



NASA-CR-200152

*IN-33-CR
OVERRIDE
7990
P. 12*

AIAA 95-3561

Electronic Collection System For Spacelab Mission Timeline Requirements

J. P. Lindberg, J. R. Piner, and A. K. H. Huang
Teledyne Brown Engineering
Huntsville, Al

(NASA-CR-200152) ELECTRONIC
COLLECTION SYSTEM FOR SPACELAB
MISSION TIMELINE REQUIREMENTS
(Teledyne Brown Engineering) 12 p

N96-18526

Unclas

G3/33 0099871

**AIAA 1995 Space Programs and
Technologies Conference
September 26-28, 1995/Huntsville, AL**

ELECTRONIC COLLECTION SYSTEM FOR SPACELAB MISSION TIMELINE REQUIREMENTS

James P. Lindberg, John R. Piner, and Allen K. H. Huang

Teledyne Brown Engineering
300 Sparkman Drive
Huntsville, Al 35807-7007

ABSTRACT

This paper describes the Functional Objective Requirements Collection System (FORCS) software tool that has been developed for use by Principal Investigators (PIs) and Payload Element Developers (PEDs) on their own personal computers to develop on-orbit timelining requirements for their payloads.

The FORCS tool can be used either in a totally stand-alone mode, storing the information in a local file on the user's personal computer hard disk or in a remote mode where the user's computer is linked to a host computer containing the integrated database of the timeline requirements for all of the payloads on a mission.

There are a number of features incorporated in the FORCS software to assist the user. The user may move freely back and forth between the various forms for inputting the data. Several methods are used to input the information, depending on the type of the information. These methods range from filling in text boxes, using check boxes and radio buttons, to inputting information into a spreadsheet format. There are automated features provided to assist in developing the proper format for the data, ranging from limit checking on some of the parameters to automatic conversion of different formats of time data inputs to the one standard format used for the timeline scheduling software.

INTRODUCTION

Teledyne Brown Engineering (TBE) serves as the Spacelab Payload Mission Integration Contractor (PMIC) for the NASA Marshall Space Flight Center (MSFC) and has been heavily involved in Spacelab payload integration for a number of years. As part of the Continuous Process Improvement efforts, TBE has been involved in developing new tools for the payload integration process. One of the new tools developed is the Functional Objective Requirements Collection System (FORCS) for defining the payload on-orbit activity timelining requirements. Another software tool developed by TBE for collecting payload requirements is one for defining Command and Data Management System (CDMS) requirements.

Spacelab missions involve Principal Investigators (PI) and Payload Element Developers (PED) geographically dispersed throughout the world. Previous methods for collecting Spacelab payload activity timelining requirements from the PI/PED teams involved them filling out paper FO forms and mailing or FAXing the forms to the payload mission integration team. The information was then entered into computer files on a personal computer by the payload integration team mission planning personnel. Copies of the FO forms were then distributed to members of the payload mission integration team and FAXed back to the PI/PEDs for markup. The cycle was then repeated until the FOs

met everyone's satisfaction. This process is illustrated in Fig. 1.

