_horizon(budyrs, trcycle, funnyfunding, NSF, styear, budhor) chooses one of those 12. Since Select_budget_horizon is the most complicated macro in the model, we won t describe it here except to say that it selects the optimal F&E expenditure stream over the next 30 years and pastes that stream into row 108. Select_budget_horizon is discussed in detail in the Modules and Macros section, below.

Row 109 is the final repository for the correct F&E cost stream in the appropriate year. The budget stream stored in row 108 is pasted into row 109, starting in the correct year, using the same push*n* macros that paste maintenance costs from row 106 into row 107.

Row 110 is the automatic sum of the maintenance costs before first refresh, in row 104, and those after first refresh, in row 107. Row 110 contains the entire 30-year maintenance costs, which are then depicted on the Output Chart worksheet. Similarly, row 111 is the automatic sum of the first refresh F&E costs (from row 105) and the F&E cost stream for all subsequent refreshes (from row 109.) Row 111 provides the F&E figures seen on the Output Chart worksheet.

Cells A112 through A121 are repositories for macro variable values, principally for the Select_budget_horizon macro. While the model was being constructed, it was necessary to check the operation of the combined macros. This was accomplished by programming the macros to output key variable values to this range. The reporting and repositories are retained, since they will be useful when the model is expanded. Figure 3-2 pictures these repositories; referring to them will be useful in the discussion on macro operation.

Figure 3-2. Auditing Variables

5	trcycle	cell for auditing macros
5	budyrs	cell for auditing macros
0	NSF	cell for auditing macros
7	styear	cell for auditing macros
5	styear1	cell for auditing macros
7	styear2	cell for auditing macros
0	funnyfundi	cell for auditing macros
10	budhor	cell for auditing macros
7	push	cell for auditing macros
10	pull	cell for auditing macros

FAA SYSTEMS

This worksheet holds data on FAA programs. On the left, all of the data components programmed into the model are listed; columns on the right list the number of each component in each FAA program. New FAA programs can be added to expand the model by continuing to fill out columns to the right. Buttons above columns G through K copy each column of user-input programs to Calc1. **After this user input, the user must still return to the GUI worksheet and specify first refresh and repeat refresh cycles and click on the Calculate Now but - ton to run the model.**

Data on this sheet are copied by the macros VSCS_prgm_data, HOCSR_pgm_data, and PAMRI_pgm_data.

FAA SYSTEM OUTAGES

This page holds the social costs of a downed system if the system fails as a result of obsolescence. For each FAA system, there is a probability of an adverse event after the end of physical life (EOPL) is reached, and generally that probability increases every year. Up to 10 different events can be captured for the existing systems. Examples of events range from loss of backup systems, which has zero social cost, to the inability of an entire en route center to control aircraft, which has social cost in the billions of dollars. For each event, the social cost in terms of airline direct operating cost (ADOC) and passenger time can be recorded.

The intent of this page is to forecast in probabilistic terms the social cost of an outage if refreshment of a system is delayed beyond EOPL.

Presently this page does not interact with any other pages and is not connected to the model.

PARTS

This page is the model s main database. Here COTS components are entered and all their related economic data are stored. This page is linked to the next three pages, HW maint, SW maint, and Refresh. It is not referenced by macros; however, there are a few housekeeping macros that apply to it. They are run during model expansion and upkeep.

This page s column headings are defined below. Costs are in constant dollars unless indicated otherwise. On original programming of the model, dollars were in year 2000 terms. Not all the columns are used by each cost page. (The cost pages are HW maint, SW maint, and Refresh.)

Initial Hardware Cost acquisition cost of the original COTS hardware or equipment in constant dollars; the component s price.

Initial Hardware Support Cost annual cost of hardware maintenance during the initial years of a hardware component s life cycle (the phase in which original equipment manufacturer support is usually offered for free); the annual service fee.

Initial Bridge Hardware Support Cost annual cost of hardware maintenance after the end of a component s market life. Initial refers to the first year s cost; after that the cost is increased by the growth rate.

Bridge Hw Support Cost Growth Rate annual growth rate of cost of hardware maintenance after the end of a component s market life. Commonly found to equal 20 percent.

Initial Software Cost acquisition cost of the original operating system for the hardware component listed. Although the operating system and the hardware system are often procured in the same acquisition, in this model the hardware costs and operating system acquisition costs should be listed separately. If they are sold as one unit, the user should try to estimate the separate cost of each.

Initial Software Support Cost annual cost of software maintenance and support in the first years of a component s life. Although hardware maintenance costs and operating system costs are often bundled in a single maintenance contract, in this model the two costs should be costed separately or estimated as separate expenses.

Second Software Support Cost annual cost of software maintenance after the end of the operating system s market life.

2nd SW Support Growth Rate annual growth rate of cost of hardware maint e-nance after the end of a component s market life. Commonly found to equal 20˚percent.

Program Cost Percent the ratio of all other costs associated with acquisition in relation to prime mission equipment, specifically the COTS equipment. These costs include program management, initial spares, initial maintenance, logistics, test, evaluation, certification, systems engineering, training, documentation, de-livery, installation, and the like.

End of Hardware Market Life the year relative to the first year of sale, in which a piece of hardware is no longer offered for sale to all customers by its original equipment manufacturer and linked retailers.

End of Hardware Service Life the year in which the original equipment man u-facturer will no longer maintain, provide service for, or support a piece of hard-ware, relative to its introduction year.

End of Hardware Phys Life the year in which a piece of hardware must either undergo a SLEP, be replaced by a different piece of hardware, or be considered inoperative.

End of Software (Op Sys) Market Life the year relative to the first year of sale in which an operating system is no longer offered for sale to all customers by its original equipment manufacturer and linked retailers.

End of Software (Op Sys) Service Life the year in which an operating system manufacturer will no longer maintain, provide service for, or support an operating system, relative to its introduction year.

End of Software (Op Sys) Phys life the year in which an operating system should be considered inoperative because of the inability to obtain compatible hardware, inability to find programmers familiar with the language, or general progression of programming languages.

Unable to Mod SW Due Entropy the year in which an operating system should be considered inoperative because of the accumulated effect of small bug fixes over its useful lifetime.

SLEP Cost cost of a SLEP for this piece of hardware; generally, to re-engineer and produce one subcomponent that is failing in all hardware of this type.

EOL End of Life; the lesser of end of physical life for an operating system or for hardware. (See EOPL for hardware and software, above.)

Probability of Failure After EOL probability, per 100,000 hours of operation, of failure in the period after probabilistic end of life.

The source of input data on this page is covered in Chapter 5, Data Sources.

HW MAINT

This worksheet contains the maintenance cost algorithm for hardware, derived from FAA program experience. Costs are in year 2000 constant dollars, unescalated, and each row represents only one generic piece of equipment. Columns represent years into the future. This sheet depicts the standard lifetimes expected of various items of COTS hardware, regardless of whether they are used by a particular FAA system, and unmultiplied by the requirements of that system. For example, this sheet shows the lifetime maintenance costs of a PC, a router, and a hub. Other worksheets combine those elements in terms of 250 PCs + 0 routers + 20 hubs, and sum up their maintenance costs. This same logic is used in the SW maint and Refresh worksheets.

It is possible to have a separate maintenance cost algorithm for each type of COTS hardware. The model as delivered contains two related lengthy algorithms. They encompass the same cost-turning points, but because some hardware components lacked basic data (e.g., LAN) a different costing algorithm was used.

For basic computer equipment, the cost of maintenance through the end of market life is the hardware support cost. After this, the bridge support maintenance cost is in effect, increasing in price 20 percent per year from the point of end of market life. At the end of physical life, SLEP costs begin to occur. Experience reveals that SLEPs must be accomplished about twice per service life duration. We have no experience with the increase in SLEP costs after the second SLEP, so it was assumed that SLEP cost would increase by 10 percent each year. SLEP costs are annualized in this schedule, though in reality they vary between zero and a very high spike value every few years.

For communications elements of the HW maint worksheet (e.g., LANs, switches, routers, and hubs), the use of the computer algorithm produced higher costs in the mid-years than in the out-years, defying logic. Though we have little experience with out-year costs, we felt they should be at least as great as those for mid-years maintenance. The problem was that, using the computer algorithms, bridge period support costs increased dramatically, beyond the cost of SLEPs. So the formula was rewritten slightly, to remove the bridge period costs until we have more information. The logic of the algorithm remained the same. Use of formula 1 and formula 2 results in nearly the same estimates throughout computer equipment life; and the estimates are the same in the near years for all types of hardware.

Formula 1 (for computer equipment):

IF(D$2<=EOML_HW,suppt_HW,IF(D$2>EOPL_HW,(2*SLEP_cost*(1+(D$2-EOPL_HW)*0.1)/EOSL_HW),Bsppt_HW*(1+sppt_gr_HW)^(MAX(1,D$2-EOSL_HW))))

Formula 1 translation: Before the end of hardware market life, maintenance cost is equal to basic support cost. After the end of hardware physical life, a SLEP is required every half-service-life, and SLEP cost (annualized) increases by 10 percent each year. In all other years, bridge support contract prices apply, growing annually by the set growth rate.

Formula 2 (for communication equipment):

IF(D$2<=EOML_HW,suppt_HW,IF(D$2<=EOPL_HW,suppt_HW*(1+sppt_gr_HW)^(MAX(1,D$2-EOSL_HW)),IF(D$2>EOPL_HW,(2*SLEP_cost*(1+(D$2-EOPL_HW)*0.1)/EOSL_HW),suppt_HW)))

Formula 2 translation: Before the end of hardware market life, maintenance cost is equal to basic support cost. Between the end of hardware market life and the end of physical life, hardware support cost grows annually. After the end of hardware physical life, a SLEP is required every half-service-life, and SLEP cost (annualized) increases by 10 percent each year.

SW MAINT

This worksheet contains the licensing, support, and maintenance cost algorithm for software (operating systems), derived from FAA program experience. Costs are in year 2000 constant dollars, unescalated, and each row represents one license only. Columns heading to the right represent years into the future.

It is possible to have a separate software maintenance cost algorithm for each type of COTS hardware. The model as delivered contains only one algorithm for all operating systems.

For equipment that does not have an operating system (e.g., LANs, switches, and routers), operating protocols are assumed to take the place of operating systems. The move from Internet Protocol version 4 to Internet Protocol version 6 is an example of a distinct and dramatic generational upgrade for routers and switches that limits their useful lives. Even fiber optic cable, with a 30-year projected physical life, depends on state-of-the-art coding techniques when it is buried. For example, wave division multiplexing using a multitude of colors is an emerging technology that enables one fiber to carry dozens of channels. But cables buried with older repeaters may be limited to non-color use by their repeater/amplifiers. So, for media like fiber optic cable, operating system is extended to include coding technologies.

The formula for software states,

IF(D$2<=EOML_SW,suppt_SW,IF(D$2>EOPL_SW,Bsppt_SW,suppt_SW*(1+sppt_gr_SW)^(MAX(1,D$2-EOML_SW))))

Translation: Before the end of software market life, support costs apply. After the end of software physical life, the B support schedule applies. Between these two periods, software support cost grows at an annual rate.

The software formula differs dramatically from the hardware formula, because the support procurement for software differs dramatically. Generally, support for an operating system is obtained up front for the expected physical life of the software, usually about 10 years. After 10 years have passed, if the user still requires support, one of two things happens: support is either exorbitant or nonexistent (and nonexistent is free.) Thus, during the life of the software, annual maintenance costs tend to be fairly linear, increasing slightly from year to year in accordance with contract allotments. After the life of the software has ended, the B schedule of costs is incurred.

During the B period, if the system is truly COTS that is, it is unmodified for the user by this time the bugs are out and the software can run for a few more decades without internal problems, support is unavailable and not really needed. If the system is modified COTS, then the FAA has a proprietary license for the software that becomes exorbitantly expensive at the end of the software s expected physical life. The latter has been the case for most of the systems we studied at the FAA. It is impossible for us to predict, in this generalized model, whether future FAA systems will go the way of true COTS or modified COTS. We have modeled the latter. The advanced user is cautioned, however, that instead of spiking upward, actual license costs may suddenly drop to near zero, when an unmodified system reaches the end of its life.[1]

REFRESH

This worksheet contains the cost algorithm that calculates refresh cost for a bundled hardware and software piece of equipment, in any given year. Because COTS equipment is updated with each replacement release, as time goes on new releases have less and less in common with the installed versions until after a few versions, there is no commonality at all. This is why refresh costs increase over time; as time moves on, more of a bridge must be built between the new and the installed systems.

The model was constructed to assume that we are at year zero of an expected 10-year lifespan for operating systems. Operating systems completely re-invent

[1] This occurred in the system that preceded VSCS, when Western Electric Company opted to discontinue leasing the T-bar WECO switches to the FAA and instead sold them to the FAA the following year for a dollar. Leasing costs suddenly dropped to zero. It is unlikely that this corporate largesse will be repeated for VSCS and PAMRI operating system license fees.

themselves approximately every 10 years; the conversions from OS-DOS to MS-DOS and to Windows 95 are examples. The model could be more accurately used if the user can identify the last operating system revolution for the system being analyzed and work from there. For example, if the user knows that the last major operating system was introduced 5 years ago, then the user should program the model as if today is year 5, and years 1 through 4 have already gone by. The large spikes in maintenance costs will occur much sooner than if the analyst uses the default zero year, but the estimate will be closer to reality.

It is possible to have a separate software maintenance cost algorithm for each type of COTS hardware. The model as delivered contains only one algorithm for all systems. Costs are in year 2000 constant dollars, unescalated, and each row represents one bundled hardware and operating system combination only. Columns show years in the future.

The basic formula for refresh cost estimating is:

IF(I$2<=EOML_HW,cost_HW*PrgmCost,IF(I$2<=EOML_SW,(cost_HW+30%*cost_SW)*PrgmCost,IF(I$2<EOPL_HW,(cost_HW+cost_SW)*PrgmCost,IF(I$2>=EOPL_HW,(cost_HW+cost_SW+SLEP_cost)*PrgmCost))))

Translation: until the end of the hardware s market life, refresh cost is equal to that of replacing the hardware times the mark-up for program costs. From the end of the hardware s market life until the end of the operating system s market life, the refresh cost is the cost of replacing the hardware and porting the software to the new hardware (30 percent of the operating system cost), times the mark-up for program costs. From the end of the operating system s market life until the end of the hardware s physical life, the refresh cost is the cost of the new hardware and the new operating system, times the program cost. After the end of the hardware s physical life, the refresh cost is the cost of replacing the hardware and the software and of building an intermediary system (about equal to SLEP cost), times the program cost mark-up.

BUDGET

The Budget worksheet is fairly involved, since it is where all the error checking for user-inputted budgets occurs and where the model formulates optimal F&E funding profiles. It executes its computations in steps, which are grouped into rows. Accordingly, the worksheet will be explained by rows.

In row 1, cells E1 through Q1 list the years of analysis, from 1 to 13. Cell D1 is a holding cell for a key variable, userbudvar, which is set by userbudsel1, userbudsel2, or userbudsel3, all of which are button-activated from the GUI worksheet. When the main engine is run, this variable is picked up by choose_ budget(userbudvar), and the value of userbudvar tells choose_budget whether to run default_budget1, default_budget2, or paste_input_budget.

Row 2 re-displays (from Calc2) the cost of refresh in any year from 1 through 13 for a selected FAA program. This row is displayed automatically once a program is selected, before any of the main engine macros are invoked. Each cell displays the total cost of refreshing the program if the program were to be refreshed for the first time in that year (i.e., the cell for year 4, H2, shows the cost of refreshing that FAA program for first refresh in year 4). Simultaneously, the cell for year 5, I2, shows the cost of refreshing that FAA program for first refresh in year 5. In the third row, cells E3 through Q3 bring down the first refresh cost in the year the user has selected for first refresh and zero out all the other columns. Cell D3 shows the sum (i.e., the amount of refresh in the year the user has selected). All the updating in row 3 is automatic. Basically, rows 2 and 3 perform the same function as a lookup() function, but use less memory. This could also be written into a macro function, to consume even less memory.

In the sixth row, cells E6 through Q6 do basically the same thing as row 3, except that they show a 1 in the year of tech refresh and zeroes in all the other years. Rows 7 and 8 figure out when F&E funding should be spent in a model-planned budget, showing 1 s in the years of positive funding. Row 7 plans for a 3-year acquisition and row 8, a 4-year acquisition. Rows 7 and 8 are sensitive to the first refresh year and contain if-then statements to ensure that funding precedes fielding and that the final year of funding and the year of fielding are the same. Row 9 spreads out the percentage of funding by year corresponding to the defined 4-year funding cycle, 10%—20%-40%—30%. Cells S6 through X9 contain funding percentage templates for row 9 to reference if the user selects a 4-year funding profile but refresh takes place before year 5. If a user wants to modify the model to replace the 10%—20%—40%—30% profile with a more commonly used one, this would be the place to make the modifications. All of rows 6 through 9 are updated automatically.