making it much easier for all of the payload integration disciplines and the

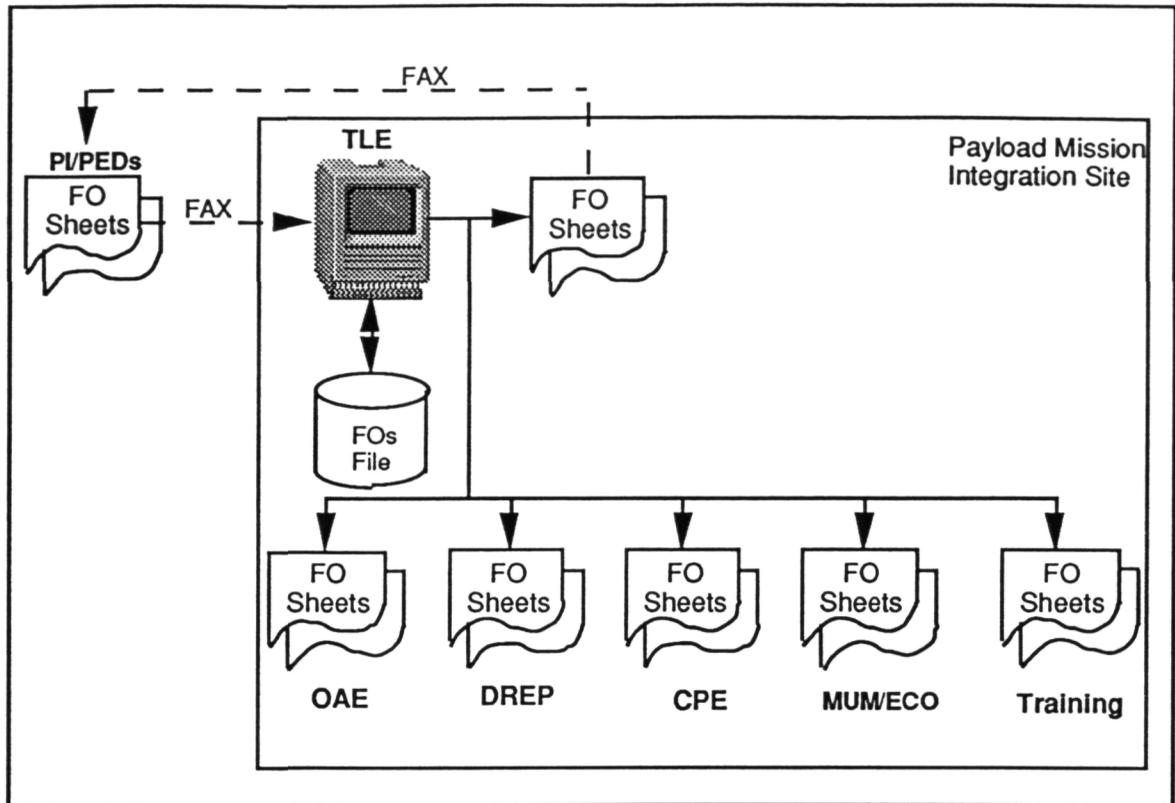


Figure 1. Previous FO Development Method

FORCS was developed to address two basic problems related to FO requirements. The first had to do with improving the interface with the PI/PEDs in the collection of the FO requirements. The other had to do with the dissemination of the FO information after it had been collected.

FO information is required by a number of the different engineering disciplines within the payload integration team. For a significant period of time, the FOs are in a dynamic state of change and it is important for the engineering disciplines to have the latest information readily available to them. Distributing paper copies of the FOs to the different payload integration disciplines is a cumbersome and difficult process. Therefore, a centralized database has been implemented to store the FO information,

PI/PED teams to have ready access to the latest information.

A number of benefits are anticipated from using FORCS, including the following:

- Original data provided in electronic form,
- Data easily updated by the originator or by the payload integration team,
- Data can be formatted such that it is directly recognizable by the timelining software,
- Reports of FO requirements can be readily created,
- Provides the potential for obtaining data from different PI/PED teams in a more consistent format,
- Information is readily available electronically on-line for easy access by payload integration personnel and PI/PED teams.

TIMELINE GENERATION PROCESS

The payload activity timeline is the integrated time ordered sequence of all on-orbit payload activities to be accomplished on a mission. The timeline also includes the integrated resources required to support these payload activities. The mission planners need timeline requirements from all of the PI/PED teams to generate the payload activity timeline. These requirements are defined by means of Functional Objectives (FO). A Functional Objective is a major collection of an experiment's procedures which accomplishes a definite purpose, such as, activating an experiment, collection of scientific data, etc. The FOs define the specific time sequenced activities to be done, along with constraint information (such as time period desires or restrictions, necessary orbit related conditions, etc.) and resource requirements (such as electrical power, data collection requirements, crew support needs, etc.).

The software program used to develop Spacelab payload activity timelines is the Experiment Scheduling Program (ESP), developed by the MSFC. The FOs are the mechanism for getting the scheduling requirements from the PI/PED teams. The FOs have been structured to interface with the investigator community (PIs) in terms most familiar to them and to insulate them from the strict data format and structures required by the ESP scheduling program. The mission planners take the information, which may include some important subjective information, from the FOs and develop ESP models. The ESP models are the direct inputs to the ESP scheduling program. Building the ESP models has been a totally manual process, but another software program has been developed by TBE to automate a substantial amount of transferring the information from the FO format to the ESP model format. This program is now in the initial stages of being used in normal production.

FO DESCRIPTION

A Functional Objective is organized into FO performance scheduling information and step data. An FO step is a sub-division of an FO, usually represented by a change in resource requirements or constraints. A Functional Objective (FO) includes a substantial amount of information. Documentation for an FO includes up to seven different forms. The possible forms are:

1. FO Sheet
2. FO Continuation Sheet
3. Comments
4. Intra-Step Video
5. TV Requirements
6. Photo Requirements
7. FO Expansion Sheet

The FO Sheet contains a Header section and a Step Data section. The Header includes information related to the total FO performance scheduling requirements and the Step Data includes the detailed resource and constraint information for the FO. The Comments sheet includes supplementary information, usually of a qualitative type, related to the FO. An example of an FO Sheet is shown in Fig. 2.

The Intra-Step Video includes video scheduling information when video scheduling requirements vary within a step duration. The TV and Photo Requirements contain detailed information related to definition of the TV and Photo scenes desired. The FO Expansion Sheet contains supplementary information about the FO that is used by the Operations Support personnel in the Payload Operations Control Center (POCC) during the mission.

FUNCTIONAL OBJECTIVE REQUIREMENTS SHEET		Page 1 of <u>3</u>					
Experiment Name: <u>GBX Fibers Supported Droplet Combustion</u>		Date: <u>2/13/95</u>					
FO Name: <u>FSDC Operations With Methanol</u>		FO Number: <u>3</u>					
No. of Performances: Required: <u>1 (A)</u> Desired: _____		FO Acronym: <u>GBX3</u>					
Performance Delay: Pref: _____ Min: _____ Max: _____		Prerequisite FO(s): <u>GBX1</u>					
Rqd Time Frame (MET): Earliest Start: _____ Latest End: _____		Sequenced FO: _____					
		Concurrent FO(s): _____					
STEP NUMBER		1	2	3	4	5	6
STEP DELAY (dd/hh:mm)	Preferred						
	Minimum						
	Maximum						
STEP DURATION (dd/hh:mm)	Preferred	0:10	0:45	0:05	0:03	0:04	0:05
	Minimum						
	Maximum						
CREW	Number Required	1	1	1	1	1	1
	Number Monitor Pds.						
	Mon. Pd. Dur. (hh:mm)						
ORBIT OPPORTUNITY							
ATTITUDE REQUIREMENT							
CONSTRAINTS (L, W, T)							
ACCELERATION LIMIT (g)							
AVERAGE POWER (kw)			0.042	0.140	0.125	0.164	0.090
EXPERIMENT COMPUTER	ECAS Task Label						
	ECAS Task Size						
	No. STL Buffers						
	MTL Required						
DATA	ECIO/Voice/Ground Cmd				V	V	
	Video (SL/EXP)	RT Mandatory			EXP1,2(C)		
		D/L or Record				EXP1(E)	
		Chg Out Only				EXP2(C)	
	Analog/Digital	RT Mandatory					
		Dwn RT or Dump					
	Ded Channel Rate (mbps)						
Photo Equipment							
SPECIAL OPS/EQUIPMENT							
STEP NO	STEP TITLE	STEP DESCRIPTION					
1	Unstow Equipment						
2	Set-up In GBX	Install cables, video, clamp, flashlight & fuel supply into fit unit. Rotate ignitor wire assembly. With RT video downlink to ensure proper set-up.					
3	Set-Up In GBX						
4	Test						
5	Normal Reconfigure	Includes exp. fan changeout of test volume gas. Reposition needles and photographic check.					
6	Ignitor Wire Change-Out						

Figure 2. First Page Of FO Sheet

FORCS

Overview

The total FORCS includes a software application (the part usually being addressed when we refer to FORCS) for the PI/PED teams to define their FO requirements, coupled with an Oracle centralized electronic database at the payload integration site to store and maintain the FO information for all payloads. The new process is illustrated in Fig. 3.