Rows 13 through 15 apportion the F&E funding requirements as identified in cell C14 among the funding years, shown in rows 7 through 9. C14 is the same as cell D3, but expressed in thousands, to match the user-input budget figures. Row 14 multiplies the needed funding by the predefined budget profile of 33%—33%—33%; in row 15 the same is done for the predefined budget profile of 10%—20%—40%—30%. Rows 13 through 15 update automatically.

Rows 21 through 23 show the chosen budget stream in dollars and in thousands of dollars. Row 22 is written by macros only. If the user inputs a budget stream, that budget is pasted here, into row 22. If the user selects a model-defined budget stream, that stream is copied from row 14 or row 15 and pasted into row 22. Row˚23 is updated automatically from row 22, converting to dollars from thousands of dollars.

Rows 24 through 31 perform the automatic error checking on budget profiles. Row 24 counts up the number of years in which there is F&E funding. The user may input a budget stream 7 years long, if desired. When planning the repeat refresh cycles, the model needs to know how many years are used out of all possible

funding years. This number is provided in cell O24, which sums the 1 s in row 24. Cell O24 is called by the select_budget_horizons macro and used automati - cally by rows 40 through 44.

Row 25 contains cells that compare F&E cumulative expenditure to total and show a 1 when 95 percent of required funding is reached. The 1 indicates the year the model considers sufficient funding to have been reached, and fields all refresh in this year. Cell O25 houses the NSF key variable; it is one if sufficient funding is never reached and zero otherwise.

Row 26 compares the desired first refresh year with funding and computes the percentage of funding being spent after first refresh. If more than 50 percent of funding falls after fielding, cell O26 indicates this, triggering an error message on the GUI worksheet under the budget section. Row 27 similarly computes the amount of F&E expenditure in the fielding year, while Row 28 computes the amount of F&E expenditure in the year immediately preceding fielding. Cells O27 and O28 automatically indicate if *less than* 10 percent of F&E funding falls in either of these years; O29 is the funnyfunding key variable triggered if both O27 and O28 are 1. That is, the funnyfunding variable equals one if less than 20˚percent of F&E funding is spent in the year of fielding and the year preceding fielding.

Rows 30 and 31 identify the year in which budget expenditure is sufficient to field the refreshment. O31 is the key variable styear2. If the user-input F&E stream is insufficient to support a tech refresh, cell O30 sums the year of first refresh and the regular refresh cycle to generate the year of first refresh. Thus the model will keep adding higher and higher maintenance costs on the system to be re- freshed, until that system enters its regular refresh cycle. For example, if the user asks for first refresh in year 3 and regular refresh every 5 years thereafter, then enters an insufficient funding profile, the model will schedule refresh in year 3+5=8. This was done to reflect what may happen when budgets are cut so much that refresh may not be fielded in the near future.

Rows 35 through 48 automatically generate repeat F&E funding profiles for the model to use over the next 30 years. All of them are based on if-then statements, and all are automatically driven. Rows 35 through 38 repeat the user-input budget stream for 3, 5, 7, and 10-year repeat cycles. Rows 40 through 45 try to use the user-input F&E stream if the only adjustment it needs is temporal. For instance, if the user inputs a budget stream for years 1, 2, and 3, and selects a refresh cycle of 5 years, the model will re-use the user s budget numbers but will push them out so that the final year of funding corresponds to the year of fielding. Rows 45 through 48 are model-generated 30-year funding profiles based on the default profiles of 33%-33%-33% and 10%-20%-40%-30%. Each profile is numbered (rows 35˚through 48); the numbers correspond to the calling functions of the se- lect_budget_horizon macro, which weighs the input variables budyrs, trc y- cle, funnyfunding, NSF, styear, budhor, and selects the proper budget

cycle to use. It copies one of the 12 available budget profiles and pastes it into the Calc2 worksheet.

Note: One known bug of the model is that a 1-year funding profile corresponding to a 10-year refresh cycle will probably be depicted inaccurately. This is a result of a limit on excel if-then functions, in row 43 of the model.

MODULES AND MACROS

The model s 59 macros are embedded in 6 pages; the pages are also called modules by Excel. Table 3- lists the module names and the macros in each. By selecting the module page and clicking on view code, it is possible to see and edit the macros that run the model. (One can access the modules in Excel 98 by selecting Tools from the menu bar, then Macros. Select any of the macros in the drop-down box, and click on edit macro.) Each macro s purpose, impacts on Excel pages, and use of key variables are explained following Table 3-1.

Table 3-1. List of Macros and Their Module Locations

Module (page)	Macros
Budget	choose_budget(userbudvar)
	paste_input_budget()
	Default_budget2()
	Default_budget1()
	userbudsel1()
	userbudsel2()
	userbudsel3()
FAAprograms	clear_old_prgmdata()
	VSCS_prgm_data()
	HOCSR_pgm_data()
	PAMRI_pgm_data()
	new_prgm_data1()
	new_prgm_data2()
	new_prgm_data3()
	new_prgm_data4()
	new_prgm_data5()
finishing	output()
	show_data()
	return_to_graph()
	gotobudgetGUI()
	return2CEGUI()
	switch_data_series()
housekeeping	Update_parts()
Main_engine	Define_cycles_3()
	Define_cycles_5()
	Define_cycles_7()

Table 3-1. List of Macros and Their Module Locations (Continued)

Module (page)	Macros
	Define_cycles_10()
	Horizons()
	choose(styear1, styear2, styear)
	Start_cycles_right(styear)
	select_budget_horizon(budyrs, trcycle, funnyfunding, NSF, styear, budhor)
	Push1()
	Push2()
	Push3()
	Push4()
	Push5()
	Push6()
	Push7()
	Push8()
	Push9()
	Push10()
	Pull1()
	Pull2()
	Pull3()
	Pull4()
	Pull5()
	Pull6()
	Pull7()
	Pull8()
	Pull9()
	Pul10()
	Pul11()
	Pul12()
	Pul13()
NvrRefr	Never()

MACROS OF THE BUDGET MODULE

userbudsel1()

The user calls userbudsel1() by clicking on the button corresponding to the 33%—33%—33% default budget on the GUI worksheet. Its sole action is to set a value of 1 in the userbudvar key variable, housed in cell D1 on the Budget worksheet.

userbudsel2()

The user calls userbudsel2() by clicking on the button corresponding to the 10%—20%—40%—30% default budget on the GUI worksheet. Its sole action is to set a value of 2 in the userbudvar key variable, housed in cell D1 on the Budget worksheet.

userbudsel3()

The user calls userbudsel3() by clicking on the button corresponding to the Analyze the budget I ve enterted below on the GUI worksheet. Its sole action is to set a value of 3 in the userbudvar key variable, housed in cell D1 on the Budget worksheet.

choose_budget(userbudvar)

Choose_budget is a selection macro. It goes to the Budget worksheet, finds the key variable userbudvar in cell D1, and runs the macro that corresponds to the userbudvar value. Userbudvar can be 1, 2, or 3, and these values summon the macros Default_budget1, Default_budget2, and paste_input_budget, respectively. Only one of these three will be run.

Default_budget1()

Default_budget1 goes to the Budget worksheet; copies an optimally timed, automatically generated budget profile from row 14; pastes it on the Budget worksheet into row 22 (working space); then pastes it into the GUI worksheet in the user s budget boxes (cells C38 through L38); clears its clipboard; and then quits. Cell M38 on the GUI worksheet automatically sums the entries of cells C38 through L38.

Default_budget2()

Default_Budget2 goes to the Budget worksheet; copies an optimally-timed, automatically generated budget profile from row 15; pastes it on the Budget worksheet to row 22 (working space); then pastes it into the GUI worksheet in the user s budget boxes (cells C38 through L38); clears its clipboard; and then quits. Cell M38 on the GUI worksheet automatically sums the entries of cells C38 through L38.

paste_input_budget()

Paste_input_budget copies the user-input budget from the GUI worksheet (GUI!C38:L38) and pastes it into the Budget worksheet s active workspace, cells E22 through N22.

MACROS OF THE FAA PROGRAMS MODULE

VSCS_prgm_data()

The user calls VSCS_prgm_data() by clicking on the VSCS program button on the GUI worksheet. It goes to the worksheet containing all FAA component information, FAA Systems, copies the column corresponding to VSCS (column D), and pastes it into Calc1 s column D. The macro writes VSCS on the budget analysts GUI space, clears, and then quits. Calc1 s column D s values are automatically multiplied by components year-sensitive maintenance and refresh costs in the columns to the right of D. Those values are automatically updated to Calc2.

HOCSR_pgm_data()

The user calls HOCSR_pgm_data() by clicking on the HOCSR program button on the GUI worksheet. It goes to the worksheet containing all FAA component information, FAA Systems, copies the column corresponding to HOCSR (column E), and pastes it into Calc1 s column D. The macro writes HOCSR on the budget analysts GUI space, clears, and then quits. Calc1 s column D s values are automatically multiplied by components year-sensitive maintenance and refresh costs in the columns to the right of D. Those values are automatically updated to Calc2.

PAMRI_pgm_data()

The user calls PAMRI_pgm_data() by clicking on the PAMRI program button on the GUI worksheet. It goes to the worksheet containing all FAA component information, FAA Systems, copies the column corresponding to PAMRI (column F), and pastes it into Calc1 s column D. The macro writes PAMRI on the budget analysts GUI space, clears, and then quits. Calc1 s column D s values are automatically multiplied by components year-sensitive maintenance and refresh costs in the columns to the right of D. Those values are automatically updated to Calc2.

clear_old_prgmdata()

This macro is summoned by the Make inputs now button on the GUI worksheet. This macro goes to Calc1, clears whatever program data are in column D from the last macro run, puts a 1 in one of the rows in column D, takes the user to FAA Systems, leaves the cursor at the top of column G, and then quits. The 1 is left in a single row in column D so that #DIV/0 errors do not corrupt the model s worksheets while new program data are being entered. The cursor is left at the top of column G so that the user can input a list of components for a new system for the model to analyze. After the user inputs data, the button on top of

the new data column should be clicked in order to input the data to the model (see next macro).

new_prgm_data1()

New_prgm_data1, new_prgm_data2, new_prgm_data3, new_prgm_data4, and new_prgm_data5 are nearly identical macros that copy the contents of user-input systems from columns G through K on FAA Systems into column D of Calc1. These macros are activated by clicking on the buttons above the new input, labeled new 1, new 2, etc. From there, the model follows the same process as with the model-supplied programs (i.e., the user must return to the GUI worksheet, specify first refresh and regular refresh, and press the Calculate Now button).

To have a program in columns G through K permanently added to the model, an advanced user should create a button inside the Programs Group Box on the GUI worksheet; write a macro that does what VSCS_prgm_data, HOCSR_pgm_data , and PAMRI_pgm_data do; and assign the new macro to the new button. To write the new macro, it should be possible to copy any of the *N*_pgm_data macros on the FAA Systems module, paste the code into a new macro, rename the macro (e.g., from HOCSR_pgm_data to Radar_pgm_data), and change the columns copied by the macro to the new program input (e.g., from E6:E5000 to H6:H5000). For more detail, see the chapter on Expanding the Model. Alternatively, the user can call the model developer at LMI at 703-917-7167 for changes.

MACROS OF THE FINISHING MODULE

output()

This macro is the last macro called by the Horizons main engine macro. It pastes the model output from the Calc2 worksheet into the Output Chart worksheet and formats the data in dollar format. The Output chart is automatically generated from these numbers.

show_data()

The show_data macro is called by a button on the Output Chart worksheet; it redirects the screen view from the Output chart to the cells containing the numerical output.

return_to_graph()

The return_to_graph macro is called by a button on the lower part of the Output Chart worksheet; it redirects the screen view from the numerical output back to the Output chart.

gotobudgetGUI()

This is another positioning macro. GotobudgetGUI is activated by a button on the GUI worksheet. Its function is to reposition the viewscreen so that the budget analyst s input GUI is in view. Because viewscreens vary in size, this macro and other positioning macros may not work perfectly. It is possible to write a macro that takes over a user s workbook and re-sets the defaults to that the viewscreen fits exactly; however, this can be very annoying to the user and so this function was not integrated into the model.

return2CEGUI()

Return2CEGUI is a positioning macro that returns the analyst from the budget inputs view to the Tech Refresh parameters screen on the GUI worksheet. It was created to help the user return to the Calculate Now button once budget inputs have been made.

switch_data_series()

This macro re-arranges the rows displayed on the output chart so that F&E or maintenance can be brought to the forefront. In a particular model run, maintenance may be larger than F&E, and so F&E on the chart will be obscured by maintenance. This button brings F&E to the front and reverses the rows again if so desired.

MACROS OF THE HOUSEKEEPING MODULE

The macro update_parts is the only one in the Housekeeping module. It is not used by the model directly; it is one of the macros provided for expanding and updating the model. When new hardware components are added to the model—for example, Ku-band radar—they should be added to the end of the Parts worksheet, in column D, and their relevant parameters should be filled in in the columns to the right. The update_parts macro can then be activated to copy the column D information to the relevant pages in the model that contain identical information.

For more information on expanding the Parts database, see Chapter 4, Expanding the Model.

MACROS OF THE MAIN ENGINE MODULE

Define_cycles_3()

The Define_cycles_3 macro sets up the Calc2 worksheet for analysis of a repeat refresh cycle every 3 years. It pastes a 3 in the key variable space

Calc2 :A102 for reference by the select_budget_horizon macro and pastes the 3-year repeating maintenance cycle from F84:AI84 into row 106, where it will be copied by the start_cycles_right macro into row 107. This macro is called by the 3 years button in the Regular Refresh group box on the GUI worksheet. It operates before the main engine macros.

Define_cycles_5()

The Define_cycles_5 macro sets up the Calc2 worksheet for analysis of a repeat refresh cycle every 5 years. It pastes a 5 in the key variable space Calc2 :A102 for reference by the select_budget_horizon macro and pastes the 5-year repeating maintenance cycle from F85:AI85 into row 106, where it will be copied by the start_cycles_right macro into row 107. This macro is called by the 5 years button in the Regular Refresh group box on the GUI worksheet. It operates before the main engine macros.

Define_cycles_7()

The Define_cycles_7 macro sets up the Calc2 worksheet for analysis of a repeat refresh cycle every 7 years. It pastes a 7 in the key variable space Calc2 :A102 for reference by the select_budget_horizon macro and pastes the 7-year repeating maintenance cycle from F86:AI86 into row 106, where it will be copied by the start_cycles_right macro into row 107. This macro is called by the 7 years button in the Regular Refresh group box on the GUI worksheet. It operates before the main engine macros.

Define_cycles_10()

The Define_cycles_10 macro sets up the Calc2 worksheet for analysis of a repeat refresh cycle every 10 years. It pastes a 10 in the key variable space Calc2 :A102 for reference by the select_budget_horizon macro and pastes the 10-year repeating maintenance cycle from F87:AI87 into row 106, where it will be copied by the start_cycles_right macro into row 107. This macro is called by the 10 years button in the Regular Refresh group box on the GUI worksheet. It operates before the main engine macros.

Horizons()

Horizons is the director macro of the main engine macros. It is called when the user clicks the Calculate Now button, and it in turn calls all the macros of the main engine. Specifically, it calls Choose_budget and then defines the variables trcycle, budyrs, funnyfunding, NSF, styear1, and styear2. These variables are stored in cells on the Calc2 and Budget worksheets. Some are automatically defined; others are written by macros. (Table 3-2 lists all the key variables and their locations.) These variables are needed by the next macros to be called. Horizons then calls choose(styear1, styear2, styear), se-

lect_budget_horizon(budyrs, trcycle, funnyfunding, NSF, styear, budhor), and start_cycles_right(styear) and finishes with output().

Table 3-2. Key Variables Functions and Locations

Variable	Stored location	Function	Written by
Trcycle	Calc2!A102	Defines repeating refresh cycle in years	Define_cycles_3 Define_cycles_5 Define_cycles_7 Define_cycles_10 or Never()
Budyrs	Budget!O24	Indicates number of years with F&E funding	Automatic
Funnyfunding	Budget!O29	Indicates a stop and restart in user-defined F&E funding	Automatic
NSF	Budget!O25	Indicates insufficient user-defined F&E funding	Automatic
Styear1	Calc2!A101	User-desired first refresh year	Automatic
Styear2	Budget!O31	Year in which F&E funding is sufficient to allow first refresh fielding	Automatic
Styear	Macro variable	Greater of styear1 and styear2	Choose()
budhor	Macro variable	Tells macro which budget stream to use	Select_ budget_horizon()

Note that most of these variables are also listed in column A on worksheet Calc2, from rows 112 through 121. Those cells are reporting locations rather than storage locations. The reporting and storage functions should not be confused.

We will now discuss the working of the macros called by horizons.

choose(styear1, styear2, styear)

Choose() is simply the visual basic code equivalent to an Excel MAX() function. Styear is set equal to the greater of styear1 and styear2, using if-then statements.

The model-based purpose of this macro is to not allow refresh to take place if the F&E budget is insufficient, a requirement of the budget analysts. Thus a budget analyst can analyze what would happen if 10 million dollars were pushed out a year or two for a particular system.

select_budget_horizon(budyrs, trcycle, funnyfunding, NSF, styear, budhor)

This is the model s most complicated macro. The function of select_budget_horizon is to choose a regular refresh F&E cost stream. The model automatically generates 12 F&E cost streams to choose from on the Budget worksheet. Select_budget_horizon is basically a series of if-then statements to set the value of budhor, and the value of budhor determines which of the 12 F&E cost streams is used. Budhor is then used in a select-case function to call the correct pull *n* function which will copy the desired budget stream from Budget to row 108 on Calc2. Pull *n* indicates the set of pull1, pull2, pull3, pull4, and other macros, up to pul13. Each pull*n* function copies a separate budget horizon; only one of the pull*n* macros is called by select_budget_horizon.

If the user inputs an F&E stream and the stream is optimally timed, that stream will be re-used over the 30-year cycle. Budget horizons 1 through 4 re-generate the user s optimal budget stream over 30 years. Budget horizon 1 (budhor=1) regenerates the user s budget to fund a 3-year regular refresh cycle. Budget horizon 2 (budhor=2) regenerates the user s budget to fund a 5-year refresh cycle. Budget horizon 3 (budhor=3) regenerates the user s budget to fund a 7-year regular refresh cycle. Budget horizon 4 (budhor=4) regenerates the user s budget to fund a 10-year refresh cycle.

If the user inputs a budget stream longer than the regular refresh cycle, the model uses one of its own default budgets for the regular refresh cycle. Budget horizons 9 through 12 (budhor =9,10,11,or 12) are the model-generated F&E streams for 3-, 5-, 7-, and 10-year regular refresh cycles respectively.

If the user inputs a budget stream that is chronologically front-loaded but otherwise sufficient, the model pushes the funding stream back in time and uses it to generate 30-year budget streams. Budget horizons 5 through 8 (budhor = 5,6,7 or 8) are the time-altered budget streams for 3-, 5-, 7-, and 10-year regular refresh cycles respectively.

If the user s F&E budget stream is insufficient or activates the funnyfunding variable, or if the user did not input a budget stream, then the model-generated budget horizons 9 through 12 are used.

If the user has selected never refresh, select_budget_horizons detects this from the key variable trcycle and runs a unique macro that zeroes out all F&E funding over 30 years.

Pull1 Through Pul13

These macros do the copying work associated with the select_budget_horizon macro. They pull the required 30-year F&E cost stream from the Budget

worksheet and paste it into row 108 on Calc2, so that it is in place when the pushn macro is summoned. Only one pull *n* macro runs in each model run; select_budget_horizon uses a select-case routine to select which one to run.

Start_cycles_right(styear)

This function copies the regular maintenance cost stream and the model-generated F&E stream over the next 30 years and pastes them so as to start in the correct year relative to the first refresh. For example, the model user may select a first refresh in year 5 and regular refresh every 7 years afterwards. The model creates F&E and maintenance cost streams for the first 5 years and a repeating refresh cycle based on 7 years. This macro then pastes the repeating refresh cycle so as to start in year 6. If the user then alters his choices to have first refresh in year 7, the model re-generates first refresh cost streams and pastes the repeating refresh cycle so as to start in year 8.

Specifically, the macro runs a select-case routine on the variable styear, act i-vating one of the Push*n* macros (push*1*, push*2*, push*3*, push *4*, and so on, to push*10*.) The push*n* macro copies the repeating maintenance cycle from cells F106:AI106 and pastes it into row 107, and copies the repeating F&E stream from cells F108:AI108 and pastes that into row 109. If styear equals 1, push1 is run. If styear equals 2, push2 is run. If styear equals 3, push3 is run, and so on, up through 10. Push1 pastes into G107:AJ107 and G109:AJ109; push2 pastes into H107:AK107 and H109:AK109; push3 pastes into I107:AL107 and I109:AL109, and so on.

Because the macro depends on the variable styear, the repeating maintenance and F&E streams do not start until sufficient F&E funding is expended to finish the first refresh cycle. The exception occurs when F&E funding is positive but never enough to fully fund the first refresh. Then the model will assume that first refresh is started but never fielded, and maintenance costs will grow until the regular refresh cycle kicks in.

Push1 through Push10

These macros do the copying work associated with the start_cycles_right() macro. They push out the regular refresh maintenance and F&E cost streams to the year after the first refresh is accomplished. Only one push*n* macro runs in each model run; start_cycles_right() uses a select-case routine to select which one to run.

MACROS OF THE NVRREFR MODULE

The NvrRefr module is so named because Excel does not allow a macro and a module to have the same names, and never and never refresh were already

taken when the module was written. For this reason, it is important that the user NOT change module names.

The sole macro of NvrRefr is Never(). Never is a fairly simple macro that accomplishes the equivalent of the Define_cycles_*n* and paste_input_budget macros. It zeroes out the first refresh variable, zeroes out the F&E budget stream, and sets the key variable trcycle to 100. As is the case with any other tech r e-fresh set of options, Calculate Now must be activated and Horizons must be run for the model to produce the correct result.

Chapter 4
Expanding the Model

ADDING FAA PROGRAMS

New FAA programs can be temporarily or permanently added to the model. It is advisable to add as many actual acquisitions as possible to the model permanently, so that the model can eventually be used as a database to derive cost-estimating relationships for FAA tech refresh. Temporary additions may be desired, however, if a product team wants to consider alternative technologies for a given goal and wants to include tech refresh cost estimates as one of the parameters for deciding between competing architectures.

Temporary Data Input

To add a program to the model temporarily, the user should click on the Make Inputs Now button in the Programs group box on the GUI worksheet. This macro will deliver the user to the FAAPrograms worksheet, column G, where a new program can be entered. Hopefully, before getting to this point, the user already has a good idea of the required components and their number for a particular program. The user should enter the number of each piece of hardware needed for the entire program. For example, if a LAN, a RAID, a UPS, and 20 workstations are needed at each center for this program, the user should enter 22 LANs, 22 RAIDs, 22 UPSs, and 440 workstations. The units fielded at the FAA Academy, FAA William J. Hughes Technical Center, and Central Flow should also be included in this column. The user should include a label for the new program in cell G5, especially if a number of alternatives are to be considered. Each alternative architecture should be entered in a new column. Once a column is finished, the user can click on the button (e.g., new 1) at the top of the new column, and the program data will be copied into Calc1 and the first model computations can be made. After this, the user must return to the GUI worksheet, specify first refresh and regular refresh, enter an F&E budget stream if desired, and press the Calculate Now button. The model will calculate the tech refresh costs of one alternative at a time, so for a number of inputs the user will have to follow the following steps:

1. Click on Make inputs now on the GUI worksheet.

2. Find the next blank column on the FAA Systems worksheet. In that co l-umn enter the number of each component needed. Enter a label in the fifth row (e.g., 3/99 version, vendor_version, etc.).

3. Click the new program button at the top of the column just entered (e.g., new 1, new 2, new 3, new 4, new 5).

4. Enter tech refresh choices on the GUI worksheet.

5. Press Calculate Now.

Users will likely find that the listing of COTS hardware does not include all of the hardware pieces they need in order to fill in a new FAA system. Refer to the section on Adding to the Parts Database, below.

Permanent Data Input

To have a program in these columns permanently added to the model, users can program in the inputs and macros themselves, if desired. The steps a user should follow are listed below. It is recommended that the user be familiar with Visual Basic and/or Excel macros before attempting this modification. Alternatively, the user can call the model developer at LMI at 703-917-7167 for changes.

The steps to add a program permanently are as follows:

1. Create a button inside the Programs Group Box on the GUI worksheet with the name of the new program.

2. Be sure the new button is part of the Programs Group Box. Note: this will ensure that only one button is highlighted at a time. It does not affect the running of the model.

3. Copy and paste any of the *N*_pgm_data macros on the FAA Systems module. Change the name of the copied sub (e.g., from HOCSR_pgm_data to Radar_pgm_data) and change the columns copied by the macro to the new program input (e.g., from E6:E5000 to H6:H5000). Also change the label updating to the budget-users GUI, cell C26.

4. Disable or change the reference of the new_prgm_data *n* macro associated with the new program s column. For instance, if you entered a new program in column G, either disable new_prgm_data1() (by inserting a comment mark at the beginning of each line of code) or change its references from G6:G5000 to an unused column (e.g., L6:L5000). Move or remove the link from the button on the FAA Systems worksheet associated with that new_prgm_data *n* macro.

5. Re-set the macro clear_old_prgmdata() so that the last line deposits the user at a blank column of FAA Systems.

Users will likely find that the listing of COTS hardware does not include all the hardware pieces they need in order to fill in a new FAA system. Refer to the section on Adding to the Parts Database, below.

ADDING TO THE PARTS DATABASE

When new hardware components are added to the model e.g., Ku-band radar, motherboards, radios, vocoders they should be added to the *end* of the Parts worksheet, in column D, and their relevant parameters should be filled in in the columns to the right. New parts must be added to the end of column D, because if they are added by inserting rows in between the existing parts data, then the numbers associated with HOCSR, PAMRI, and VSCS on other worksheets will no longer align with the proper components. (In some cases, blank rows are provided for upgraded models of listed parts—such as row 46, reserved for higher speed hub, where a limited number of new components can be added.)

The user can keep track of the pre-programmed data on HOCSR, PAMRI, and VSCS and re-enter the data after having inserted rows in the Parts database. However, this procedure is tedious, and the user s database will no longer match up with other users programs.

If new parts are added to blank rows at the end of existing data in the Parts database, then the update_parts macro can be activated to copy the column D information to the relevant pages of the model that contain identical information. Note that it will probably be necessary to copy and drag down the calculating formulas for new rows in the HW O&M, SW O&M, Refresh, and Calc1 worksheets; these have been left blank to preserve calculating memory for older computers.

It may be desirable for users of the model to send their additions to the model back to the model designer on an annual basis, so that all additions can be combined and encoded in a logical manner. For example, a user may compile a list of 4,000 COTS components that he/she has added to the database over time as programs have come up, but the components are not arranged in logical groups. The model could be re-issued with the new data in logical groups, and existing program data preserved. If there is sufficient demand for it, the model could be re-hosted in a database program that is indifferent to the order in which data are added.

Crane Surface Naval Warfare Center COTS Refresh Model

Crane Surface Naval Warfare Center has developed a COTS Refresh Model over the past 4 years for NAVSEA, populating it extensively with the COTS components the Navy has used. That model is used to predict the obsolescence of installed systems and to project refresh requirements. Crane keeps its prediction

capability up to date by polling manufacturers every quarter and recording the end of market life, end of service life, and introduction of new products.

The Crane model began as an Excel spreadsheet, was enhanced by the addition of ACE-IT tools, and eventually was re-coded into an Access database with an excel-like user interface. Crane s Jerry Braun[1] and Mike Robey, who maintain the model, are just beginning to derive estimating relationships from their vast storehouse of data. It would be fruitful to combine and compare the Crane data-rich model with our rule-based model; each model would gain something.

ADDING OR CHANGING ESTIMATING RELATIONSHIPS

The relationships between components and their ages and the cost of maintaining or replacing them as depicted on the Parts worksheet are the result of studying a limited number of FAA programs, research into commercial practices, and research on the subject of tech refresh. The user may decide that he or she has uncovered a superior relationship and may want to program this relationship into the model. New relationships can easily be programmed on new lines as if they were wholly new components in the Parts database. It is more difficult to change the existing relationships, because they are fairly complex. Any user attempting to change them is urged to consult Chapters 2 and 3. The following examples illustrate some ways in which the database can be altered.

Example 1. Change the Lifetime of a Component

An analyst finds that, according to FAA field data, Unix servers live 4.5 years and then are completely unusable. There are 2 possible ways to reflect this in the model.

Replace solution: Under Computers—Unix, the user changes End of Hardware Phys. Life from 10 to 4.5. Use this line when programming in the components of this project.

Add solution: Go to the bottom of the parts database row 61 and enter Unix servers into the component list. Enter the reliability and lifetime data as desired. Enter 4.5 under End of Hardware Phys. Life.

Example 2. Change Several Parameters of a Component

An analyst on the terminal integrated product team finds that training PCs cost $5,000 new and are obsolete in 2 years. Moreover, the maintenance agreement costs $500,000 per year for the facility, but it includes tech refresh.

[1] Jerry Braun, Commercial Technology Support Branch, NSWC Crane Division. Telephone: 812-854-5328

Solution: create a new line for training PCs. Find out more about the maintenance agreement. A little research reveals that there are 10 PCs in the room, facility costs are about $900 per month, instructor costs are GS-14 equivalent, and the organization doing the training replaces PCs when they are 2 years old. Their services are contracted for on an annual basis. The analyst deduces that the FAA is paying $2,780 per month in maintenance, licensing, and application fees. From experience with similar cases, the analyst decides that 80% of the fees are for the emulation software (the application).

Fill in the Parts table to reflect a 2-year physical life, a new price of $5,000, and enter annual support/maintenance costs for hardware and software of $3,342 each. (If 80% of fees are for licensing the application, then 20% must be for maintenance. Assume half of the maintenance is for hardware, half for the operating system.). Fill in the same amount for the bridge and B support costs; under this contract, the maintenance cost does not rise over time. Under support growth rates for hardware and software, put 0%.

Alternatively, if capturing the unvarying nature of the payments is important, the analyst could input a 0$ price, a lifetime of 2 years, and an annual hardware and software maintenance cost of $3,592 each. Fill in the same amount for the bridge and B support costs; under this contract, the maintenance cost does not rise over time. Under support growth rates for hardware and software, put 0%.

ADDING SOCIAL COST

The model was originally envisioned to capture social costs. The FAA Sys Outages worksheet is the first cut at providing that capability. The design intent is to enter the likelihood of component failure per E6 hours of operation (i.e., probability of failure per 1,000,000 hours of operation) *after* the system has reached the end of its physical or operational life. The purpose was to allow the model to calculate the social cost of an outage in the case in which a system had not been replaced far beyond its required refresh age at a point where replacement parts and maintenance support cannot be obtained. Since the FAA is extremely cautious about air safety, it is only under these circumstances that it is envisioned that a system outage could occur that would have an impact on air travel operations.

Failure rate per E6 hours can be estimated for any system, given its current age, its current outage rate (obtainable from maintenance records), and a Weibull obsolescence function. It is mainly a matter of footwork within the FAA to obtain the maintenance records and apply the obsolescence function.

The worksheet design allows for the varying degrees of outage in the overlapping functionality of the air traffic control network, as designed by the FAA. Examples of outages range from loss of primary communications in a sector, loss of secondary communications in a sector, loss of radar view in a sector, or loss of air-ground communications in a sector all the way up to a loss of all radar views or air-ground communications in a center. Several very good models are available

for estimating the time associated with each adverse event, as well as the costs of lost aircraft direct operating cost (ADOC) and passenger value of time; two are discussed here as possible candidates for incorporation.