Remote Or Local Operations

There are two different environments that the FORCS may operate in and which can be selected by the user when he/she runs the software application. One, is operating on a remote personal computer linked to a host computer where a centralized database of FOs is maintained. The other is where the personal computer with the FORCS software application is operated in a standalone mode. When used in the standalone mode, the FOs created are maintained in a file on the

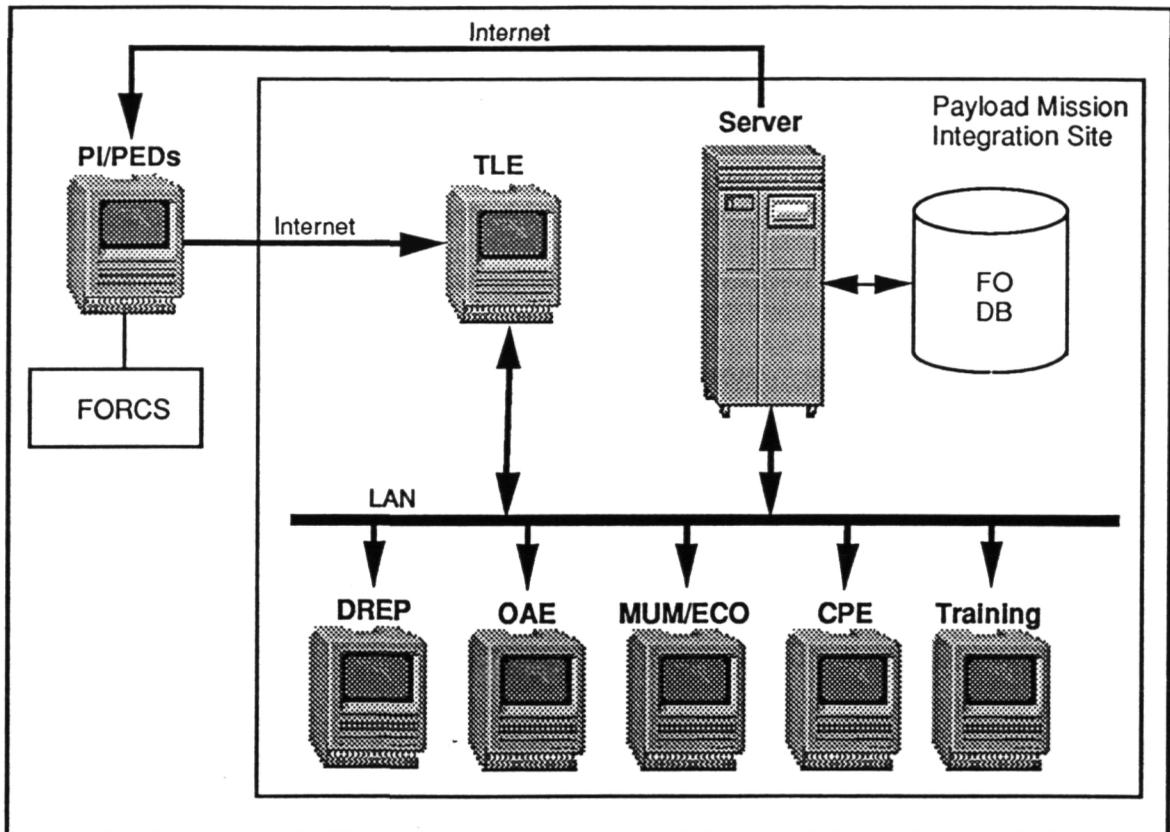


Figure 3. Revised FO Development Method

Hardware System Requirements

The FORCS was developed using the SuperCard development environment and runs on a Macintosh computer. Best response is obtained with a Macintosh IIfx with at least 4 Mbytes of RAM and a 14 inch monitor. The system can run on slower versions of the Macintosh, but system response is substantially degraded.

user computer's hard disk. Data may be interchanged via SQLNET between an individual computer and the centralized database on the host computer.

If there is a large amount of editing between FOs required, there is a substantial gain in speed when the FO editing is done in the local mode and subsequent files are transferred to the central database. Files may also be down

loaded from the central database to the individual computers operating in the standalone mode.

User Interface

The FORCS is a typical windows type of application where the user interacts with the system through a series of windows for defining the input information. There are a number of different forms (windows) that are used to input the data. These forms are patterned after the various output forms so that the user will be able to readily relate the information he/she is inputting to how it will be presented in the output. However, the input forms have a number of additional features incorporated other than simply filling in a set of blanks.

The different forms used to input the data are the following:

1. Header
2. Step Data
3. Comments
4. Step Titles/Descriptions
5. FO Expansion Sheet
6. Intra-Step Video
7. Television Requirements
8. Photographic Requirements

entries. The different forms usually have a combination of data entry formats. Check boxes and radio buttons are also used on some of the Step Data form pop-up windows to help the inexperienced user in making properly formatted entries.

The user may access any of the data input forms at any time by accessing the desired form through the menu bar at the top of the screen. The user may enter information in any order he/she chooses. Information that is used across forms, such as Step Titles and Descriptions, is automatically and dynamically available as it is entered.

Data Protection

A password system has been incorporated so the capability exists to control the users that have access to change data. The password system will be implemented by the FORCS system administrators for the information in the centralized Oracle database. Passwords can be used to control write access to the different missions in the database and to different sections of the data for the individual FOs.

On-line Help

The system provides context sensitive on-

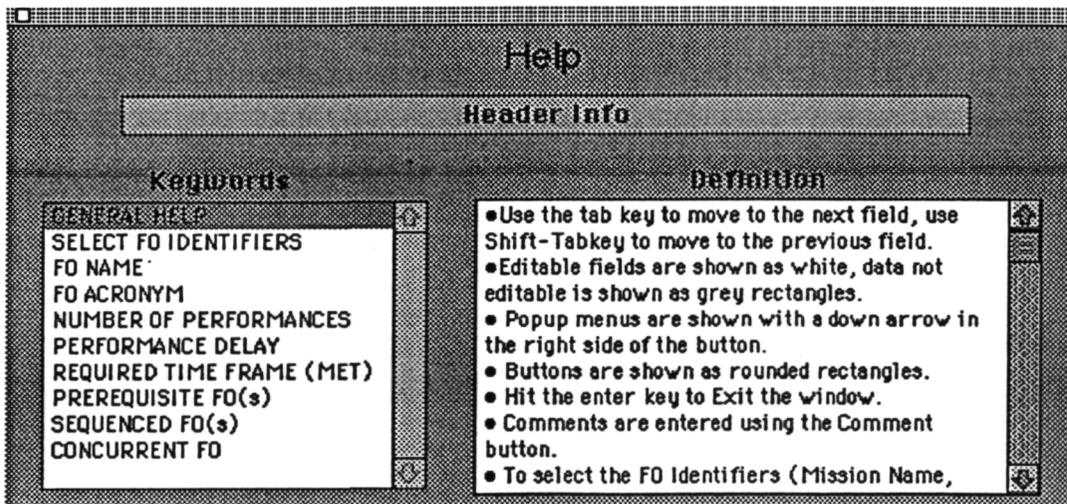


Figure 4. Help Window

Each of the above forms uses different styles for input of the data, from using text box entries to spreadsheet type

line Help. All primary data input forms and the pop-up windows incorporated with the Step Data input form have a set of

help messages related to the particular form. The Help is accessed by clicking on a Help button on the data input forms. This causes a scrolling list of the Help subjects for that data input form to be displayed (Fig. 4). The user then selects the desired subject by clicking on that subject in the scrolling list. The Help message for that subject is then displayed in the scrollable Definition window.