System Outage Disruption Model

The FAA Office of System Architecture and Investment Analysis (ASD) owns and operates a model called the System Outage Disruption Model (SODM) that estimates annual savings to passengers and airlines based on reducing the failures per E6 of FAA air traffic control systems. The SODM User s Guide states the following:

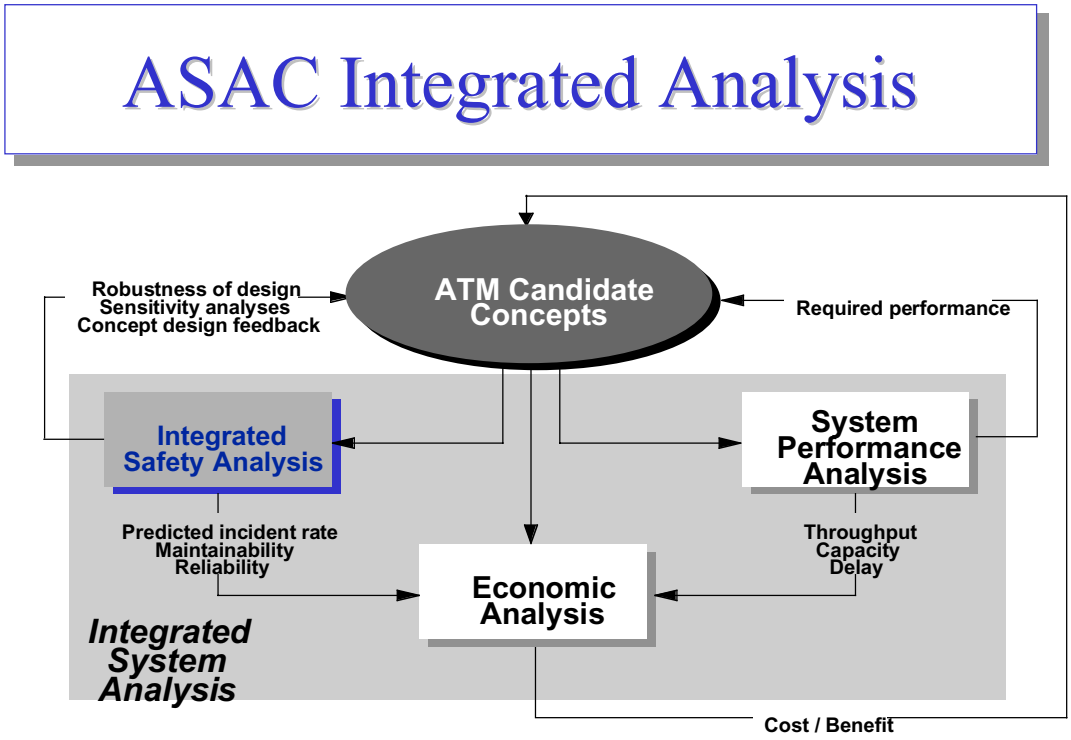
> Disruption is modeled in terms of the outage-induced reduction in capacity, the duration of the outage, and the arrival demand at the affected terminals. The model generates a probability distribution of the amount of disruption due to system component outages relative to the baseline year, and can be used to estimate the impact of alternative NAS system designs and implementation schedules.

Since SODM is reliability-based, it seems like a natural complement to our model, although the existing FAA Sys Outages interface would have to be redesigned, because SODM is not based on individual outages but rather on system reliability. Also, SODM brings its own reliability estimates; those developed in the Technology Refreshment Cost Estimating and Planning Model would not be needed.

Aviation System Analysis Capability

Alternatively, LMI has a Web-based model called the Aviation System Analysis Capability (ASAC), to which LMI has added the System Safety and Operational Availability Model. The Safety Model works with ASAC to estimate delays in the air traffic system based on missing functionality. A model flow diagram is shown in Figure 4-1.

Figure 4-1. Flow Diagram of ASAC and the Integrated System Safety Model



Together, ASAC and the Safety model are capable of projecting traffic load in any given year up to 2027, constraining demand based on long-term acquisition projections, calculating baseline delays and cancellations, then including reliability factors, back-up systems, human responses, and probabilistically generating incident rates and the costs of those incidents. An example for a radar outage is shown in Figure 4-2.

Figure 4-2. Radar Outage Example in System Safety Integrated Model

State of Function	State Definition	System Impact	Simulation Impact
Fully Operational	Primary Radar indication of all aircraft in TRACON; Secondary Radar data available for all aircraft equipped with functioning Transponders	Position estimate of all aircraft in TRACON presented to controller is sufficient to control normal approach	Normal position errors and flight paths for all aircraft
Primary Only	Loss of Secondary Radar	Position estimate of all aircraft in TRACON presented to controller is limited to accuracy provided by Primary Radar	Vertical position error of all aircraft with functioning Transponders increased from normal to reflect loss of Secondary Radar information
Secondary Only	Loss of Primary Radar	Position estimate available for only aircraft with functioning Transponders	Position error of all aircraft without functioning Transponder increased from normal to reflect loss of Primary Radar
Failed	Primary and Secondary Radar not functioning	Aircraft permitted to land but under contingency procedures	Position error of all aircraft increased from normal to reflect loss of Primary and Secondary Radar information

Forecasting traffic up to 2027 may be desirable given the long planning horizon of the Technology Refreshment Cost Estimating and Planning Model. ASAC is capable of estimating the delays associated with lack of radar in a sector or center, which comports nicely with the systems already captured in this model. ASAC reports the cost of a single adverse event, while SODM reports the probability-weighted cost of reliability degradation. Each model has desirable features, and either could be added to the Technology Refreshment Cost Estimating and Planning Model in a future task.

SOFTWARE ESTIMATION

Our model includes a placeholder for software lines of code (SLOC). There are a number of off-the-shelf models available that estimate SLOC and SLOC costs; the FAA has owned and maintained a number of them. In order to make cost estimates more closely approximate actual acquisition cost, it is recommended that SLOC costs be included in the model, both for new code and for re-using code. Many refresh projects will require some re-writing of code, whether in porting the application software, easing operating system transition, or providing additional functionality.

The recommended approach for incorporating SLOC in the model would be to identify the average cost per thousand lines of code of writing new code and re-using old code, for several popular languages, and enter each on a line in the Parts worksheet. Every major software writing effort requires recurring support as bugs are discovered long after shakedown; the software support cost should be recorded as a maintenance cost. If extra hardware is required to test and re-write code, that expense should be included as a cost under the spreadsheet s hardware columns. If it is FAA s experience that code eventually becomes stable and needs no more maintenance, that stability should be recorded as a second maintenance stage in place of the bridge contract provisions.

GRADUATED REFRESHMENT

Graduated refreshment refers to a cycle of replacing COTS components within a larger system. For example, if you were to refresh your home computer, you might decide that it would be easier on your pocketbook if you bought a new CPU one year, a new monitor the following year, and a new printer the year after that. One interpretation is that if you are constantly buying new components, the system never becomes completely obsolete. Many FAA systems are so extensive that graduated refreshment is virtually a requirement. The thoroughness of shakedown testing and fault isolation for new critical components prohibits simultaneous replacement of linked systems. And in many FAA facilities, there isn t enough room in the equipment room to replace more than one system at a time.

Contrary to the idea of graduated replacement, our model was constructed to cost the replacement of all of a system s COTS components simultaneously. However,

by creating sub-programs within each FAA program, it is possible to model graduated replacement in the current model. For instance, to model graduated replacement of the VSCS program, a user could create an FAA program called, VSCS control computer with the current refreshment project, and then create a second FAA program, VSCS PBX, with first refresh in 7 years. Each sub-program would occupy a separate column in the FAA Programs worksheet. Under the current model, the user would have to run each program separately through the model but could copy the output cost streams from each run from the Output Chart worksheet and combine the streams for each sub-program on a separate spreadsheet. Figure 4-3 through 4-8 illustrate this process.

Figure 4-3. Step 1 Enter Sub-programs on FAA Systems Worksheet

					○ new1	○ new2
					enter a new program name in the s	
COTS/CAS system		VSCS	HOCSR	PAMRI	VSCS control computer	VSCS PBX
Computers - mainframes		0	44	0	0	0
Computers - Unix		0	0	0	0	0
PC servers -high		0	0	0	0	0
PC servers - medium		132	0	0	132	0
PC servers - low		0	0	44	0	0
Basic PC		0	22	0	0	0
	computer peripherals in general	0	0	0	0	0
	computer disk drive	0	0	0	0	0
	computer head	0	0	0	0	0
	computer hard drive	0	0	0	0	0
	computer memory	0	0	0	0	0
	computer track ball	0	0	0	0	0
		0	0	0	0	0

Figure 4-4. Step 2 Enter Desired Parameters for the First Sub-program on the GUI Worksheet

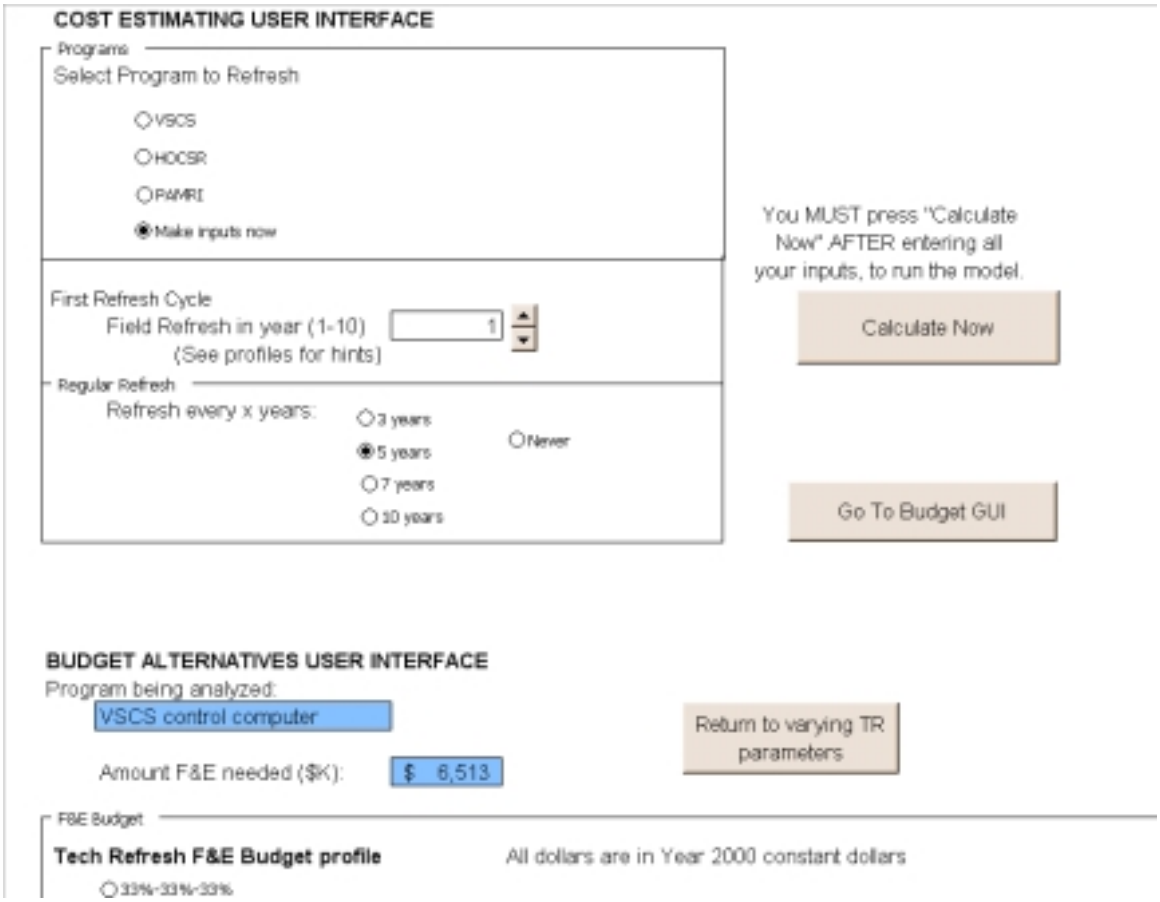


Figure 4-5. Step 3 Copy the Output Numbers from the First Program Run to Another Worksheet

year	1	2	3	4	5	6	7	8	9	10
Maint	$ 288,876	$288,876	$ 2,000	$ 2,000	$ 2,000	$ 44,200	$ 44,200	$2,000	$ 2,000	$ 2,000
F&E	$5,861,449	$861,272	$ -	$ -	$ -	$ 586,449	$ 651,272	$ -	$ -	$ -

Figure 4-6. Step 4 Run the Second Sub-program

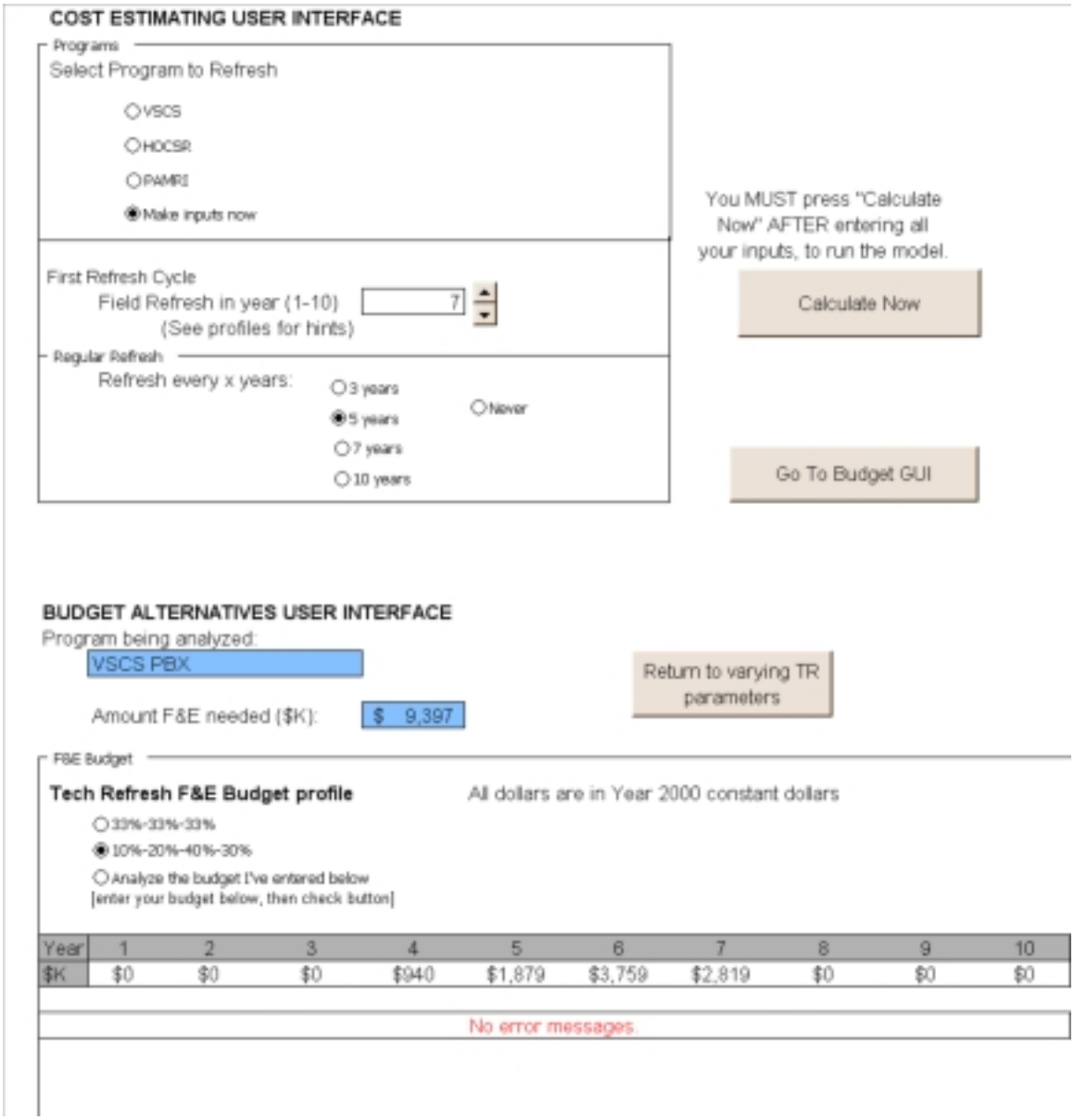
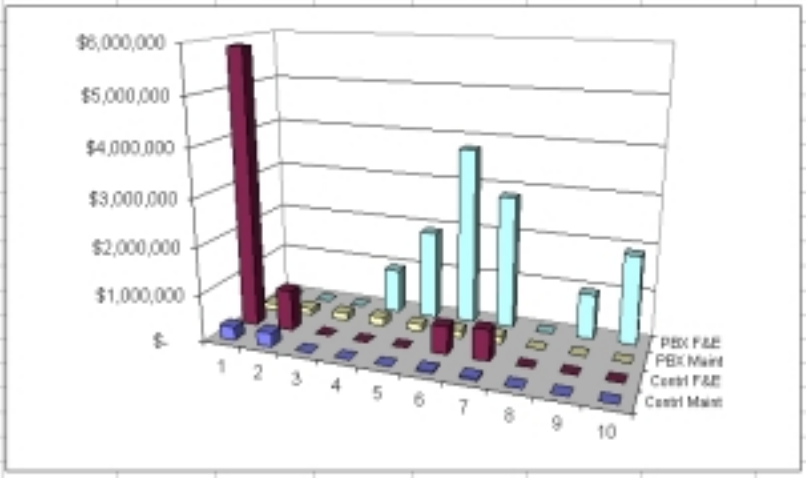


Figure 4-7. Step 5 Copy the Output Numbers from the Second Program Run to Another Worksheet

year	1	2	3	4	5	6	7	8	9	10
Maint	$ 137,214	$137,214	$152,805	$176,505	$ 148,812	$ 150,410	$ 150,275	$2,000	$ 2,000	$ 2,000
F&E	$ -	$ -	$ -	$939,686	$1,879,372	$3,758,744	$2,819,058	$ -	$939,686	$1,879,372

Figure 4-8. Step 6 When You Are Done Running Sub-programs, Combine Your Data on the New Worksheet to Tell the Proper Story

year	1	2	3	4	5	6	7	8	9	10
Contrl Maint	$ 288,876	$288,876	$ 2,000	$ 2,000	$ 2,000	$ 44,200	$ 44,200	$2,000	$ 2,000	$ 2,000
Contrl F&E	$5,861,449	$861,272	$ -	$ -	$ -	$ 586,449	$ 651,272	$ -	$ -	$ -
PBX Maint	$ 137,214	$137,214	$152,805	$176,505	$ 148,812	$ 150,410	$ 150,275	$2,000	$ 2,000	$ 2,000
PBX F&E	$ -	$ -	$ -	$939,686	$1,879,372	$3,758,744	$2,819,058	$ -	$939,686	$1,879,372



AUTOMATIC OPTIMIZER

An obvious and easy addition to the existing model would be the inclusion of an automatic optimizer. Right now the cost analyst must look at the cost profiles, try out a combination of parameters, and try different model runs to home in on the most economical combination of first refresh and repeat refresh. The analyst can see which direction the repeat costs are going and so should be able to discover the most economical combination in four model runs or fewer.