The Help information included covers general instructions on how to use the features of the system and explanations concerning the various data entry parameters that exist on the form. The Help information also includes the proper abbreviations, etc. that should be used on the forms. Also, in some cases, there is technical information about the Orbiter/Spacelab vehicle that applies to a data input parameter, such as possible acceleration g-levels that can be encountered on-orbit.

Data Input Methods

There are several different methods in FORCS for data input. These methods are as follows:

1. Spreadsheet (grid)
2. Text Boxes
3. Selection Lists
4. Pop-Up Windows
5. Check Boxes
6. Radio Buttons

The majority of data input is through the Step Data form, which is a spreadsheet type of input. After experimentation with different ways to input this type of data, it was concluded that the spreadsheet format was the most efficient method. The Step Data form can be scrolled back and forth horizontally (there is no need for vertical scrolling) through the steps. Data can be input into the step data cells in any order the user desires.

The next most used method of data entry is the text box. It is used extensively on all of the data input forms other than the Step Data form. Text boxes are also used on the pop-up windows, which are optional

methods of inputting data on the Step Data input form.

Scrolling Selection Lists are used with the Header form to assist the user in specifying any Predecessor, Sequenced, or Concurrent FO operations requirements. The user may enter the information (Experiment Acronym and FO Number) directly on the Header form or select the appropriate FOs from a scrollable list maintained by the system. For a user operating in the local environment with the user's own local FO file maintained within the software application, the FO list is automatically updated as new FOs are added by the user. The list may be initialized with FOs for a particular mission from other PI/PEDs by the FORCS system administrators prior to delivering a FORCS application to the PI/PEDs. When the user is running as a remote user linked to the centralized Oracle database, the FO list will be obtained automatically from the database for the mission/version that the user logs on under.

Selection lists are also used with the Step Titles/Descriptions, Comments, and FO Expansion Sheet forms. The information in these selection lists is all based on the inputs of the user in defining an FO and is dynamically updated as the information is added by the user. The information in these selection lists may be added to, modified, or deleted by the user at any time.

Pop-up windows are used with the Step Data input form as an aid for an inexperienced user in inputting the data in the proper format. Having the data formatted properly is a major factor in improving the efficiency of the payload integration mission planners in developing the integrated payload activity timeline. This also allows the use of the data to be input automatically into the timeline scheduling software. As the user gets familiar with the proper format for the data, he/she can bypass the use of the pop-up windows.

Pop-up windows are associated with the spreadsheet cells on the Step Data input form. The user can access a pop-up window by holding down the Option key while clicking with the mouse on the desired cell. An example of a pop-up window is shown in Fig. 5.

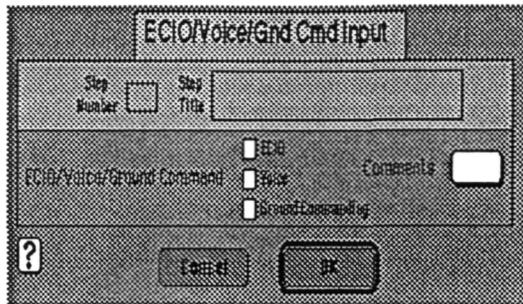


Figure 5. Step Data Pop-Up Window With Check Boxes

Figure 5 also illustrates the use of check boxes. The user may click on any of the check boxes to select options that he/she desires. In the case of check boxes, multiple options may be selected. In the case of radio buttons, only one of several options are possible. This provides a means of controlling valid inputs that the user can make.

Automatic Features

There are a number of features incorporated in the FORCS to automatically perform certain operations to assist the user. These features range from automatic manipulation of input data to automatic formatting of the information displayed on the FO output forms generated by the system.

All times maintained in the system are in the DD/HH:MM format. The user may chose to input the times in any of several different formats, including time in minutes, and the sytem will automatically convert the times after they are input into the standard format. This frees the user from having to manually convert times if he/she is more comfortable in working with a particular time format.

Reference letters (letters of the alphabet) are used on the Step Data output form to reference back to a comment on the Comments Sheet. When comments are entered, the system automatically assigns reference letters to them in sequential order. If a comment is subsequently deleted, the other comments are automatically re-labeled and updated wherever the comments have been used on the Header and Step Data input and output forms.

Using certain standard formats for the data in the cells on the Step Data form greatly facilitates the job of the mission planners. When the user uses the pop-up windows for input of this data, the system automatically inserts the information in the proper format. Once the user understands what the proper format is, he/she may bypass the pop-up windows and enter the information directly into the cells on the Step Data form. Examples of the desired entries for constraints are "L" for Lights, "W" for Orbiter Water Dump, and "T" for Vernier Thruster Particulate Contamination constraints.

Prerequisite and Concurrent FO identification information (Experiment Acronym and FO Number) is displayed in the Header section of the FO Sheet output form. The form has sufficient space to show the information for only one FO. When more than one entry is required for either one of these, the system automatically creates a comment. The reference letter for the comment is then displayed in the appropriate place on the FO Sheet output.

There are several places where the user has the option of inputting preferred, minimum, and maximum values. These include Number of Performances, Performance Delays, Step Delays, and Step Durations. If the user inputs values in addition to the preferred, the system will automatically check that the proper relationships (preferred \leq maximum, preferred \geq minimum, minimum $<$ maximum) exist in the input. If an error was made by the user, the system will

display an error message with the error identified. The user can then correct the error or he/she may chose to accept the input and ignore the warning.

If the user has a requirement to use the Onboard Experiment Applications Software (ECAS), certain information must be provided for the timeline generation process. The information required includes the Number of Subordinate Timelines To Be Used, the ECAS Task Size, and the ECAS Task Label. The Experiment Scheduling Program (ESP) has certain constraints for the format of this information, as follows:

Number of STL Buffers <= 5
ECAS Task Size <= 5 characters
ECAS Task Label <= 4 characters

The system will automatically check the user inputs to verify that the values for the inputs do not exceed the format constraints. If a constraint is exceeded, the system will display an error message with the error identified.

There are some unique features included for handling footnotes used on the FO Expansion Sheet. A given footnote may be applicable to one or more FO steps on the FO Expansion Sheet. Also, multiple footnotes may be applicable to a step or steps and general footnotes may be defined that are applicable to all steps. When there are multiple pages of the FO Expansion Sheet, the system will automatically place the footnotes at the bottom of only the pages where the applicable step is printed. The general footnotes will be printed automatically on each page of the FO Expansion Sheet.

Editing Features

If the user needs to edit information on an FO stored in the database (remote mode) or in the file for a standalone application, the user only needs to load the particular forms containing the information to be edited. This can save considerable amounts of time, especially when working with the central Oracle database.

The system has a number of editing features, ranging from the standard word processing text editing functions to special editing features of the information on the Step Data input form.

There are a number of ways to edit information on the Step Data input form. Situations may occur where an FO has a step or series of steps that may be repeated several times in the FO. All of the information for a step or series of steps (represented by columns of information in the Step Data input form spreadsheet) may be copied with simple commands and placed where desired. Using standard spreadsheet procedures, the user first selects (highlights) the desired steps (columns) using the mouse and then selects the Copy command from the menu bar to copy the information selected. Then, he/she uses the mouse in conjunction with the Paste command to define where to place the steps copied. All of the information for the steps are copied to the new locations.

The user has the option during editing operations to have the Step Titles/Descriptions linked or not linked to the Step Data when manipulating (adding, inserting, deleting) the steps. This is convenient if the user needs to insert some new steps in a previously defined set of steps and maintain the step titles/descriptions of the original steps.

In some cases it may be desired to copy only portions of the data for a step to other steps. An example of this situation is when certain resource values (for instance, electrical power) or constraint conditions may be applicable for a number of contiguous steps. Rather than entering the same information for each of the steps, one step at a time, the user can use the Row Editor. The pop-up window for the Row Editor is shown in Figure 6.