The model could be programmed to run every possible combination of first refresh and repeat refresh (41 combinations), output the results to a separate table, and rank them by lowest cost (both present value and constant dollars). To do this, one would write a macro setting key parameters in turn, run the Horizons macro over and over again, and then add some ending material to rank the alternatives by sums.

Since the model does not at present include existing age of components, repeat and first refresh optimums will agree within plus or minus 1 year. At this time, perhaps a more useful option would be to ask the model to optimize over repeat refresh options, for a total of 4 runs. The macro would be written the same way, allowing the user to input a first refresh year that takes into account the existing age of components. (For example, if components are 4 years old and first refresh is being considered for 3 years from now, the user can input 7 in first refresh, and disregard the output for years 1 through 4, since year 4 is now.)

FAA-WIDE TECH REFRESH ESTIMATING

Using data gathered from FAA programs, extensive DoD databases, and commercial practices and information, we created a number of algorithms that make up the engine of the tech refresh cost planning model. We have converted these algorithms into a cost-estimating relationship (CER) that can be used to estimate FAA-wide tech refresh cost, using the mean age of FAA systems in years and total hardware acquisition cost:

$$\text{Cost} = (1.6878347 - 0.195276 * \text{years} + 0.0339833 * \text{years}^2) * \text{ACQ}$$

ACQ = original hardware and operating system acquisition cost in constant dollars.

This is the CER we would apply to FAA data to derive an FAA-wide tech refresh cost estimate. This CER applies only to hardware and bundled commercially available operating systems; it does not address application software.

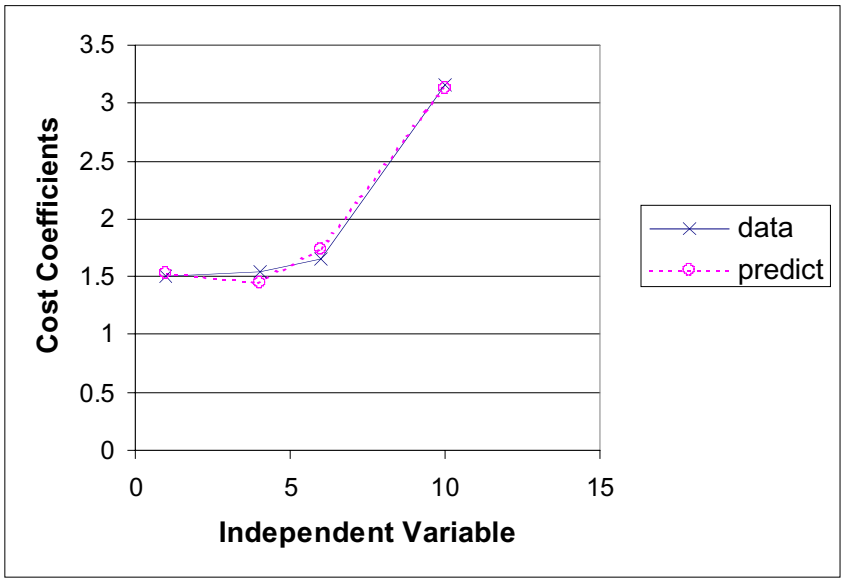
We converted from algorithm to CER in the following steps. The refresh estimating algorithm was translated from a logical statement into four equations, graduated by year. The cost associated with each stage of life was translated into terms of hardware and software acquisition cost, using mean values found in the Parts database. Table 4-1 contains representative values.

Table 4-1. Refresh Cost as a Ratio of Acquisition Cost

Year	Refresh cost as a ratio of acquisition cost
1	1.50
4	1.55
6	1.65
10	3.15

We ran a series of regressions on these numbers and found the best fit from the equation given above. Figure 4-9 compares the algorithm's predictions with actual data.

Figure 4-9. Comparison of Algorithm Predictions with Data



Chapter 5
Data Sources

BASIC ESTIMATING RELATION

In constructing this model, we first talked to several FAA program managers directing the technology refreshment for the Voice Switching Communication System (VSCS), the Peripheral Adapter Module Replacement Item (PAMRI), the Airport Surveillance Radar (ASR-9), and the Standard Terminal Automation Replacement System (STARS). We reviewed documentation from the FAA Host and Oceanic Computer System Replacement (HOCSR), the Display System Replacement (DSR), and the STARS programs tech refresh efforts, as well as the AUA Technology Refresh Planning Guide [49]. The same story line emerged over and over, leading to the basic estimating relations used in this model. We then researched tech refresh in the commercial sector to fill in parameters on lifetimes and maintenance costs that we did not get from FAA experiences.

This chapter will review the facts we learned from FAA programs history and then identify the source or sources of each parameter in the Parts worksheet. The sources are listed in the References section following this chapter.

FAA PROGRAM STORIES

According to the AUA Technology Refresh Planning Guide, any product offered to the general public for sale is COTS. This broad guidance can encompass some very specific products. Computer products in particular are often introduced for sale to a very small market. For instance, anyone willing to spend $1 million can buy an airline scheduling software tool from Decision Support Technologies, so arguably it is a COTS product, though the dozen users of that software would probably not put it in the same category as Windows 95. Many COTS components are manufactured for a small client base. For example, a firm specializing in a particular area (e.g., telecommunications) finds it easy to modify a design used for a previous customer and offer it to the FAA. If the client base of the new product is limited to the FAA, probably the product is modi fied COTS.

VSCS

VSCS is like a local exchange telephone office for air traffic controllers. It is an all-analog switching system for controllers voice communications with aircraft and with other controllers. It provides

- communication with other controllers within an Air Route Traffic Control Center (ARTCC) (ground-ground intercom);
- communication with other controllers in other facilities such as towers, Terminal Radar Approach Controls (TRACONs), and other ARTCCs (ground-ground interphone);
- interfaces with remote radio equipment over several hundred miles, enabling the air traffic controller to speak with pilots over radio frequencies (air-ground communication); and
- back-up recordings of controller-pilot conversations, per legal requirements.

VSCS is fielded at the 21 ARTCCs in the contiguous United States and Alaska. VSCS resides on several processors, so if one processor goes down, a central controller instructs the other processors to take on more of the load. VSCS also consists of redundant LANs, two each for ground-to-ground and air-to-ground communications. If a VSCS processor goes down, the air traffic controller does not notice a difference in performance. If a LAN goes down, there is another LAN. If the controller computer goes down, there is a back-up system for air-to-ground and a separate back-up system for ground-to-ground communications. However, the back-up systems are no more reliable than VSCS. If both the VSCS air-to-ground system and the back-up air-to-ground system go down, the result would be a severe impact on air traffic. Controllers would be unable to talk to aircraft in an area covering hundreds of square miles. Aircraft would have to rely on onboard systems for separation until another way was found to communicate between the air and the ground. Every U.S. aircraft has several radios, so there are many ways to resolve this situation, but huge delays would be incurred across the United States, and it is possible that there could be a mid-air collision.

During procurement, there was a choice between three competing computer systems with slightly different architectural approaches for the VSCS controller computer, so the controller computer can be considered modified COTS. The controller computer is one component out of many in this system. The controller computers used were Tandem computers with a Tandem Pascal proprietary operating system. It is this component that needs earliest refresh.

VSCS was delivered between January 1994 and June 1996, and it reached operational readiness deployment between June 1995 and February 1997. The Tandem computers reached the end of their market life in 1995. During procurement, the

FAA paid for 10 years worth of Tandem operating system license options, in paid-up-front fees (PUFFs.) In the summer of 1999, the original equipment manufacturer (OEM) announced that hardware support would no longer be available at any price in December 2000 (roughly 5_ years after the end of market life.) The hardware support cost had been increasing, so that in mid-2000 the cost of maintaining a single site was $22,000 per month, or over a quarter of a million dollars per year, for hardware maintenance. Software support is no longer available from the OEM and is being supplied by a dedicated contractor. The PUFF licenses on the operating system begin to expire in 2003; new license fees, exclusive of any maintenance at all, will cost $1.26 million in 2003. The VSCS-VTAB Technology Evolution Plan (V2TEP) program office estimates that license fees on the operating system have been escalating at 20 percent per year. During this time, the FAA has been unable to obtain funding authorization to refresh what Congress and OMB see as a new system.

The Tandem computers can be considered COTS; the current operating system is proprietary. Because of efforts of the project office to recode the application for transport, under the next tech refresh the application software could reasonably be hosted on virtually any new operating system, including a COTS-based operating system.

Identification of the VSCS components listed for refreshment is based on conversations with the program manager and VSCS web sites. We did not have an architectural diagram of the VSCS program. There is room for improvement in our component-by-component itemization of the program, and we hope that by distributing this model to offices within the FAA that more accurate entries will be made. Further, graduated refreshment is being pursued in the VSCS program. The control computers are up for immediate replacement, and the 20-by-20 PBXs are slated for replacement in about 7 years. In our model, all refreshable components are assumed to be replaced simultaneously.

It is possible to model graduated replacement in the current model, by creating sub-programs under each FAA program. For instance, to model graduated replacement of the VSCS program, the user would create an FAA program named VSCS control computer with the current refreshment project, and would then create a second FAA program, VSCS PBX, with first refresh in 7 years. Each sub-program would occupy a separate column in the FAA Systems worksheet. Under the current model, the user would have to run each program separately through the model but could copy the output cost streams from each run from the Output Chart worksheet and combine the streams for each sub-program on a separate spreadsheet. See Chapter 4, Expanding the Model, for more details and diagrams.

HOCSR

HOCSR provides for the replacement of the Host Computer System, the Oceanic Display and Planning System, the Off-Shore Flight Data Processing System, and

other related equipment used to provide en route and oceanic air traffic control for the United States and large areas of international waters. The main component of the Host Computer System is the IBM 3083 mainframe and its operating system, the 370 instruction set.[46]

The Host Computer System is the heart of U.S. air traffic control. The Host computer receives all radar data about aircraft position; it receives data tags for each aircraft indicating its name, identity, where it is going, what frequency it is talking on, and its altitude, speed, path, and proximity to other aircraft. The Host computer at each ARTCC gathers this information for every aircraft in its airspace and sends out an annotated radar-like picture to air traffic controllers in the local ARTCC, in TRACONS, and in towers. The Host exchanges this information with other Hosts at other ARTCCs. Without the Host or its equivalent, controllers would be plotting aircraft position on pieces of paper, and the volume of air traffic we enjoy today would not be possible. The Host has a back-up, the Enhanced Direct Access Radar Channel (EDARC). Both have been known to fail. Usually a Host failure is limited to one scope, but in St. Louis in September 1997, both systems went down and there was no radar assistance to controllers over an area covering several hundred square miles. There was no automated flight data exchange, including all flights in and out of St. Louis Lambert airport.

The HOCSR program is structured in 4 phases. Phase 1 builds a 370 instruction set emulator and ports the application to new mainframe hardware, fielding the emulator between the new operating system and the application software. Phase 2 re-writes the application software to run on the new operating system without an emulator. Phases 3 and 4 of the program provide limited functionality improvements to the application that we do not consider part of Tech Refresh.

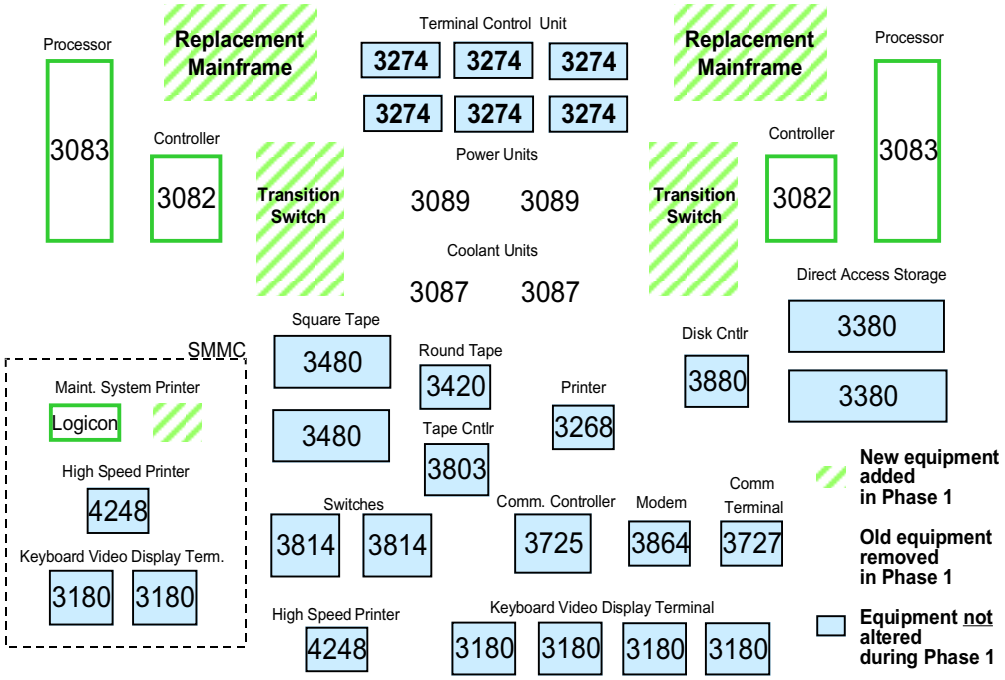
The experience of the HOCSR program illustrates one of the key issues in performing tech refresh: that many legacy applications have hooks into a particular operating system, and thus porting them from the operating system is troublesome. Unfortunately, when it becomes necessary to replace hardware because of component failure, often the hardware on the market is incapable of running an older operating system, so either an emulator must be built or the application must be re-written. Because of the critical nature of the Host Computer System, the HOCSR program finds it necessary to do both. This also explains the popularity of the computer languages C and C++, which can run on virtually any operating system, without hooks.

It should be noted that the Host Computer System hardware, the 3083 mainframe, is being replaced with another mainframe, the G3, and that mainframes are no longer available on the market. The G3 was last manufactured in 1996, and the FAA bought all the G3s available as an end of lifetime buy. Mainframe operating system generations are more revolutionary than those of desktops and servers. Each mainframe has its own operating system, with far less in common with previous generations than one would find in desktop and server markets. Future rehosts of the Host Computer System will likely be to smaller computers, though

the smaller desktop computers of today provide much of the functionality of the outgoing 3083. This situation makes tech refresh cost estimating more difficult for this program.

The components listed under FAA Systems for refreshing are based on the HOCSR Acquisition Program Baseline (APB), briefing packages supplied by the program office, and publicly available documents we found on FAA web sites. Our architectural diagram of the program was taken from a powerpoint presentation; the diagram is reproduced in Figure 5-1[55]. However, we did not have the benefit of program office review in our estimation. Again, there is room for improvement in our component-by-component itemization of the program, and we hope that the FAA will make more accurate entries in this model.

Figure 5-1. Graphic from a PowerPoint Presentation Showing HOCSR Phase 1 Components Replacement



Source: Reference 55.