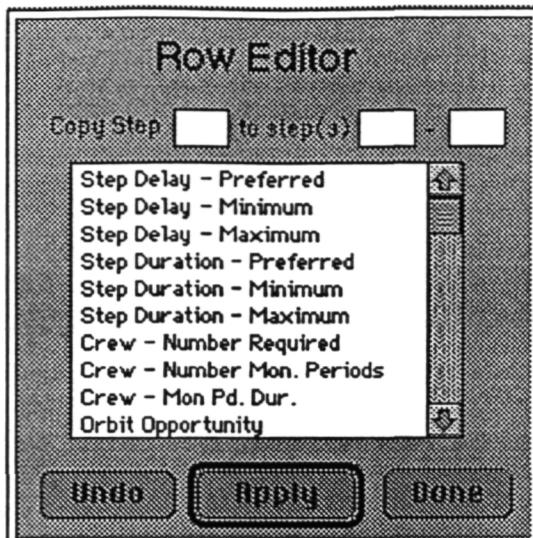


Figure 6. Row Editor Pop-Up Window

The user first inserts the Step Number in the Copy Step input text box in the upper portion of the form that he/she wishes to copy information from. Next, the beginning and ending Step Numbers of the contiguous block of the set of steps he/she wishes to copy to are entered in the appropriate text input boxes. The user then selects the parameters in the scrolling list he/she wants copied and, finally, selects the Apply button to implement the copy operation.

The user may change experiment identifier information (Experiment Name and Experiment Acronym) or FO identifier information (FO Number, FO Name, FO Acronym). This can be very useful when the same operations are performed for two or more different FOs. Rather than inputting the FO information redundantly, it can be easier to create one FO, change the identifier information, and then re-save the FO. This can be easily done with the capabilities provided by the FORCS.

Outputs

The user may get hard copy outputs of the standard FO forms. Considerable flexibility is provided using the Print options available. The user may print all of the forms associated with an FO or he/she may select any subset desired. For

example if the user is only interested in the FO Expansion Sheet, that can be selected. In addition, the user may also select certain experiments and /or FOs to print out.

In the past, numerous hard copies of FOs were reproduced for major Spacelab program milestone reviews, such as PDR (Preliminary Design Review), CDR (Critical Design Review), etc. This is not only expensive, but also inefficient. A desired goal is to be able to review electronic documents via Internet. TBE is currently taking the LMS (Life Science and Microgravity Spacelab) mission as a trial case, converting the baselined FOs into html (hypertext markup language) files and storing the files in a World Wide Web (WWW) server. The reviewer can access the applicable documents by using a commercial available browser, such as Mosaic or Netscape to view the documents through Internet. A corresponding Postscript file will also be available for users to download to their local computers for printing.

CONCLUSION

Previously, the Spacelab program has used a paper-based system for defining and collecting payload on-orbit activity timing requirements from the P/PEs. Now with the proliferation of personal computers and the interconnectivity provided by readily accessible data networks, it is practical to implement electronic methods for collecting and storing these requirements. The FORCS is an operational system that has been developed to collect and store Spacelab payload on-orbit activity planning requirements. The development effort for FORCS has progressed through several different approaches for interfacing with the user in specifying his/her requirements and has evolved to the present concept which appears to be a very suitable type of interface. Using the FORCS software, which includes a number of aids for the user to help in defining his/her requirements, greatly facilitates the process by eliminating

several time consuming manual steps that were necessary to get the information into the computer. FORCS also provides the information in a readily accessible form via the database and provides easier mechanisms for keeping the data updated.

REFERENCE

Cutts, Dannie E. and Widgren, Brian K. "The Use Of Artificial Intelligence Techniques To Improve The Multiple Payload Integration Process", 1992 Goddard Conference On Space Applications Of Artificial Intelligence, Goddard Space Flight Center, Greenbelt, Maryland, May 5-6, 1992.