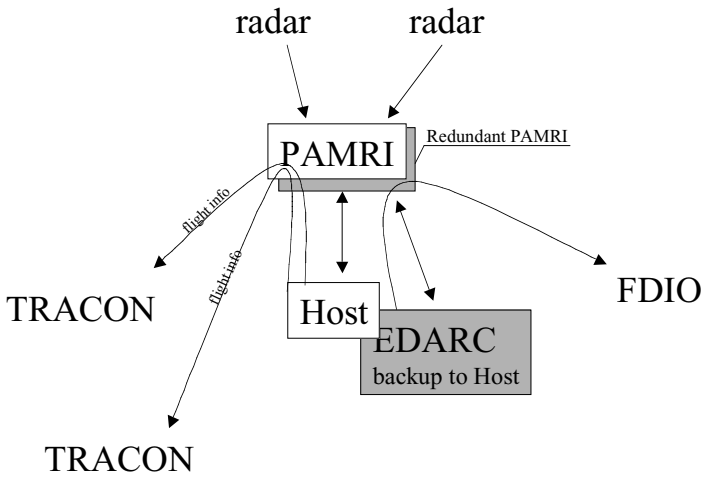
PAMRI

The Peripheral Adapter Module (PAM) was a separate processor that converted radar data into a format readable by the Host Computer System. When IBM began building replacement architecture for the FAA in the Advanced Automation System (AAS) acquisition of the early 1990s, the PAM was already a maintenance problem. IBM replaced the PAM with the PAM Replacement Item (PAMRI), a custom-designed 286 desktop box with an IBM-proprietary operating system. The PAMRI was intended to last about 3 years, since it would be replaced when the

Automated Communication Computer Complex of AAS was fielded. However, the entire AAS program was shut down by Congress not long after this replacement. FAA bought 10 years worth of PUFF maintenance options from IBM. The first 10-year PUFF expires in December 2000; the last site-specific PUFF expires in June 2000. The IBM installation that was under contract to provide maintenance has been sold to Lockheed Martin; the fact that, contractually, Lockheed Martin is not obligated to provide the pre-arranged maintenance, reveals another hazard of maintaining COTS equipment.

The PAMRI is a vital part of enroute and terminal air traffic control. Figure 5-2 demonstrates how the PAMRI is an integral part of delivering radar to the Host. Solid lines show information transfers between the PAMRI, Host, EDARC, radar with varying data protocols, and the Flight Data Input-Output Computer (FDIO). If the PAMRI and its backup were to fail, the outcome would be the same as if the Host Computer System failed; the entire ARTCC would be without radar use.

Figure 5-2. PAMRI in Relation to Radar Input and the Host Computer System



Note: Figure based on conversations with FAA personnel.

Servers are being investigated as replacements for the PAMRI. When we talked to the program manager, the program team was still developing requirements and specifications. Our estimate of the COTS components of the PAMRI architecture is based on a conversation with the program manager and a 1997 General Accounting Office (GAO) report [114]. We identified the need for a mid-line server and 10 interface-specific firmware cards. The program manager also identified the need for application software to be re-coded, which should be a large part of the conversion to COTS, but we were unable to obtain any estimates of the magnitude of this job. Therefore our estimate for PAMRI refresh will fall well short of needed funds for the program because it does not encompass application re-coding.

Since the model is meant to apply to COTS hardware only, none of the model s estimates include any special re-coding needs for converting legacy applications.

STARS

STARS replaces the radar displays for terminal area controllers. Existing (legacy) radar displays are called Automated Radar Terminal System (ARTS) and are identified by their radar display processor s generation: IIA, IIE, IIIA, and IIIE. The four generations of ARTS radar display processors each generate a different data format. Standardization of terminal area radar displays has been a desire of the FAA for over a decade, and computer technology has progressed so that now COTS technology can fill the need.

Interviews with FAA personnel revealed that the current plan is to transport the terminal radar display application from Raytheon and Egelund processors and operating systems to an open system. The first open systems processors, Suns, are in the process of being fielded. Because the procurement numbers in the hundreds, the deployment schedule will be lengthy and will overlap more than one market generation of processors. The program team already has had to write a small program to help transport the application to the HyperSparc operating system; Ultra Sparcs had been used in test and evaluation.

Instead of buying thousands of today s processors, to freeze the generation of the procurement, the STARS team is buying with the market; so in 2 years there will be more than one generation of processors handling the FAA s terminal radar displays. This practice has the disadvantage of having varying equipment and varying capabilities in the field, and it leads to a need for configuration management. Vendor-provided configuration management is under consideration.

STARS is also one of the programs used to derive the CERs of the AUA Technology Planning Guide.

ASR-9

The ASR-9 is one of a family of terminal area radars that track aircraft within 60˚nautical miles of an airport. ASR-4, -5s, -6s, and -7s are primary-only radars that feature tube technology; the ASR-11 is an advanced digital, combined primary, secondary, and weather radar being procured jointly with the Department of Defense. The ASR-9 is solid-state and it is a step between older radar technologies and newer, advanced, integrated radar. The ASR-9 is still being fielded, but fielded units are already experiencing obsolescence and intensive support requirements. ASR support costs are increasing dramatically, and the FAA finds itself in the position of robbing its modernization budget to cover support costs. Robbing the modernization budget means putting off installation of lower-maintenance components, putting the FAA in a dilemma that it has been unable to resolve. Part of the problem can be traced to the FAA s practice of trickle-down fielding; when a new radar with enhanced capabilities is acquired, the FAA fields it to a busy terminal

facility and moves the older radar already serving that facility to a smaller terminal. Thus, as the ASR-7 and -8 were fielded, many ASR-4s, -5s, and -6s remained in service. This practice provides greater safety and fewer delays for the traveling public at more airports but leaves the FAA with a greater support requirement. The ASR-11 fielding is projected to move the trickle down out far enough that all ASR-4s, -5s, and -6s will be taken out of service.

The ASR-9 is a stopgap, filling a need for more modern radar during a delay in the ASR-11 procurement. Some of the ASR-9 components are older technology. The FAA is currently studying the extent of the ASR-9 maintenance problem; a preliminary finding includes that the ASR-9 relies on maintenance-intensive timing clocks. The FAA will identify the highest-maintenance-consuming line replaceable units (LRUs) of the ASR-9, such as the clocks, and replace them. In the case of the clock, a COTS GPS receiver could perform the same function with less support.

In addition to the functional improvements of the ASR-9, the ASR-9 is the first of the ASR family to incorporate remote maintenance monitoring (RMM). RMM is being fielded with all the FAA s new procurements, to reduce maintenance by supporting condition-based maintenance. RMM basically monitors key status parameters of the radar and transmits them back to the local maintenance facility. Other functional improvements of the ASR-9 include a moving target detector to track low-altitude aircraft precisely, as well as a weather channel to display the National Weather Service s six levels of weather, any two at a time [120].

Lessons Learned

From FAA experiences and references [8, 12, 13, 17, 42, 44, 47, 49, 53, 84, 89, 92, 105, 106, 112, 126, 127], we derived a series of COTS lessons learned.

- COTS obsolescence is driven both by hardware generations and by software generations.
- Porting an application to the next-generation operating system gets more complicated the longer the delay in moving.
- Porting from proprietary operating systems is more expensive than porting from COTS operating systems.
- Porting legacy applications to new operating systems is extremely difficult because of coding hooks to the operating system. It is often necessary to re-code the application in a more modern code such as C before it can be transported. With or without re-coding, sometimes an emulator must be created to ease the transition.
- New generations of computer and telecommunications equipment are issued approximately every 18 months. Operating system upgrades occur

about every 2 hardware generations. Operating system revolutions occur every 10 years.

- The length of FAA s acquisition and fielding cycles results in obsolescence of some fraction of COTS components during fielding. This situation is virtually unavoidable.
- Graduated refreshment within a system is highly desirable, to even out procurement cycles and minimize system disruption.
- OEM maintenance even that contracted for up-front for a duration of 10˚years can be terminated by the OEM before the FAA is ready to replace the system.
- Some COTS hardware can reliably perform far past its expected physical lifetime. Generally this hardware is replaced when the application requires expanded capabilities.
- Sacrificing modernization procurement in order to meet support funding needs increases next year s support funding need.

PARTS DATABASE TABLE OF SOURCES

Tables 5-1 through 5-17 list the cells on the Parts data sheet and the number of the reference or references used to derive the cell s value or estimating relationship. Each table covers approximately a single column. The tables are arranged as if one were reading the Parts database by columns, from top to bottom. The references are briefly annotated.

As the Parts database was developed, it was possible to derive estimating relations from already collected data and use them to fill in missing parameters. The use of previously collected data as an estimating basis is denoted by cell reference.

Note: if a cell reference is omitted from Tables 5-1 through 5-17, then there is no value in that cell. For example, many of the radar parameters in the Parts database are blank because we did not expend effort to find them. If the cell is included in the first column but no reference is listed, then the estimate given in the database is based on best judgment only and there is no citable source. If the reference is zero, then we did not find a cost in that area, and correspondingly there is a zero in the Parts database under that value.

Table 5-1. Sources for Parts Worksheet Entries: Reliability and Initial Hardware Cost

Cell	Note	Reference
E11:E27	Reliability	41, 43
F6	Mainframe	3, 30, 32
F7	Unix	86, 38, 71, 109
F8	PC server	25, 38, 71, 96
F9	PC server	25, 38, 71, 96
F10	PC server	25, 38, 71, 96
F11	PC	25, 39, 72
F20	Big-screen display	70
F26	Radar	50
F28	Fiber LAN	2
F29	IP router	18, 23, 24, 25, 26
F30	Crypto router	23, 24, 25, 26
F31	Mulitprotocol router	23, 24, 25, 26
F32	Voice data router	23, 24, 25, 26
F33	8-slot router	23, 24, 25, 26
F34	8-slot with tunnel switch	23, 24, 25, 26
F35	Ethernet router	18, 23, 24, 25, 26
F36	Cisco 13-slot router	23, 24, 25, 26
F37	KVM	22, 97, 118
F38	KVM	22, 97, 118
F39:F45	Hub	19, 20, 21, 26, 40, 67, 128
F47	RAID	67
F49	RAID	77, 115, 116, 117
F52	Rack	73, 108
F53	Rack	73, 108
F54	Rack	73, 108
F55	UPS	74, 29
F56	UPS	74, 29
F59	Printer	28, 27
F66	PBX	82
F67	Ringing generator	90, 99
F71	TIM, analog	2, 90, 124
F72	TIM, digital	2, 90, 124
F73	Signal analyzer, analog	90, 124, 5
F74	Signal analyzer, digital	90, 124, 5
F75	Oscilloscope, analog	33, 35, 90
F76	Oscilloscope, digital	34, 36, 90
F77	A/D converter, 4-wire	90, 111, 124
F78	Liftcart	76, 90, 124
F111, F112	Firmware	1, 83, 91, 129
F115	Sunsoft	77

Table 5-2. Sources for Parts Worksheet Entries: Initial Hardware Support Cost

Cell	Note	Reference
G6	Mainframe	3, 7, 32, 75
G7	Unix	86
G8	PC server	119
G9	PC server	119
G10	PC server	119
G11	PC	119
G20	Big-screen display	70
G28	Fiber LAN	2, 81, 101, 102
G29:G38	Router	Derived from computer support average
G39:G45	Hub	100
G47	RAID	43
G49	RAID	43
G52	Rack	44, 73, 108
G53	Rack	44, 73, 108
G54	Rack	44, 73, 108
G55	UPS	6
G56	UPS	6
G59	Printer	Derived from computer support average
G66	PBX	Derived from computer support average
G67	Ringing generator	Derived from computer support average
G71:G77	Telecom test equipment	33, 34, 35, 36, computer support average
G78	Liftcart	
G111,G112	Firmware	1, 83, 91, 129, derived from G28

Table 5-3. Sources for Parts Worksheet Entries: Initial Bridge Hardware Support Cost

Cell	Note	Reference
H6	Mainframe	3, 7, 32, 75
H7	Unix	86
H8	PC server	119
H9	PC server	119
H10	PC server	119
H11	PC	119
H20	Big-screen display	70
H26	Radar	
H28	Fiber LAN	2, 6, 81
H29:H38	Router	65, 119, derived from computer support average
H39:H45	Hub	65, 119, derived from computer support average
H47	RAID	Derived from computer support average
H49	RAID	Derived from computer support average
H52	Rack	44, 73, 108
H53	Rack	44, 73, 108
H54	Rack	44, 73, 108
H55	UPS	6
H56	UPS	6
H59	Printer	Derived from computer support average
H66	PBX	45
H71:H77	Telecom test equipment	104
H78	Liftcart	76
H111:H112	Firmware	Derived from computer support average
H115	Sunsoft	0

Table 5-4. Sources for Parts Worksheet Entries: Bridge Hardware Support Cost Growth Rate

Cell	Note	Reference
I6:I25, I29:I49	Computer equipment	78, 65
I28	Fiber LAN	101, 102
I52:I56	Rack, UPS	73, 108
I59	Printer	78, 65
I66,I67,I71:I78	PBX, Test equipment	104
I110:I111	Firmware	0; LRU needsreplacement only

Table 5-5. Sources for Parts Worksheet Entries: Initial Software Cost

Cell	Note	Reference
J6	Mainframe	3, 7, 30, 32
J7	Unix	77
J8	PC server	93, 94, 95, 96
J9	PC server	93, 94, 95, 96
J10	PC server	93, 94, 95, 96
J11	PC	Windows price
J20	Big-screen display	70
J26	Radar	
J28	Fiber LAN	100
J29:J38	Router	100
J39:J45	Hub	100
J47	RAID	77
J49	RAID	77
J52	Rack	44, 73, 108
J53	Rack	44, 73, 108
J54	Rack	44, 73, 108
J55	UPS	0
J56	UPS	0
J59	Printer	0
J66	PBX	Derived from computer support average
J71:J77	Telecom test equipment	Derived from computer support average
J78	Liftcart	0
J111:J112	Firmware	0

Table 5-6. Sources for Parts Worksheet Entries: Initial Software Support Cost

Cell	Note	Reference
K6	Mainframe	3, 7, 32, 75, 103
K7	Unix	86
K8	PC server	96, 98, 103, 119
K9	PC server	96, 98, 103
K10	PC server	96, 98, 103
K11	PC	119
K20	Big-screen display	0
K26	Radar	
K28	Fiber LAN	2, 101, 102
K29:K38	Router	0
K39:K45	Hub	0
K47	RAID	0
K49	RAID	0
K52	Rack	44, 73, 108
K53	Rack	44, 73, 108
K54	Rack	44, 73, 108
K55	UPS	0
K56	UPS	0
K59	Printer	0
K66	PBX	Derived from computer support average
K71:K77	Telecom test equipment	104, derived from computer support average
K78	Liftcart	0
K111:K112	Firmware	0

Table 5-7. Sources for Parts Worksheet Entries: Second Software Support Cost

Cell	Note	Reference
L6	Mainframe	54, 55, 56
L7	Unix	87, 65
L8	PC server	65
L9	PC server	65
L10	PC server	65
L11	PC	65
L20	Big-screen display	0
L26	Radar	
L28	Fiber LAN	2, 101, 102
L29:L38	Router	0
L39:L45	Hub	0
L47	RAID	
L49	RAID	
L52	Rack	44, 73, 108
L53	Rack	44, 73, 108
L54	Rack	44, 73, 108
L55	UPS	0
L56	UPS	0
L59	Printer	0
L66	PBX	K66
L71:L77	Telecom test equipment	K71:K77
L78	Liftcart	K78
L111:L112	Firmware	0

Table 5-8. Sources for Parts Worksheet Entries: Second Software Support Cost Growth Rate

Cell	Note	Reference
M6:M25	Computer equipment	78, 65
M28	Fiber LAN	101, 102
M29:M38	Router	65, 68, 78
M37:M38	KVM	65, 78
M39:M45	Hub	65, 78
M47:M50	RAIDS	67
M51:M54	Rack	73, 108
M55:M56	UPS	6
M59	Printer	65, 78
M66	PBX	79, 82, 104
M67	Ringing generator	104, 111
M71:M78	Test equipment	104
M110:M111	Firmware	0

Table 5-9. Sources for Parts Worksheet Entries: Program Cost Percentage

Cell	Note	Reference
N6:N25	Computer equipment	Based on percentage from PRICE-H software estimation for computer equipment
N26:N45	Telecom hardware	Reduced percentage based on limited replacement scheme
N47,N49	RAID	Based on percentage from PRICE-H software estimation for computer equipment
N52:N59	Hardware	Reduced percentage based on limited software interaction or replacement scheme
N66	PBX	Percentage increased based on increased software/protocol integration needs of hardware
N67	Ringing generator	Percentage increased based on increased software/protocol integration needs of hardware
N71:N77	Telecom test equipment	Based on percentage from PRICE-H software estimation for computer equipment
N78	Liftcart	Reduced percentage based on limited software interaction or replacement scheme
N111:N112	Firmware	Percentage increased based on increased software/protocol integration needs of hardware

Table 5-10. Sources for Parts Worksheet Entries: End of Hardware Market Life

Cell	Note	Reference
O6	Mainframe	13, 30, 49, 52, 75
O7	Unix	13, 49, 86, 113
O8	PC server	13, 49, 86
O9	PC server	13, 49, 86
O10	PC server	13, 49, 86
O11:O17	PC	8, 13, 44, 49
O20	Big-screen display	53
O26	Radar	53
O28	Fiber LAN	81, 101
O29:O36	Router	100
O37:O38	KVM	O8:O20
O39:O45	Hub	100
O47,O49	RAID	O8:O20
O52:O54	Rack	108
O55,O56,O59	UPS, printer	O8:O20
O66,O67	PBX, Ringing generator	O8:O20
O71:O77	Test equipment	O8:O20
O78	Liftcart	76
O111	Firmware	1, 83, 91, 129

Table 5-11. Sources for Parts Worksheet Entries: End of Hardware Service Life

Cell	Note	Reference
P6	Mainframe	30, 49, 52, 75
P7	Unix	49, 86, 113
P8	PC server	49, 86
P9	PC server	49, 86
P10	PC server	49, 86
P11:P17	PC	8, 44, 49
P20	Big-screen display	53
P26	Radar	53
P28	Fiber LAN	81, 101
P29:P36	Router	100
P37:P38	KVM	P8:P20
P39:P45	Hub	100
P46,P48,P50,P51	No equipment	Set to 100 to prevent #DIV/0 errors in Calc1 sheet
P47,P49	RAID	P8:P20
P52:P54	Rack	108
P55,P56,P59	UPS, printer	P8:P20
P66,P67	PBX, Ringing generator	P8:P20
P71:P77	Test equipment	P8:P20
P78	Liftcart	76
P111:P112	Firmware	1, 83, 91, 129

Table 5-12. Sources for Parts Worksheet Entries: End of Hardware Physical Life

Cell	Note	Reference
Q6	Mainframe	13, 30, 49, 52, 75
Q7	Unix	49, 53, 86, 113
Q8	PC server	49, 53, 86
Q9	PC server	49, 53, 86
Q10	PC server	49, 53, 86
Q11:Q17	PC	8, 44, 49, 53
Q20	Big-screen display	49, 53
Q26	Radar	47, 53
Q28	Fiber LAN	101, 81
Q29:Q36	Router	100
Q37:Q38	KVM	Q8:Q20
Q39:Q45	Hub	100
Q47,Q49	RAID	Q8:Q20
Q52:Q54	Rack	108
Q55,Q56,Q59	UPS, printer	Q8:Q20
Q66,Q67	PBX, Ringing generator	Q8:Q20
Q71:Q77	Test equipment	Q8:Q20
Q78	Liftcart	41; similar hydraulic parts
Q111:Q112	Firmware	1, 83, 91, 129

Table 5-13. Sources for Parts Worksh eet Entries: End of Software (Operating System) Market Life

Cell	Note	Reference
R6	Mainframe	32, 49, 52, 75, 98, 103, 105, 112
R7	Unix	49, 113, 103, 105, 112
R8	PC server	49, 110, 103, 105, 112
R9	PC server	49, 110, 103, 105, 112
R10	PC server	49, 110, 103, 105, 112
R11:R17	PC	8, 13, 49
R19:R25	Display	No relevant end of life; 100 years used
R26:R27	Radar	No relevant end of life; 100 years used
R28	Fiber LAN	2, 102
R29:R36	Router	100
R37:R38	KVM	R8:R20
R39:R45	Hub	100

Table 5-13. Sources for Parts Worksheet Entries: End of Software (Operating System) Market Life (Continued)

Cell	Note	Reference
R47,R49	RAID	R8:R20
R52:R54	Rack	108; standards evolve so that standard configurations change over given period
R55,R56,R59	UPS, printer	R8:R20
R66,R67	PBX, Ringing generator	R8:R20
R71:R77	Test equip-ment	R8:R20
R111:R112	Firmware	1, 83, 91, 129

Table 5-14. Sources for Parts Worksheet Entries: End of Software (Operating System) Service Life

Cell	Note	Reference
S6	Mainframe	13, 49
S7	Unix	49, 113
S8	PC server	49, 110
S9	PC server	49, 110
S10	PC server	49, 110
S20	Display	49, 70
S28	Fiber LAN	No relevant end of life; 100 years used
S29:S36	Router	100
S37:S38	KVM	100
S39:S45	Hub	100
S47,S49	RAID	Based on S8:S45
S52:S54	Rack	108
S55,S56,S59	UPS, printer	Based on S8:S45
S66,S67	PBX, Ringing generator	104
S71:S77	Test equipment	104
S111:S112	Firmware	1, 83, 91, 129

Table 5-15. Sources for Parts Worksheet Entries: End of Software Physical Life (unable to modify further)

Cell	Note	Reference
T6	Mainframe	13, 49
T7	Unix	13, 49, 113
T8:T10	PC server	13, 49, 110
T20	Display	Dummy value; no meaning
T28	Fiber LAN	No relevant end of life; 100 years used
T29:T36	Router	13, 100
T37:T38	KVM	13, 100
T39:T45	Hub	13, 100
T47,T49	RAID	Based on S8:S45
T52:T54	Rack	108
T55,T56,T59	UPS, printer	Based on S8:S45
T66,T67	PBX, Ringing generator	13, Based on S8:S45
T71:T77	Test equipment	13, Based on S8:S45
T111:T112	Firmware	1, 83, 91, 129

Table 5-16. Sources for Parts Worksheet Entries: SLEP Cost

Cell	Note	Reference
U6:U27	Computers	SLEP costs of two SLEP studies by Coast Guard and NAVAIR were used to parametrically derive a SLEP cost in relation to original equipment costs. 108, 127
U28:U56	Telecommunications components	SLEP formula derived for computer equipment produced unrealistically high figures for telecom equipment, so replacement costs were used.
U59,U66,U67:U78	Advanced telecom components	These components are more like computers than telecom. Based on CG/NAVAIR formula.
U111:U112	Repeat cards	Based on telecom figures; these components are not SLEP d since they can be reproduced almost indefinitely.

Table 5-17. Sources for Parts Worksheet Entries: End of Life and Failure Rate After EOL

Cell	Note	Reference
V6:V11	End of life	Lesser of the physical EOL of hardware or physical EOL of software.
W6:W11	Probability of failure	56

References

[1] Advantec. Single Board Computers. http://www.advantech.com/products/sbc.htm *Firmware*

[2] Allnutt, Jeremy. Introduction to Optical Fiber Communications. Virginia: Virginia Tech University, 2000. *Fiber networks, lasers, LEDs, diodes, fiber LANs, repeaters, power conversion; protocols and lifetimes*

[3] Amdahl Company. Summary of Amdahl s System Control Programs and Operating Systems Support Commitment. January 2000. www.amdahl.com. *Mainframe support*

[4] Ameritec Corporation. AM5XT-200 Enhanced TIMS 23 Nov 99. http://www.ameritec.com/databank/products/index_AM5XT-200.html *VSCS equipment*

[5] Ameritec. Solutions: AM8a T1 PCM Signalling Analyzer. http://www.ameritec.com/databank/products/index_AM8a.html *Test equipment*

[6] APC Corporation. Enterprise Power Reliability Package http://www.apcc.com/support/service/enterprise_power_reliability_package.cfm *UPS support contracts*

[7] Argecy Computer Corporation. Inventory and Support www.argecy.com *Server inventory and prices, mainframe inventory and prices*

[8] Belair, Doug, and Mike Eagan. Engineering Off-The-Shelf Solutions: A Life Cycle View *Naval Engineers Journal* May 1996: 181-191.

[9] Braun, Jerry L. and Mark Chestnutwood Technology Refresh Cost Prediction Naval Surface Warfare Center Briefing, 14 April 2000.

[10] Braun, Jerry L. Commercial Technology Refresh Budgeting and Planning Naval Surface Warfare Center Briefing, 2000.

[11] Braun, Jerry L. Commercial Technology Refresh Planning and Budgeting. Naval Surface Warfare Center Briefing , 2000.

[12] Braun, Jerry L. COTS Electronic Technology Assessment/Refresh Cost Model Naval Surface Warfare Center Briefing , 2000.

[13] Brooks, Jr.,F. P. *The Mythical Man-Month: Essays on Software Engineering*. Reading, Massachusetts: Addison Wesley Longman, Inc., October 1995.

[14] C2Cost, Electronic Systems Center, Hanscom Air Force Base. COTS Software Primer. www.c2cost.com/cotssoftwareprimer.htm.

[15] C2Cost, Electronic Systems Center, Hanscom Air Force Base. COTS System O&S. www.c2cost.com/system.htm.

[16] C2Cost, Electronic Systems Center, Hanscom Air Force Base. Sample ACE-IT Sessions for C2 Systems. http://www.c2cost.com/sampleaceit.htm_*COTS Refresh estimating*

[17] Ciufo, Chris End-of-Life: New Issues, New Choices With COTS Technology. *COTS* January/February 2000. <www.rtcgroup.com/costsjournal/>

[18] CNET Shopper. CNET: Ethernet Adapter. http://techsearch.cnet.com/search?cat=10 apter&tag=st.cn1.srch.sr&search=+Go%21+. *Ethernet equipment*

[19] CNET Shopper. CNET: Hardware: Networking: Hubs: 3Com OfficeConnect Dual Speed Hub 16. http://computers.cnet.com/hardware/0-7053-404-131578.html *Hubs*

[20] CNET Shopper. CNET: Hardware: Networking: Hubs: Reviews and Prices. http://computers.cnet.com/hardware/searc - 1,00.html?tag=st.co1037.dir.7053-402-0. *Hubs*

[21] CNET Shopper. CNET: Hardware: Networking: Hubs: Specifications and Prices. http://computers.cnet.com/hardware/search/results/0,10238,0-7053-402-0-1,00.html?tag=st.co.1037.dir.7053-402-0 *Hubs*

[22] CNET Shopper. CNET: Shopping: KVM: products and prices. http://shopper.cnet.com/shopping/search/results/1,10214,0-1257,00.html?st.sh.1257.sbsr&qt=Belkin+KVM&cn=&ca=1257&tag=st.cn.sr.sh.more *KVM*

[23] CNET Shopper. CNET: Shopping: Network & communications: Bridges/routers/gateways: Gateways. http://shopper.cnet.com/shopping/0-11637 tml ?tag=st.sh.11023-301-0.1.lst.list_11637. *Gateways*

[24] CNET Shopper. CNET: Shopping: Network & communications: Bridges/routers/gateways: Routers. http://shopper.cnet.com/shopping/0-11643 tml ?tag=st.sh.11023-301-0.1.lst.list_11643. *Routers*

[25] CNET Shopper. CNET: Shopping: Network & communications: Internet server & access units: Internet servers. http://shopper.cnet.com/shopping/search/results/1,11699 tml?tag=st.sh.1 1027-301-0.1.lst.list_11699 S*ervers*

[26] CNET Shopper. CNET: Shopping: Network & communications: Perip h-eral servers & sharing units: Miscellaneous server & sharing units. http://shopper.cnet.com/shopping/0-1,11710 tml?tag=st.sh.11029-301-0.1.lst.list_11710. *Servers*

[27] CNET Shopper. CNET: Shopping: Printers: Laser Printers. http://shopper.cnet.com/shopping/0-11869-301-0-0.html?tag=st.sh.11199-301-0.lst.list_11869 *Printers*

[28] CNET Shopper. CNET: Shopping: Printers: Line Printers. http://shopper.cnet.com/shopping/0-11876-301-0-0.html?tag=st.sh.11200-301-0.lst.list_11876 *Printers*

[29] CNET Shopper. Product: Power Protection & Batteries: UPS Systems: online. http://shopper.cnet.com/shopping/0-11837-301-0-0.html?tag=st.sh.11156-301-0.lst.list_11837 *UPS*

[30] CNNfn. IBM Makes Introduction: Computer maker plans major mai n-frame upgrade. 9 June 1997. http://europe.cnnfn.com/digitaljam/9706/09/computers/. *Mainframes, support*

[31] Comp Source Website. PC drives. http://www.c-source.com/csource/OResult.asp?part_no=699078 *PCs*

[32] Computer Related Concepts Corporation. Customer Support. http://www.crcinc.com/cgi-bin/index?Page=/multiprise3000.htm S*upport*

[33] Datum-Efratom. Low Profile Rubidium Oscillator. Adobe Acrobat electronic file. 2000. *Test equipment*

[34] Datum-Efratom. Modular Frequency and Time System. Adobe Acrobat electronic file. 2000. *Test equipment*

[35] Datum-Efratom. Portable Rubidium Frequency Standard. Adobe Acr o-bat electronic file. 2000 *Test equipment*

[36] Datum-Efratom. StarLoc II: Precision GPS Time & Frequency Reference TM. Adobe Acrobat electronic file. 2000. *Test equipment*

[37] Day, Virginia, Collaborative Obsolescence Management and Evaluation Tool (COMET). *The MITRE Advanced Technology Newsletter*. July 1997. www.mitre.org/pubs.

[38] Dell Computer Corporation Enterprise Server Series. http://www.dell.com/us/en/fed/products/series_enter_servers.htm *Servers*

[39] Dell Computer Corporation PCs www.dell.com *PCs*

[40] Delta Lu Corporation. Ethernet Hub — Ethernet LAN Hub 10 BaseT, 10 Base 2. http://www.deltalu.com/10ehub.htm *Ethernet*

[41] Denson, W., G. Chandler, W. Crowell, and R. Wanner. *Nonelectronic Parts Reliability Data*. Rome, New York: Reliability Analysis Center, 1991. *Reliability and lifetime data*

[42] Eagan, Michael, and Sidney Hankerson. Industry Government Partne r-ship: Incorporation of COTS/NDI Hardware & Software in the Trident Strategic Fire Control System. Powerpoint Slides. 1998. http://www.acq.osd.mil/dsac/s98chrts/eagan/tsld001.htm *COTS Refresh*

[43] *Electronic Parts Reliability Data*. Rome, New York: Reliability Analysis Center, 1997. *Reliability and lifetime data*

[44] Electronic Parts Obsolescence, *USAF Manufacturing Technology Status Report*, Air Force Research Laboratory, 1999. Pp 8-10.

[45] ETC Telecomm Corporation. Telephone, Telecomm and Network Equipment for the Central Office. http://www.etc.telecomm.com/switching.html *PBX*

[46] Federal Aviation Administration. *Acquisition Program Baseline for Host and Oceanic Computer System Replacement*. Washington, D.C.: FAA: 1 May 1998. *HOCSR*

[47] FAA. *Air Traffic Services NAS Infrastructure Sustainment Shortfalls, A Special Study Report Prepared by Air Traffic Systems Requirements Service (ARS) Integrated Requirements Team (IRT), Version 1.0* Wash-ington, D.C.: FAA: 1999.

[48] FAA. Airport Surveillance Radar, Model 9. http://www.tc.faa.gov/act310/projects/asr9/asr9.htm *ASR-9*

[49] FAA. *AUA Technology Refresh Planning Guide* Washington, D.C.: FAA-AUA, 5 January 1998. *FAA systems*

[50] FAA. Benefit and Cost Analysis of Low-Cost Airport Surface Detection Equipment (ASDE-X). FAA-ASD-100, October 1993. *Radar costs*

[51] FAA. FAA Contract Opportunities Database, excerpt for Software Maintenance Compaq/Tandem Himalaya K2000. 16 March 2000. http://www.asu.faa.gov/faaco/posting.cfm?PostIDtemp =957. *VSCS*

[52] FAA. *Enroute/Oceanic Domain Mission Need Statement (MNS-309)* Washington, D.C.: FAA, 6 November 1997. *HOCSR*

[53] FAA. *Guidance: Choice of Economic Service Life (ESL) for FAA Analysis Purposes*. Washington, D.C.: FAA, 29 September 1998.

[54] FAA. HOCSR Tech Refresh Budget Excel spreadsheet. 1998. *HOCSR*

[55] FAA. *Host Oceanic Computer System Replacement (HOCSR):Pre Site Survey Briefing, ZDV*. Powerpoint briefing by En Route Integrated Product Team and Oceanic and Offshore Integrated Product Team. 11 June 1998. *HOCSR*

[56] FAA. *Investment Analysis Report (IAR)(Acquisition Phase) For (MNS-309/IHOST/ODAPS/OFDPS).* Washington, D.C.: FAA. 1998. *HOCSR*

[57] FAA NAS Architecture Surveillance Modernization 29 July 99. http://www.faa.gov/nasarchitecture/AEEC99/Surv-a1.ppt. *ASR-9*

[58] FAA. NAS Architecture: Transition of Surveillance Capabilities 26 Mar 99. http://www.faa.gov/nasarchitecture/blueprnt/surv2.htm *ASR-9*

[59] FAA. Resource Planning Data Database: excerpt for VSCS. 24 March 1999. *VSCS*

[60] FAA. *Rough Order-of-Magnitude Costs of Ground-Based Elements for the Terminal Area Productivity Program.* FAA-ASD-420, Dec. 1996. *Program cost factors as a percent of PME*

[61] FAA. *System Outage Disruption Model (SODM): A User s Guide*. Washington D.C.: FAA-ASD-400, March 2000. *Social cost of outages*

[62] FAA. Surveillance Integrated Product Team (AND-400) 19 Feburary 1999. http://www.faa.gov/and/and400/400home.htm *ASR-9*

[63] FAA. *Special Topics in Investment Analysis Technology Refreshment of NAS Equipment.* Washington, D.C.: FAA-ASD. 20 November 1998 *FAA systems*

[64] FAA. Trouble Tracking System Database: maintenance excerpts for VSCS from 1995-1998. Electronic database. *VSCS*

[65] FAA. VSCS O&M Summary Excel spreadsheet. 17 May 1999. *VSCS*

[66] FAA. VSCS Waterfall Chart. 18 February 1997. *VSCS*

[67] Fibre Channel Technology . Light Pulse PCI Fibre Channel Adapter, Light Pulse Fibre Hub, LightPulse Mini-Hub. http://www.nzdisk.co.nz/emulex/fchan.html *Hubs, RAID stacks*

[68] Freeman, R. L. *Telecommunication System Engineering Third Edition* . New York, New York: John Wiley & Sons, Inc., 1996. *Communication systems hardware, support, lifetimes*

[69] Galorath Corporation. Introducing Operations and Support in SEER-H. www.galorath.com/downloads.pdf. *Hardware support models*

[70] Gateway 2000 Inc. 21 Monitors http://necxdirect.necx.com/hai/prod_page.html?key=0000090237&nonce=guest_gate_biz *Monitors/displays*

[71] Gateway 2000 Inc. Corporate Servers. http://www.gateway.com/prod/gv_servers.shtml *Servers*

[72] Gateway 2000 Inc. PCs www.gw2k.com *PCs*

[73] Gateway 2000 Inc. Rackmounts. http://www.gatewayatwork.com/prod/cp_rck_category.shtml *Racks*

[74] Gateway Spot Shop. Top Selling UPS- Fileservers and LANs http://necxdirect.necx.com:80/hai/top_seller_page.html?key=0001453&nonce=guest_gate_biz *UPS*

[75] Garvey, Martin J. IBM Preps Mainframe Update, Boosts CRM Support. *InformationWeek*. 2 March 2000. www.internetwk.com. *HOCSR*

[76] Genie Corporation Genie Portable Machinery Lifts http://www.genielift.com/ml-series/index2.html *VSCS equipment*

[77] Government Technology Services Sales quotation. Dec. 1996. *RAID, Sunsoft, Servers*

[78] Greer, Larry. FAAAC Tandem Maintenance. E-mail to Sadie Walthers, 9 Feb. 2000. *VSCS*

[79] Harris Corporation Products: Network Support: Network Switching Systems Product line http://www.commprod.harris.com/network-switching-systems/ *VSCS equipment*

[80] Harris Corporation. *VSCS Control Subsystem Upgrade (VCSU).* Powerpoint briefing. 1999. *VSCS*

[81] Hecht, Jeff. *Understanding Lasers*. U.S.: Howard Sams & Co. 1988. *FDDI*

[82] Hello Direct. *Catalog.* San Jose, CA: Hello Direct. 2000. *PBX*

[83] Hewlett-Packard. Single Board Computers. http://www.hp.com/es/products/744.htm *Firmware*

[84] Interview with John McDermid: The Cost of COTS. Computer June 1998. http://computer.org/computer/co1998/r6toc.htm.

[85] Kostiuk, Peter. *System Safety and Operational Availability Modeling*. Powerpoint briefing. 2000. *ASAC Safety Model*

[86] Leete, Jon. Re: Computer lives. E-mail to V. Stouffer. 3 March 2000. *Unix lifetimes, cost, and support.*

[87] Lockheed Martin. *DSR Technology Refresh Task Order 40: FAA Briefing.* Powerpoint presentation. 30 April 1998. *HOCSR*

[88] Lockheed Martin. *Display System Replacement (DSR) Technology Refresh Strategy Task Order 40: A Comprehensive Strategy to Maintain the Technological Currency of the DSR System over the Timeframe 2001-2015*. 30 April 1998. *HOCSR*

[89] McKamey, Jerry L. Designing Survivable Shipboard Equipment Using Commercial-Off-The-Shelf Technology. Undated Powerpoint briefing. www.OSD.acq.mil.

[90] Metric Company.. Quality Used Refurbished Electronic Test Equipment http://209.164.106.113/htm/used.htm. *Test equipment*

[91] Microware. Microware Ariel: Royalty Free Full Source Real Time Operating System. http://www.microware.com/ariel_eval.html. *Firmware*

[92] Navarro, Sergio. RePLACE A Generic COTS Based Solution to Aging Avionics Computer Modernization. Undated Powerpoint briefing. www.OSD.acq.mil.

[93] Oracle On-line Store. Oracle8i Standard Edition. http://oraclestore.oracle.com/cec/cstage. *OS*

[94] Oracle On-line Store. Oracle8i Enterprise Edition. http://oraclestore.oracle.com/cec/cstage. *OS*

[95] Oracle On-line Store. Oracle.com Suite. http://oraclestore.oracle.com/cec/cstage tion==espassthru2&template== walkin1.en.htm. *OS*

[96] Oracle On-line Store. Oracle E-business Integration solution. http://oraclestore.oracle.com/ebusiness/integration/solution.html. *Servers and support*

[97] Outpost.com, CNET Shopper. Product Information, Belkin (KVM Switches): OmniCube 4 Port PS/2 KVM Switch http://www.outpost.com/entry?site=cnet:shopper&sku=40938&IntraSite302=web55349. *KVM*

[98] Panettieri, Joseph and Ed Sperling, Can Oracle Hang On? *Excite News* 7 March 2000. www.excite.com. *OS support*

[99] Perkin Elmer Optoelectronics . Power Supplies - Hi-Rel, FAA VSCS Program http://opto.perkinelmer.com/products/catalog/familylisting/ *VSCS equipment*

[100] Peterson, L.L. and B.S. Davie. *Computer Networks: A Systems Approach*. San Francisco, California: Morgan Kaufmann Publishers, Inc., 1996. *Network systems, protocols, maintenance requirements, past lifetimes.*

[101] Photonics Magazine, Photonics 99. CD-ROM. Pittsfield, MA: Laurin Publishing. 1999. *Fiber networks, lasers, semiconductors*

[102] Powers, John. *An Introduction to Fiber Optic Systems, Second Edition.*. U.S. A.: McGraw-Hill Companies. 1997. *Fiber networks, lasers, semiconductors*

[103] PR Newswire. Oracle Provides Comprehensive Solutions for Call Center, Web and E-mail Customer Interactions *Excite News* 1 March 2000. *OS Support*

[104] Product Quotes. *RFGlobalnet* April 2000. http://www2.rfglobalnet.com/content/hubs/dir.asp?hub=ProductInfo. *Lifetimes of test equipment and rare hardware pieces*

[105] Razavian, Abbas. Software Lifecycle Processes. COTS January/February 2000. www.rtcgroup.com/cotsjournal/.

[106] Razavian, Abbas. A Product Obsolescence Forecasting Model for COTS Based Systems. *COTS* January/February 2000. www.rtcgroup.com/costsjournal/.

[107] Stevens, Marilyn. Re: RE: Datum-Irvine Information Request. E-mail to V. Stouffer. 10 May 2000. *Test equipment*

[108] Stouffer, Virginia, C.L. Daly, T.K. Parks, R.J. Marafioti, A.J. Colaianni, M.K. Love. Total Ownership Cost Estimates and Major Cost Factors — Great Lakes Icebreaker Capability, LMI Letter Report, CG908L1, Febru-

ary 2000. *SLEP cost factors, rack, communication and computer equipment prices and lives*

[109] Sun Microsystems. Servers. http://store.sun.com/servers, http://store.sun.com/catalog/doc/BrowsePage.jhtml?cid=48615 *Servers*

[110] Sun Microsystems. SunSpectrum Support Services http://www.sun.com/service/support/sunspectrum/index.html *Servers, OS, and support*

[111] Tellabs Corporation Signaling Products: Signalling Converter Modules, Repeaters, Information and Technical Support. 4 May 2000. http://www.tellabs.com/products/signal/index.shtml. *VSCS*

[112] Tepp, Brian. Managing the Risk of Parts Obsolescence. *COTS*, September/October 1999. www.rtcgroup.com/cotsjournal.

[113] Tyler, Jim. RE: Sun gen. E-mail to V. Stouffer. 11 May 2000. *Unix and OS lifetimes*

[114] U.S. General Accounting Office. *Air Traffic Control: Complete and Enforced Architecture Needed for FAA Systems Modernization* (GAO-AIMD-97-30). Washington, D.C.: GPO, February 1997. *PAMRI architecture*

[115] Unison, Bob. Fibre Pricing. E-mail to V. Stouffer. 22 March 2000. *Memory pricing*

[116] Unison Information Systems. *Quotation: RFFF2 Series with 69.1 GB 7200 RPM fibre channel disk drives.* 22 March 2000. *Memory pricing*

[117] Unison Information Systems. RAID I/O Fibre. http://unisoninfo.com/raid.htm. *Memory pricing*

[118] Veriplus Inc. Master View: KVM Switches. http://www.veriplus.com/idx.masterview.htm *KVM*

[119] Walsh, Donald. Service Level Agreement Pricing. E-mail to V. Stouffer. 6 March 2000. *PC and server support contracts*

[120] William J. Hughes FAA Technical Center Airport Surveillance Radar. 16 Mar 00. http://www.tc.faa.gov/act310/projects/asr9/asr9.htm. *ASR-9, solid-state technology*

[121] William J. Hughes FAA Technical Center. AOS-270 (National Airways Systems Engineering Division) Home Page, ASR-9 RDAS Branch 10 Apr. 00. http://www.aos.tc.faa.gov/aos270/Home.htm. *ASR-9*

[122] William J. Hughes FAA Technical Center. ASR-9 Terminal Radar Group. 01 Feb 00. http://www.aos.tc.faa.gov/aos270/ASR9/ASR9main.htm. *ASR-9.*

[123] William J. Hughes FAA Technical Center. Aviation Weather Develo p-ment Facility (AWDF). 14 Dec 98. http://www.tc.faa.gov/its/cmd/visitors/data/ACT-300/awdf.pdf . *ASR-9*

[124] William J. Hughes FAA Technical Center. Commercial Off-The-Shelf Products Used for VSCS, AOS-520-ACY (Communications Systems Engineering Support Branch) http://www.aos.tc.faa.gov/vscs/cots.htm *VSCS COTS testing components*

[125] William J. Hughes FAA Technical Center. Projects: Beacon Video Reconstituter 16 Mar 00 http://www.tc.faa.gov/act310/projects/bvr/page1.htm. *ASR-9*

[126] Wilson, J. R. Obsolescence: The New Enemy. *Aerospace America* March 2000: 38-41.

[127] Wingrove, Earl, M.S. Bridgman, J. Dukovich, R.P. Neisler, V.S. Stouffer, and R.G. Vuori. Consolidated Automated Support System Acquisition Alternatives, LMI Brief Report, NA905B1, April 1999. *SLEP cost factors*

[128] Yahoo. NetGear DS-116: 10 and 100MBps Ethernet Hub . *Yahoo store product directory*. http://store.yahoo.com/tekgallery/netgeards116.html. *Ethernet and network connectors*

[129] MicroFirmware. I/O Cards. http://www.firmware.com/sales/IO/index.htm *Firmware*

Appendix A
Abbreviations

AAS	=	Advanced Automation System
ACE-IT	=	brand name of a cost estimating software product
ADOC	=	airline direct operating cost
APB	=	Acquisition Program Baseline
ARTCC	=	Air Route Traffic Control Center
ARTS	=	Automated Radar Terminal System
ASAC	=	Aviation System Analysis Capability
ASR	=	Air Route Surveillance Radar
AUA	=	Air Traffic System Development Office
CAS	=	commercially available software
CD	=	compact disk
CD-ROM	=	compact disk read only memory
CER	=	cost estimating relationship
COTS	=	commercial off-the-shelf
DOS	=	disk operating system
DSR	=	display system replacement
EDARC	=	Enhanced Direct Access Radar Channel
EOL	=	end of life
EOML	=	end of market life
EOPL	=	end of physical life
EOSL	=	end of service life
F&E	=	facilities and equipment
FAA	=	Federal Aviation Administration
FAA-ASD	=	Federal Aviation Administration s Office of System Architecture and Investment Analysis
FDIO	=	Flight Data Input-Output Computer
GAO	=	General Accounting Office
GPS	=	Global Positioning System

GUI	=	graphical user interface
HOCSR	=	Host/Oceanic Computer System Replacement
HW	=	hardware
IEEE	=	Institute of Electrical and Electronic Engineers
IP	=	internet protocol
IPT	=	integrated product team
LAN	=	local area network
LMI	=	Logistics Management Institute
LRU	=	line replaceable unit
MAX	=	maximum function
MS-DOS	=	Microsoft Disk Operating System
NAS	=	National Airspace System
NAVAIR	=	Naval Air Systems Command
NAVSEA	=	Naval Sea Systems Command
O&M	=	operations and maintenance
OEM	=	original equipment manufacturer
OMB	=	Office of Management and Budget
OS	=	operating system
PAM	=	Peripheral Adapter Module
PAMRI	=	Peripheral Adapter Module Replacement Item
PBX	=	phone switchboard
PC	=	personal computer
PRICE-H	=	brand name of a cost estimating software product
PUFF	=	paid-up-front fee
RAID	=	redundant array of independent disks
RMM	=	remote maintenance monitoring
SLEP	=	Service Life Extension Program
SLOC	=	software lines of code
SODM	=	System Outage Disruption Model
STARS	=	Standard Terminal Automation Replacement System
SW	=	software
TR	=	tech refresh
TRACON	=	Terminal Radar Approach Control

UPS	=	uninterruptible power source
V2TEP	=	VSCS-VTAB Technology Evolution Plan
VSCS	=	Voice Switching Communication System

REPORT DOCUMENTATION PAGE

Form Approved
OPM No.0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources gathering, and maintaining the data needed, and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Information and Regulatory Affairs, Office of Management and Budget, Washington, DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	Jun 00	Final

4. TITLE AND SUBTITLE

Technology Refreshment Cost Estimating and Planning Model: User's Guide

5. FUNDING NUMBERS

C PO 1-0000139056

PE 0902198D

6. AUTHOR(S)

Virginia Stouffer, Robert Hemm

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Logistics Management Institute
2000 Corporate Ridge
McLean, VA 22102-7805

8. PERFORMING ORGANIZATION REPORT NUMBER

LMI— NX001C1

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Federal Aviation Administration
800 Independence Ave. S.W.
Washington, DC 20591

University of California - Berkeley
109 McLaughlin
Berkeley, CA 94720-1720

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

A: Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

The FAA is using commercial off-the-shelf (COTS) components in its major air traffic control systems and needs to know COTS future years cost impacts, in terms of refreshment and maintenance of systems with deferred refreshment. This report describes a model that estimates COTS refresh and maintenance costs as they increase through time. The Technology Refreshment Cost Estimating and Planning Model predicts the maintenance and refreshment funding needs of COTS systems for any year over a 30-year horizon. Model users can vary year of installation and year of refreshment and derive optimal refresh cycles. Budget analysts can use the model interactively in the budgeting process to determine the maintenance cost impact of delayed acquisition funding. The algorithms of the model were also used to derive a cost estimating relationship for use in predicting FAA s entire tech refresh cost burden. Chapter 1 gives model installation instructions. Chapter 2 tells how to operate the model and describes the process flow of an analysis. Chapter 3 is a page-by-page discussion of the model spreadsheets. Chapter 4 addresses model expansion options. Chapter 5 describes data sources.

14. SUBJECT TERMS

COTS, commercial off-the-shelf, NDI, Tech Refresh, Technology Refreshment, cost, cost analysis, model, cost model, budget analysis, maintenance costs, acquisition deferment, delay in acquisition, increasing support costs, computer equipment, telecommunication equipment, FAA, air traffic control

15. NUMBER OF PAGES

115

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL