User Manual
for
Aircraft Repair Model Simulation
Software version 2.0

CS 422 Software Engineering Principles

December 6, 2000

Goldrush-Team 9:

Nick Capron

Kyle Klicker

Carina Lansing

Jonathan McPherson

Andy Reynolds

Dave Stone

Russell Swannack

Miguel Vilchez

Kelly Yearout

Manual Version 2.0

This document describes the Aircraft Repair Model Simulation in partial fulfillment of the requirements for WSU-CS 422 Software Engineering Principles class. This document is subject to the restrictions listed within.

Abstract

The goal of Aircraft Repair Model Simulation (ARMS) is to compare various maintenance scheduling alternatives in order to optimize repair times and downtime costs. Seven types of commercial aircraft are evaluated using three different scheduling techniques. The program provides a graphical user interface, so the user can quickly and easily set parameters, run the simulation, compare scheduling alternatives, and view results.

This document is the User Manual for ARMS software version 2.0. It is designed to allow users to quickly access information on the various functions of ARMS by the function name. Each function has its own sub-section with information including the purpose of the function, materials needed, preparations for executing the function, inputs to the function, cautions and warnings regarding the function, invocation of the function, how to suspend or quit the function, output from the function, error conditions associated with the function, and any information that is related to the function. The format of this document complies with IEEE standard 1063-1987 [1] for software user documentation.

Restrictions
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation, to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Warranties
Team Goldrush makes no warranties, either express or implied, regarding ARMS, the ARMS user manual, and their fitness for any specific purpose. Team Goldrush provides ARMS and user manual as-is. Team Goldrush is not responsible for any adverse effect ARMS may have on a computer's hardware/software, computer peripherals, data storage devices, property, and persons resulting from use of ARMS and its user manual. Some states disallow the exclusion of implied warranties. Certain exclusions may not apply to you. Specific legal rights are provided by this warranty. There may be additional rights you have which vary from state to state.

Table of Contents

1 Introduction ***4***
1.1 Audience 4
1.2 Applicability 4
1.3 Purpose 4
1.4 User Manual Usage 5
1.5 Related Documents 5
1.6 Conventions And Terms 5
1.7 If You Have A Problem 5

2 Installing ARMS ***6***
2.1 Before You Begin 6
2.2 Unpacking ARMS 6
2.3 Compiling ARMS 6
2.4 Running ARMS 7

3 Using ARMS ***7***
3.1 Add Event 10
3.2 Create New Queue Discipline 12
3.3 Delete Item 12
3.4 Pause 13
3.5 Restart Application 15
3.6 Resume 16
3.7 Simulation Rate 17
3.8 Start 18

4 Error Messages, Known Problems, Error Recovery ***19***
4.1 Input Screen Error Messages and Error Recovery 20
4.2 Progress Screen Error Messages and Error Recovery 20
4.3 Report Screen Error Messages and Error Recovery 20
4.4 Add Event Error Messages and Error Recovery 20
4.5 Known Problems 20

5 References ***20***

6 Glossary ***21***

APPENDIX A: Examples and Additional Information ***22***
A.1 Queue Discipline – Single 22
A.2 Queue Discipline – Dual 26
A.3 Queue Discipline – Dual Assigned 27
A.4 The Seven Aircraft Types 29
A.5 Queue Disciplines 30

Table of Figures

Figure 1. Input Screen 7
Figure 2. Progress Screen 8
Figure 3. Report Screen 9
Figure 4. Add Event 10
Figure A-1. Input Screen - Single 22
Figure A-2. Progress Screen - Single 23
Figure A-3. Add Event-Event 23
Figure A-4. Add Event – Aircraft Type 24
Figure A-5. Add Event - Day 24
Figure A-6. Deleting an Event 25
Figure A-7. Report Screen - Single 25
Figure A-8. Input Screen - Dual 26
Figure A-9. Deleting an event 27
Figure A-10. Input Screen – Dual Assigned 28
Figure A-11. Deleting an Event – Dual assigned 28

List of Tables

Table A-1. The Seven Aircraft Types. 29

1 Introduction

Aircraft Repair Model Simulation, ARMS, is a tool to help users design aircraft maintenance facilities. This manual will help them become familiar with ARMS and its many features. The following section lists the requirements for running ARMS and explains the format of the document so as to allow easiest reference for all types of users.

1.1 Audience

This manual will help users become familiar with the many features of ARMS. Users should be somewhat familiar with using the Linux operating system to execute programs. Users should also have an understanding of the aircraft repair process. This will allow users to take full advantage of ARMS.

1.2 Applicability

This manual, version 2.0, covers ARMS version 2.0 only. Previous and future versions of ARMS may have some similarity in functionality to ARMS version 2.0, but to truly learn ARMS please get the appropriate version of the ARMS manual for your version of ARMS. ARMS is intended to operate on a PC using the Linux operating system (kernel 2.2 or newer) with 10 megabytes of hard drive space and 32 megabytes of random access memory. Java 1.3 is also required for ARMS to function correctly.

1.3 Purpose

The Aircraft Repair Model Simulation is a powerful tool designed to aid in the understanding of the effects of different arrangements of aircraft repair facilities. It allows users to simulate repair facilities and see the effects on downtime cost and the effectiveness of the repair facility arrangement. ARMS is easy to use, so users will be able to sit down and try it out right away. ARMS is functionally powerful, and this manual will give a detailed explanation of the many features it contains. The repair process and fundamental user requirements are defined in [2] and [3].

1.4 User Manual Usage

Sections two through five will help users become familiar with ARMS. Section two gives a brief overview on how to install ARMS. For quick reference on any function, section three lists functions alphabetically by name. Section four has information concerning error conditions. Appendix A contains a walk-through of the three different queuing disciplines.

1.5 Related Documents

For background information related to ARMS, read srs_gold_V3.6 [4], dnb_gold_V1.5 [5], and Gold9-ARMS-Test-Report-V1.5.pdf [6] found at www.tricity.wsu.edu/~goldrush. These documents will give insight to the production and history of ARMS.

For more help with installation of ARMS, the README file distributed as part of the ARMS software has detailed instructions on how to install ARMS.

1.6 Conventions And Terms

The ARMS user manual follows IEEE standard 1063-1987 [1] for user manuals. It uses the reference mode style to explain ARMS functionality.

ARMS uses three different GUIs, an Input Screen, a Progress Screen, and a Report Screen. Users can access different functions from each of these screens. This document will refer to these screens by the aforementioned names. Screen captures are located at the beginning of section three as Figures 1, 2, and 3.

Throughout this document, buttons appearing on any of the three screens will have their name, exactly as it appears on the screen, enclosed in quotes.

Labels appearing on the three screens will be underlined in this document.

Commands to the Linux shell and items to be placed into Linux configuration files will be italicized.

1.7 If You Have A Problem

If you should have any problem related to ARMS contact team Goldrush at goldrush@tricity.wsu.edu, or visit our website, www.tricity.wsu.edu/~goldrush, for online help.

If you have any suggestions on how ARMS could be improved, please email us at goldrush@tricity.wsu.edu, and include "KyleFixIt" in the subject field. For errors and improvements to the ARMS user manual, please email goldrush@tricity.wsu.edu, and include "AndyTypeIt" in the subject line.

2 Installing ARMS

This section will help with, installation and running ARMS. For more complete information on how to install ARMS, see the README file that comes with ARMS version 2.0.

2.1 Before You Begin

Before you install ARMS, certain requirements must be met:

- A computer running Linux with kernel version 2.2 and higher
- A copy of ARMS, (ARMS.tar.gz)
- Java 1.3 installed on your computer.

ARMS also needs several variables set in you shell resource file. If you have trouble compiling ARMS try replacing ~ with the absolute path. The following are examples for the bash shell:

```
PATH="$PATH:/usr/java/jdk1.3/bin"
CLASSPATH="$CLASSPATH:/usr/java/jdk1.3"
CLASSPATH="$CLASSPATH:~/Version2.0/lib/ext/AbsoluteLayout.jar"
export CLASSPATH
LD_LIBRARY_PATH="$LD_LIBRARY_PATH:~/Version2.0/lib"
export LD_LIBRARY_PATH
```

2.2 Unpacking ARMS

Before you can install ARMS, it must be unpacked. Open a shell window, and change to the directory where the ARMS.tar.gz resides. Type the command *tar xzvf ARMS.tar.gz*, and press enter. This will create a new directory named Version2.0. ARMS is now ready to be compiled.

2.3 Compiling ARMS

To compile ARMS, first change to the Version2.0 directory. Type *make*, and ARMS will compile. If you have any problems please email goldrush@tricity.wsu.edu. When ARMS is finished compiling it is ready to use.

2.4 Running ARMS

To run ARMS make sure your current working directory is Version2.0, then type *java ARMS*. ARMS will start, and you can begin using it.

3 Using ARMS

This section covers the different functions a user may interact with when using ARMS. Functions are listed alphabetically by the function's name. The various functions are accessed from the three screens ARMS uses: the Input Screen shown in Figure 1, the Progress Screen shown in Figure 2, and the Report Screen shown in Figure 3.

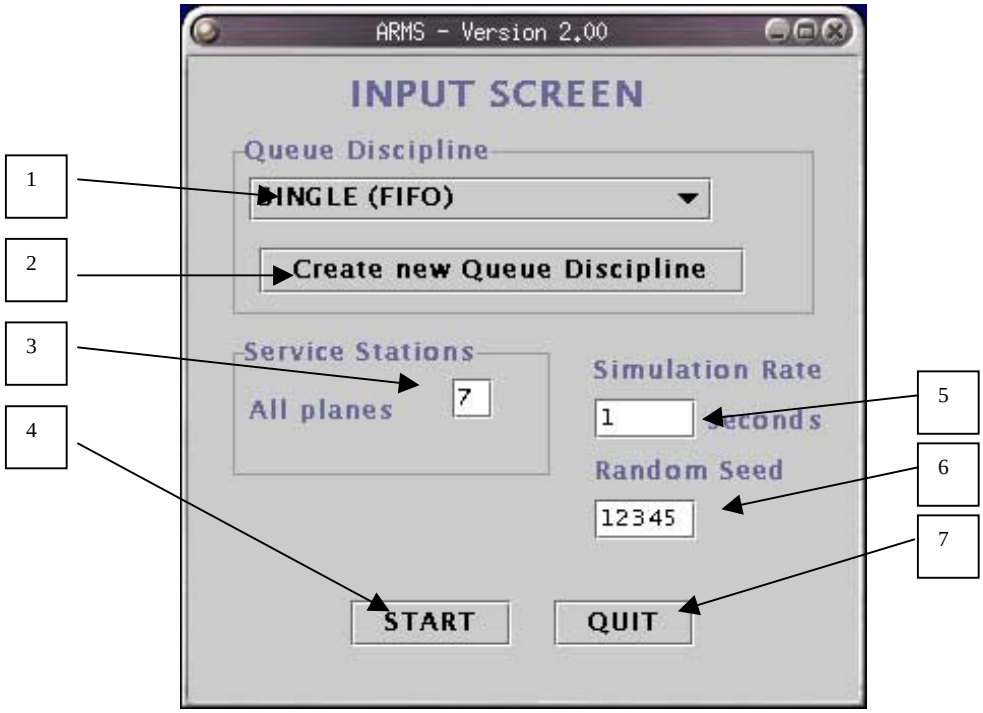


Figure 1. Input Screen

1. Queue discipline drop-down menu (Section 3.8.4).
2. "Create New Queue Discipline" button (Section 3.2).
3. Service Stations entry box (Section 3.8.4).
4. "START" button (Section 3.8).
5. Simulation Rate entry box (Section 3.8.4).
6. Random Seed entry box (Section 3.8.4)
7. "QUIT" button.

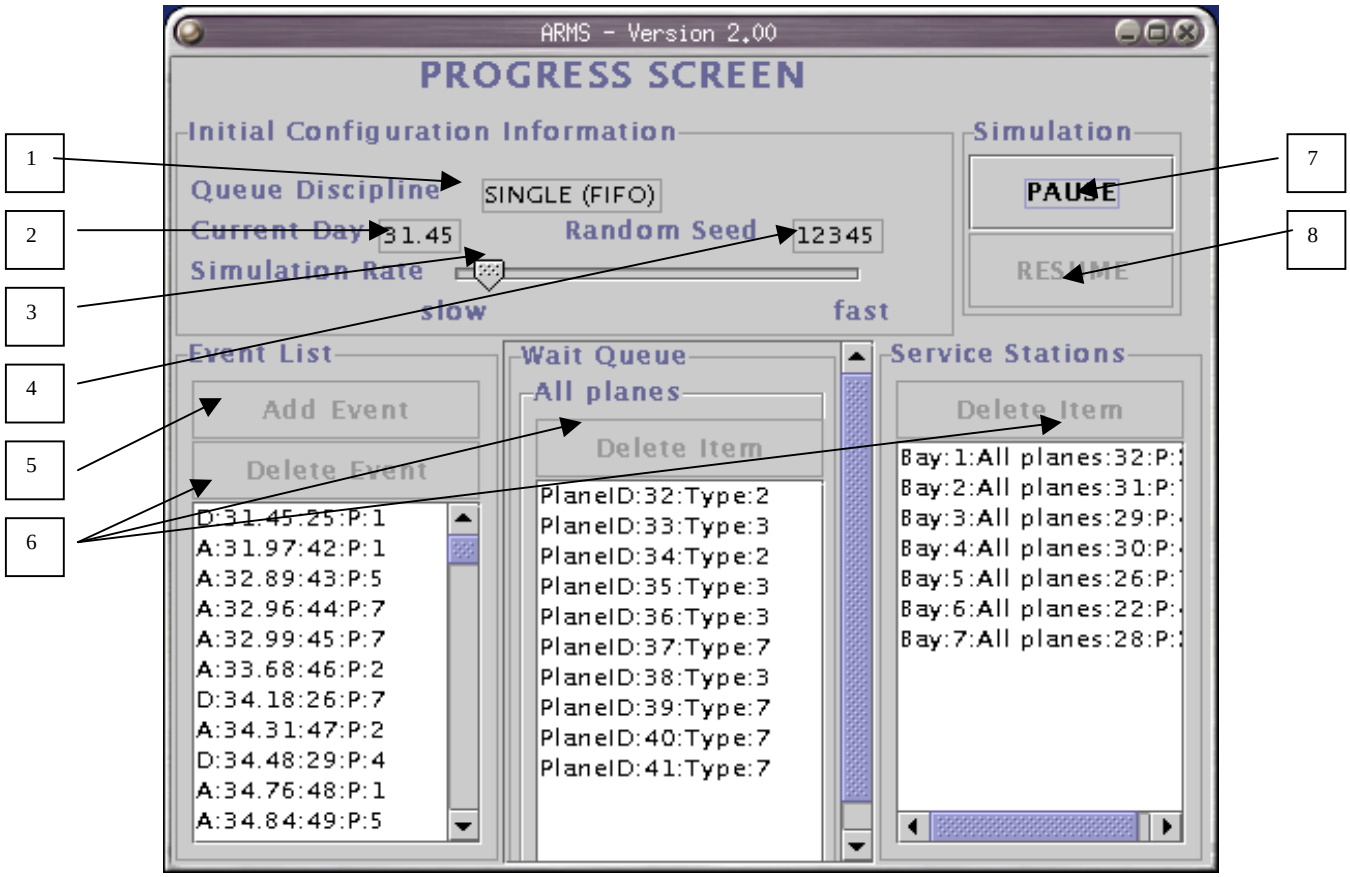


Figure 2. Progress Screen

1. Display of the Queue Discipline.
2. Display of the Current Day.
3. Simulation Rate slider (Section 3.7).
4. Display of the Random Seed.
5. “Add Event” button (Section 3.1).
6. “Delete Event”/”Delete Items” buttons (Section 3.2).
7. “PAUSE” button (Section 3.3).
8. “RESUME” Button (Section 3.6).

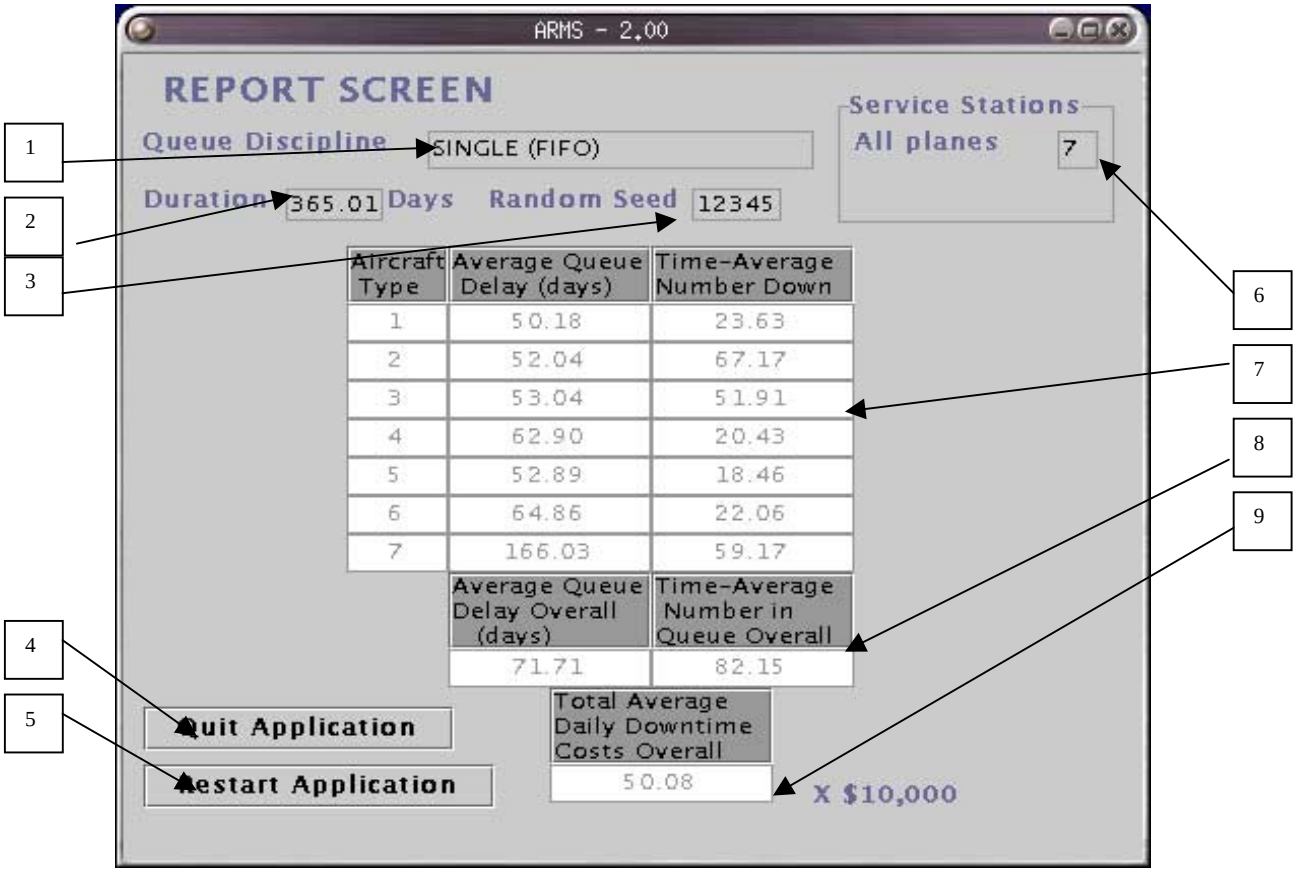


Figure 3. Report Screen

1. Display of the Queue Discipline.
2. Display of the simulation length.
3. Display of the Random Seed.
4. "Quit Application" button.
5. "Restart Application" button (Section 3.5).
6. Display of number of Service Stations used.
7. Display of simulation results by aircraft type.
8. Averages for all aircraft types.
9. Average downtime cost.

3.1 Add Event

The "Add Event" function allows users to enter a new event into the Event List. It can be accessed from the Progress Screen, see Figure 2.

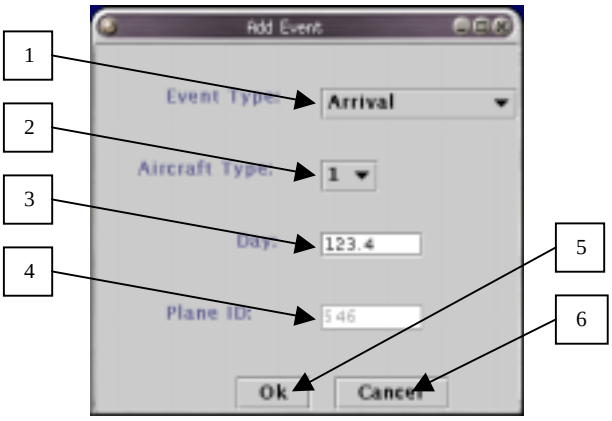
3.1.1 Purpose

The "Add Event" function adds an event to the Event List. This allows users to customize ARMS to simulate certain conditions that might arise in an aircraft repair facility.

3.1.2 Materials

If you would like more information on the seven aircraft types see Appendix A Section A.4. It contains information on each of the seven aircraft.

Figure 4 shows the "Add Event" dialog box.



1. Event Type drop-down menu.
2. Aircraft Type drop-down menu.
3. Day on which event will occur.
4. Plane ID number, assigned by ARMS.
5. "Ok" button, closes the Insert Event dialog and adds event to its list.
6. "Cancel" button, closes the Insert Event dialog but does not add the event to its list.

Figure 4. Add Event

3.1.3 Preparations

Before an event can be added, the simulation must be in paused mode, see Section 3.4.

3.1.4 Inputs

When adding an event, there are three pieces of data that must be entered: the event type, the aircraft type, and the day that the event occurs. There are three different event types: arrival, departure, and end simulation. There are seven different aircraft types, which are represented by the numbers 1-7. The day on which an event occurs is a number greater than 0.00 and less than the planned duration of the simulation.

3.1.5 Cautions and Warnings

When adding an event make sure that, multiple arrival, or departure events do not exist for the same aircraft. Use the delete function to remove multiple events. Event day must be greater than current simulation day.

3.1.6 Invocation

The "Add Event" function is accessed from the Progress Screen by clicking a button labeled "Add Event". Simply press this button to invoke the function. This causes a dialog box to appear on the screen. The dialog box has two drop down menus and a text box for entering the inputs. Select the desired input, and then click "Ok".

3.1.7 Suspension of Operations

The "Add Item" function cannot be suspended.

3.1.8 Termination of Operation

The insert event dialog box has a button labeled "Cancel". Clicking this button will close the box without adding an event.

3.1.9 Output

After the "Add Event" dialog box closes, the new event will appear in the Event List.

3.1.10 Error Conditions

If the number entered for the time is greater than 364.99 it will not be included in the statistics generated by ARMS. The "Add Event" function will not execute with incomplete or erroneous values entered, both of which will bring up error dialogs.

3.1.11 Related Information

If a user wants the simulation to end before the planned duration is up, add an "endsim" event at the current time.

The information shown in the Event List is formatted as follows:

- Items of information are separated by colons
- The first field is the type of event ('A' for arrival and 'D' for departure)
- The second field is the time at which the event will occur
- The third field is a unique identification number assigned to each aircraft by ARMS
- The last two fields are first a 'P' denoting plane and then the aircraft type.

3.2 Create New Queue Discipline

The “Create New Queue Discipline” button will allow users to add new aircraft repair facility setups, referred to as queuing disciplines in this document. It is accessed from the Input Screen shown in Figure 1. It is not implemented in this version of ARMS.

3.3 Delete Item

ARMS allows users to delete events/items from the Event List, the Wait Queue, and the Service Stations. The difference between an event and an item is only the list to which it belongs, if in the Event List then it is called an event, or if in the Wait Queue or Service Stations it is referred to as an item. The rest of this section will refer to both as “Delete Event”. The ”Delete Event” function is accessed from the Progress Screen, which is shown in Figure 2.

3.3.1 Purpose

The “Delete Event” function removes an event from the Event List, the Wait Queue, or the Service Stations. This allows users to customize ARMS to simulate certain conditions that might arise in an aircraft repair facility.

3.3.2 Materials

There are no separate materials needed for this function.

3.3.3 Preparations

ARMS must first be in paused mode before an event can be deleted. An event from the Event List, the Wait Queue, or the Service Stations must then be selected.

3.3.4 Inputs

There is no input required for this function.

3.3.5 Cautions And Warnings

When deleting events, the user should note the following:

- If a departure event is deleted, the aircraft remains where it is until a new departure is scheduled.
- If aircraft are deleted from a Service Station list another plane is immediately moved into the service station.
- Deletes are not confirmed and cannot be undone.

3.3.6 Invocation

The “Delete Event” function is invoked by highlighting an event, by clicking on it, and then clicking the “Delete Event” button corresponding to the list the item is a member of.

3.3.7 Suspension of Operations

This function cannot be suspended.

3.3.8 Termination of Operations

Once an event is highlighted and the “Delete Item” button is clicked, the delete function executes and cannot be terminated.

3.3.9 Outputs

The highlighted event will be removed from its list when the ”Delete Item” button is pressed.

3.3.10 Error Conditions

There are no error conditions associated with this event.

3.3.11 Related Information

The information shown in the Event List is formatted as follows:

- Items of information are separated by colons
- The first field is the type of event (‘A’ for arrival and ‘D’ for departure)
- The second field is the time at which the event will occur
- The third field is a unique identification number assigned to each aircraft by ARMS
- The last field is the aircraft type.

The Wait Queue has the following format:

- The first two fields are the label PlaneID and the aircrafts unique identifier.
- The last two fields are the label Type and the aircraft type.

The Service Stations list has a slightly different format:

- The first and second fields are a label, Bay, followed by the bay number.
- The third field is a label of the Aircraft type: Wide-body or Regular for the dual and dual assigned queue disciplines and All planes if using the single queue discipline.
- The forth field is the aircrafts unique identifier.
- The last two fields are a ‘P’ denoting plane and the aircraft type.

3.4 Pause

The pause function temporarily suspends ARMS. Figure 2 shows the Progress Screen from where “PAUSE” button can be accessed.

3.4.1 Purpose

The pause function allows users to temporarily stop the simulation. This allows users to scan through the Event list, the Wait List, and Service Stations to get an idea of what's happening with the simulation, and to add and delete events as desired.

3.4.2 Materials

There are no separate materials needed for this function.

3.4.3 Preparations

The simulation must be in running mode for it to be paused.

3.4.4 Inputs

There is no input required for this function.

3.4.5 Cautions And Warnings

The pause function does not do anything that could invalidate the simulation.

3.4.6 Invocation

The pause function is invoked by clicking the "PAUSE" button that appears on the Progress Screen.

3.4.7 Suspension of Operations

This function is designed to suspend the operation of the simulation.

3.4.8 Termination of Operations

The resume function is the inverse of the pause function.

3.4.9 Outputs

There is no output from the pause function.

3.4.10 Error Conditions

There are no error conditions associated with this event.

3.4.11 Related Information

To add or delete, events the user must first click the "PUASE" button. ARMS starts in paused mode.

3.5 Restart Application

Once the simulation ends, users are allowed to re-run ARMS without exiting. The “Restart Application” function is accessed from the Report Screen, which is shown on Figure 3.

3.5.1 Purpose

After the simulation has run the results are shown on the Report Screen. Users can then use the “Restart Application” button to restart ARMS.

3.5.2 Materials

There are no separate materials needed for this function.

3.5.3 Preparations

There are no preparations to make before invoking the “Restart Application” function.

3.5.4 Inputs

There is no input required for this function.

3.5.5 Cautions And Warnings

ARMS does not save the results shown on the Report Screen. Once the “Restart Application” button is pressed, the Report Screen will close. The user is responsible for recording any data they wished to be saved.

3.5.6 Invocation

The “Restart Application” function is invoked by clicking the “Restart Application” button that appears on the Report Screen.

3.5.7 Suspension of Operations

This function cannot be suspended.

3.5.8 Termination of Operations

This function cannot be terminated.

3.5.9 Outputs

When the “Restart Application” button is pressed the Report Screen will close and the Input Screen will open.

3.5.10 Error Conditions

There are no error conditions associated with this event.

3.6 Resume

A function is needed to move ARMS from a paused mode to running mode. The resume function is accessed from the Progress Screen, which is shown in Figure 2.

3.6.1 Purpose

The “RESUME” button lets users initiate or un-pause the simulation.

3.6.2 Materials

There are no separate materials needed for this function.

3.6.3 Preparations

There are no preparations to make before invoking the resume function.

3.6.4 Inputs

There is no input required for this function.

3.6.5 Cautions And Warnings

The resume function does nothing that could invalidate the results of the simulation.

3.6.6 Invocation

The resume function is invoked by clicking the “RESUME” button that appears on the Progress Screen.

3.6.7 Suspension of Operations

This function cannot be suspended.

3.6.8 Termination of Operations

The pause function is the inverse of the resume function.

3.6.9 Outputs

When the “RESUME” button is pressed, the text on the “Add Event” button, and the “Delete Event” and “Delete Item” buttons will be grayed out.

3.6.10 Error Conditions

There are no error conditions associated with this event.

3.6.11 Related Information

When the simulation is resumed and allowed to run to completion, day 365.00, the Progress Screen will close and the Report Screen will open. The Report Screen contains the statistical information produced by ARMS.

On the Report Screen there is a table with the seven aircraft types. The first column is the aircraft type number. The second column is the average number of days spent in the Wait Queue for all planes of that type. The third column is the average down time of all aircraft of the corresponding type.

Below the table of the seven aircraft types and related information are two boxes. The left is the average time, for all aircraft types, spent in the Wait Queue. The right box is the column is the average down time of all aircraft types.

Below these boxes is the total downtime cost.

3.7 Simulation Rate

It is sometimes desirable to increase or decrease the rate the simulation executes. Figure 2 shows the Progress Screen from which the simulation rate function is accessed.

3.7.1 Purpose

The simulation rate slider allows users to speed up or slow down the rate at which ARMS is executed. This allows users to quickly run through the simulation or watch every little detail as it happens.

3.7.2 Materials

There are no separate materials needed for this function.

3.7.3 Preparations

There are no preparations to make before invoking the simulation rate function.

3.7.4 Inputs

There is no input required for this function.

3.7.5 Cautions And Warnings

The simulation rate function does nothing that could invalidate the results of the simulation.

3.7.6 Invocation

The simulation rate function is invoked by clicking the slider net to the label Simulation Rate that appears on the Progress Screen. Moving the slider to the left will decrease the simulation rate and movement of the slider to the right will increase the simulation rate.

3.7.7 Suspension of Operations

This function cannot be suspended.

3.7.8 Termination of Operations

This function cannot be terminated.

3.7.9 Outputs

Events will appear to take place slower or faster in the Event List, the Wait Queue, and the Service Stations.

3.7.10 Error Conditions

There are no error conditions associated with this event.

3.8 Start

ARMS needs some information before it can start. The start function is accessed from the Input Screen as seen in Figure 1.

3.8.1 Purpose

The start function is invoked to initialize ARMS. It closes the Input Screen and opens the Progress Screen.

3.8.2 Materials

If you are unfamiliar with the queuing disciplines, see section A.5 for more information.

3.8.3 Preparations

ARMS requires several items to be set before the “START” button can be pressed.

3.8.4 Inputs

There are four different items to select before invoking the start function:

- Set the queue discipline using the drop-down menu
- Enter the number of service stations
- Enter a random seed. ARMS defaults to zero. Using the same random seed over will produce identical arrival events.
- Enter a number to set the initial simulation rate.

3.8.5 Cautions And Warnings

The user should make sure that:

- The total number of service stations does not exceed 10
- The random seed is between 1 and 99999
- The simulation rate is between 0.1 and 5.0

3.8.6 Invocation

The start function is invoked by pressing the button labeled ”START” that appears on the Input Screen.

3.8.7 Suspension of Operations

This function cannot be suspended.

3.8.8 Termination of Operations

The start function cannot be terminated.

3.8.9 Outputs

Once the “START” button is pressed the Input Screen will close and the Progress Screen will open.

3.8.10 Error Conditions

An error will occur if the user enters more than 10 total stations or if an invalid seed is inserted, an invalid random seed is entered, or an invalid simulation rate is entered. See Section 4.1 for more details on error conditions.

4 Error Messages, Known Problems, Error Recovery

This section will cover errors and problems with ARMS. Each of the three screens will be separated into sub-sections.

4.1 Input Screen Error Messages and Error Recovery

The Input Screen will open an error dialog under the following circumstances:

- More than 10 total service stations have been entered.
- An invalid random seed has been entered.
- The simulation rate is not between 0.1 and 5.0

These errors can be recovered from by first clicking "Ok" to dismiss the error dialog. Then entering no more than 10 service stations, a random seed between 1 and 99999, and a simulation rate between 0.1 and 5.0.

4.2 Progress Screen Error Messages and Error Recovery

There are no error messages associated with the Progress Screen.

4.3 Report Screen Error Messages and Error Recovery

There are no error messages associated with the Report Screen.

4.4 Add Event Error Messages and Error Recovery

The Add Item dialog will open an error dialog under the following circumstances:

- No or non-numerical data in the day field.
- Numbers less than 0.0 and grater than 365.0.

These errors can be recovered from by first clicking "Ok" to dismiss the error dialog. Then entering a number greater than 0.0 and less than 365.0.

4.5 Known Problems

ARMS will not work with the KDE window manager.

If problems are discovered please email a description of the problem to goldrush@tricity.wsu.edu and type "kylefixit" in the subject line.

5 References

1. IEEE-SA Standards Board, "IEEE Standard for Software User Documentation," IEEE Std 1063-1987, USA, December 10, 1987.
2. Law, Averill M. and David Kelton, "Simulation Modeling and Analysis," 3rd Edition, New York: McGraw-Hill Companies, October 1999.
3. Sheldon, F.T., "Project Requirements, CS422 Software Engineering Principles," http://www.eecs.wsu.edu/~sheldon/cs422.html Fall 2000.

4. Goldrush-Team 9, “Software Requirements Specification for Aircraft Repair Model Simulation,” Version 3.6, October 16, 2000.
5. Goldrush-Team 9, “Design Notebook for Aircraft Repair Model Simulation,” Version 1.5, November 8, 2000.
6. Goldrush-Team 9, “Test Report for Aircraft Repair Model Simulation,” Version 1.2, December 6, 2000.

6 Glossary

This glossary defines acronyms used in the ARMS User Manual.

ARMS – Aircraft Repair Model Simulation

GUI – Graphical User Interface

MB – megabyte, as in 1 million bytes of data storage

PC – Personal Computer

RAM – Random Access Memory

FIFO – First In First Out

Paused mode - While in paused mode ARMS does not process any events.

Running mode – While in running mode ARMS process events and updates lists.

Dialog Box – A small window that appears on the screen usually invoked by pressing a button, allowing a user to enter information required for the program to continue.

APPENDIX A: Examples and Additional Information

Appendix A contains example runs of ARMS for the three different queuing disciplines, and also a table of the seven different aircraft types.

A.1 Queue Discipline – Single

This section is an example run through of ARMS using the single queue discipline. Section A.5 gives more information on the single queue discipline.

A.1.1 Setting Up the Single Queue Discipline

ARMS' default queue discipline is the single queue discipline. Figure A-1 shows how the Input Screen should be set to use the single queue discipline.

1. Set the queue discipline to single using the drop-down menu.
2. Enter the number of bays (this needs to be less than or equal to ten).
3. Enter the simulation rate.
4. Enter the random seed. ARMS will generate a random seed; if you want to change it, type your random seed here.
5. Click the "START" button.

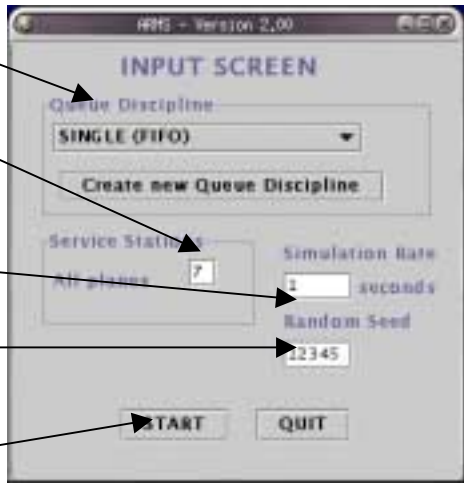


Figure A-1. Input Screen - Single

A.1.2 Viewing the Progress

When You press the "START" button on the Input Screen, the screen will close and the Progress Screen will open. Aircraft can be traced by their unique identifiers through the repair process. Arrivals of aircraft are shown in the Event List. Aircraft then move to a service station, if one is available, otherwise they go into an appropriate queue to wait for a bay to open.

1. Click the “RESUME” button to start the simulation running.

2. The current day is displayed.

3. You can change the rate of the simulation by clicking on and holding the Simulation Rate slider.

4. All arrival events are generated when ARMS starts and appear in the Event List.

5. Aircraft waiting for service are shown in the Wait Queue.

6. Aircraft being serviced are shown in the Service Stations list.

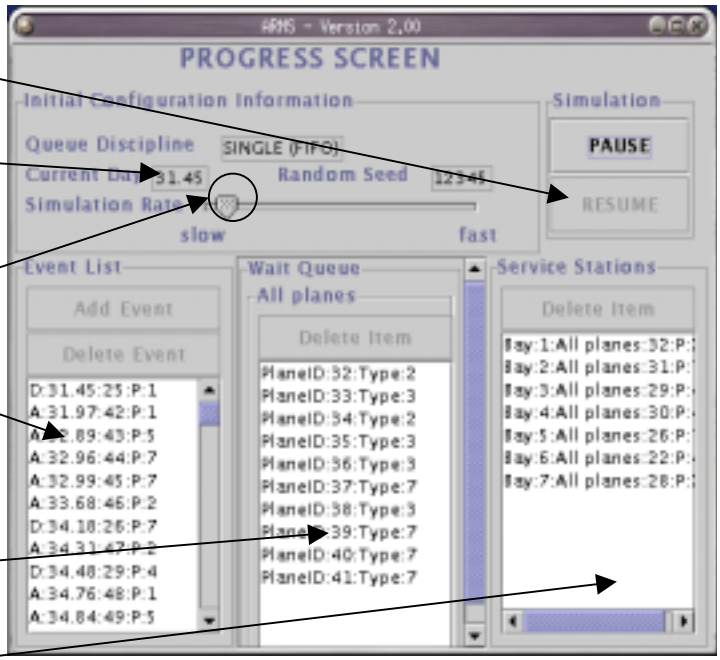


Figure A-2. Progress Screen - Single

A.1.3 Adding an Event

When you want to add an event first click the “PAUSE” button. Then click the “Add Event” button. This will open the Add Event dialog box as seen in Figures A-3, A-4, and A-5. There are three variables to set when adding an event.

Set the event type (Arrival, Departure, or End Simulation) using the drop-down menu.

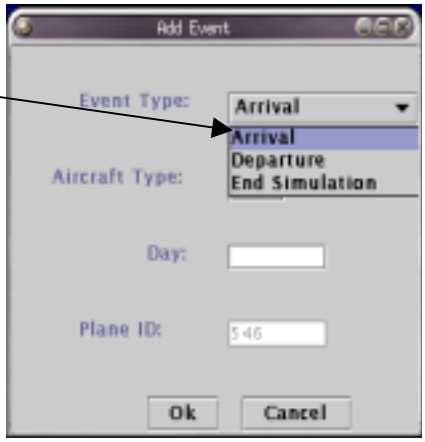


Figure A-3. Add Event-Event

Set the aircraft type (1-7) using the drop-down menu.

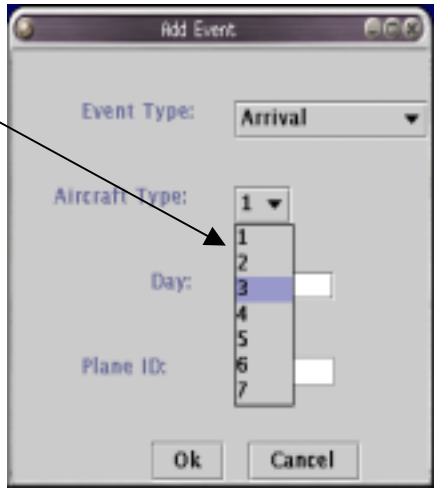


Figure A-4. Add Event – Aircraft Type

1. Enter the day that you want the event to happen (0.0-364.99).
2. Click "Ok" to add the event and close the Insert Event dialog.
3. Click "Cancel" to not add the event and close the "Add Event" dialog.

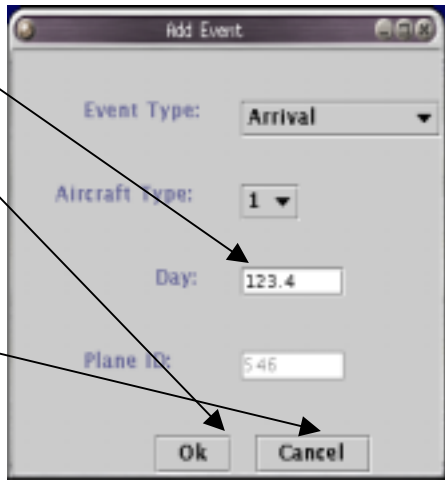


Figure A-5. Add Event - Day

A.1.4 Deleting an Event

ARMS allows events to be deleted from three different places: the Event List, the Wait Queue, or the Service Stations. Deletion is similar for each list. Figure A-6 shows a highlighted event.

1. First press the “PAUSE” button.
2. You can highlight an event by clicking on it.
3. Pressing the “Delete Event”/”Delete Item” button will delete the event.
4. Click the “RESUME” button.

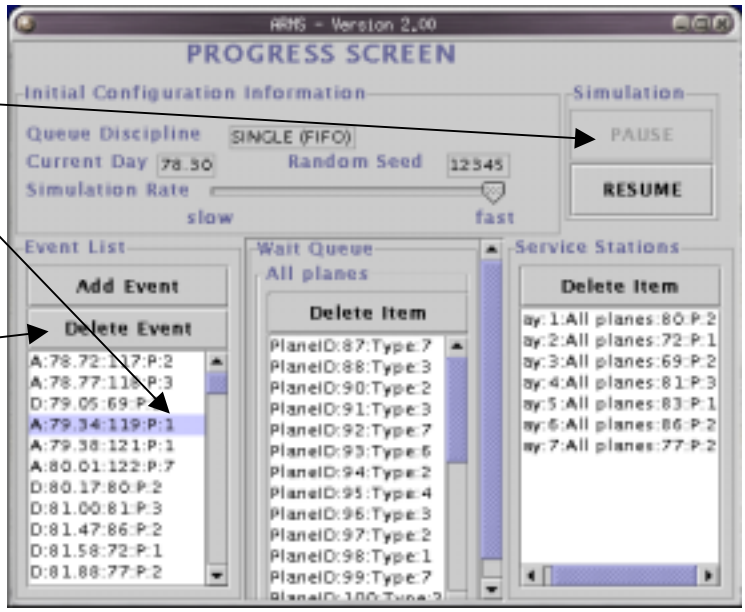


Figure A-6. Deleting an Event

A.1.5 The Report Screen

After the end simulation event occurs, the Progress Screen will close and the Report Screen will open. Figure A-7 shows the Report Screen.

1. Results of the simulation are displayed.
2. The “Quit Application” button will exit ARMS.
3. The “Restart Application” button will close the Report Screen and open the Input Screen.

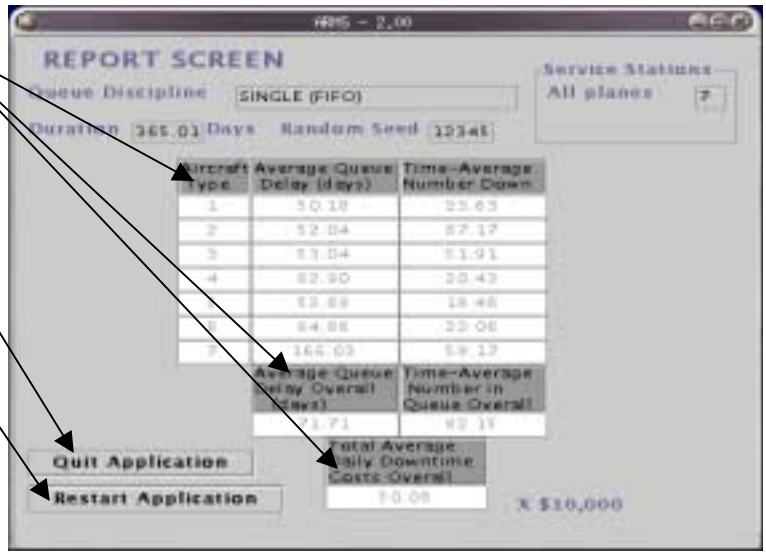


Figure A-7. Report Screen - Single

A.2 Queue Discipline – Dual

This section is an example run through of ARMS using the dual queue discipline. Section A.5 gives more information on the dual queue discipline. After the queue discipline is set and the simulation started there is no difference in how ARMS is used, so refer to sections A.1.2 through A.1.5 for information on viewing the progress, adding events, and the Report Screen.

A.2.1 Setting up the Dual Queue Discipline

Figure A-8 shows Input Screen. There are four variables to be set in order to run the Dual queue discipline.

1. Set the queue discipline to dual using the drop-down menu.
2. Enter the simulation rate.
3. Enter the number of Service Stations (this needs to be less than or equal to ten).
4. Enter the random seed. ARMS will generate a random seed; if you want to change it, type your random seed here.
5. Click the "START" button.

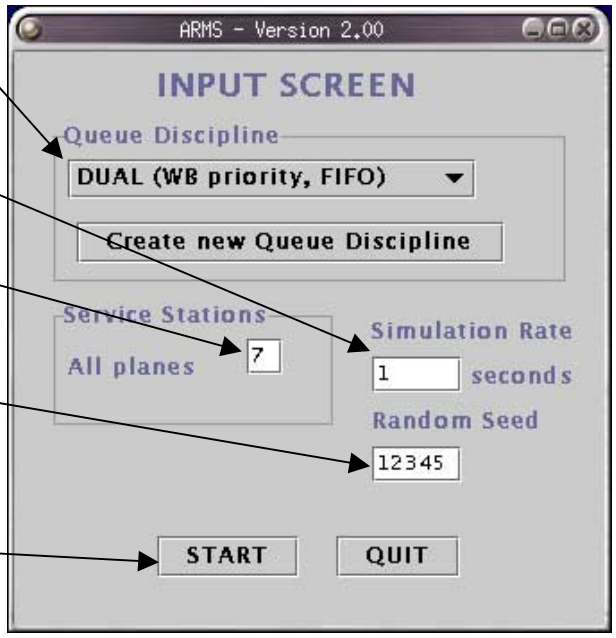


Figure A-8. Input Screen - Dual

A.2.2 Deleting an Event

The dual queue discipline shows separate queues for wide body and regular aircraft. This adds one more place an event can be deleted. Figure A-9 shows these separate queues.

1. First click the “PAUSE” button.

2. Highlight an event in one of the four lists.

3. Press the corresponding ”Delete Event”/”Delete Item” button.

4. Click the “RESUME” button.

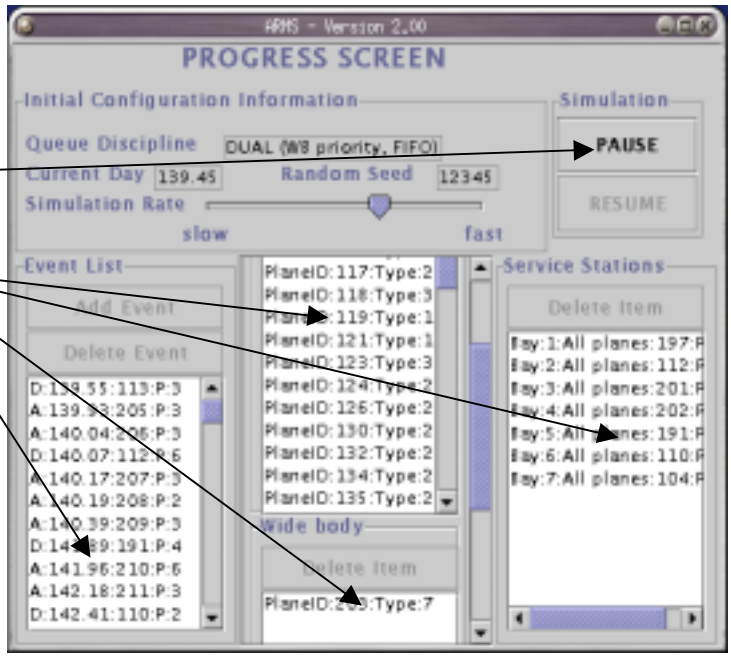


Figure A-9. Deleting an event

A.3 Queue Discipline – Dual Assigned

This section is an example run through of ARMS using the dual assigned queue discipline. Section A.5 gives more information on the dual assigned queue discipline. After the queue discipline is set and the simulation started there is no difference in how ARMS is used, so refer to sections A.1.2 through A.1.5 for information on viewing the progress, adding events, and the Report Screen.

A.3.1 Setting up the Dual Assigned Discipline

Figure A-10 shows the Input Screen. There are four variables to be set in order to run the Dual Assigned queue discipline.

1. Set the queue discipline to Dual (assigned bays, FIFO) using the drop-down menu.
2. Enter the simulation rate.
3. Enter the number of service stations for regular aircraft.
4. Enter the number of service stations for wide body aircraft (the total number of service stations must be less than ten.
5. Enter the random seed. ARMS will generate a random seed; if you want to change it, type your random seed here.
6. Click the "START" button.

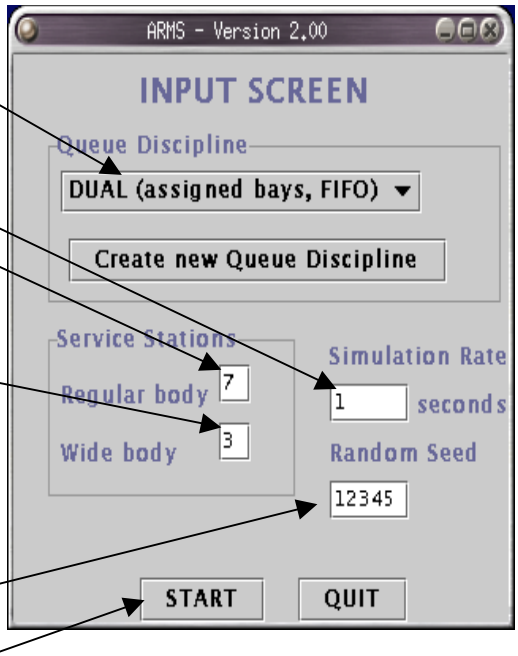


Figure A-10. Input Screen – Dual Assigned

A.3.2 Deleting an Event

The dual assigned queue discipline shows separate queues for wide body and regular aircraft. This adds one more place an event can be deleted. Figure A-11 shows these separate queues.

1. First click the "PAUSE" button.
2. Highlight an event in one of the four lists.
3. Press the corresponding "Delete Event"/"Delete Item" button.
4. Click the "RESUME" button.

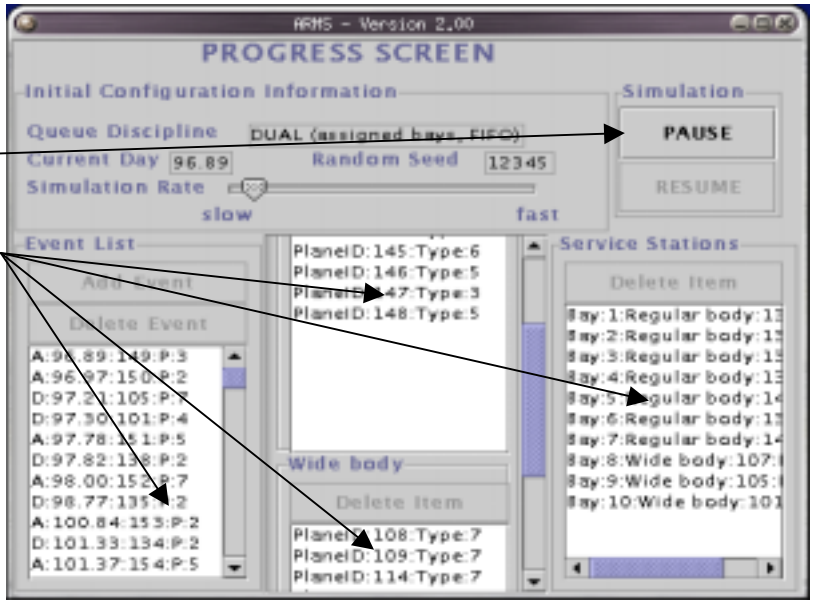


Figure A-11. Deleting an Event – Dual assigned

A.4 The Seven Aircraft Types

The following table contains information about the seven aircraft types.

Table A-1. The Seven Aircraft Types.

Plane type *(i)*	number of engines	*a(i)*	*A(i)*	*B(i)*	*P(I)*	*r(i)*	*c(i)*
1	4	8.1	0.7	2.1	0.30	2.1	2.1
2	3	2.9	0.9	1.8	0.26	1.8	1.7
3	2	3.6	0.8	1.6	0.18	1.6	1.0
4*	4	8.4	1.9	2.8	0.12	3.1	3.9
5	4	10.9	0.7	2.2	0.36	2.2	1.4
6	2	6.7	0.9	1.7	0.14	1.7	1.1
7*	3	3.0	1.6	2.0	0.21	2.8	3.7

Column	**Description**
i	An enumeration reference for the different sets of configuration data *(aircraft type)*.
Aircraft Type	Descriptive name that has a unique association with **i**.
Wide Body	The aircraft's designation as a wide-body aircraft type *(yes)* or a regular aircraft type *(no)*.
Number of Engines	The total number of engines this aircraft type has.
Arrival Interval	The mean number of days between successive arrival events *(used in conjunction with the exponential distribution to predict arrival times)*.
Inspection Time, Minimum	The minimum amount of days it takes for this aircraft to have one engine inspected *(used in conjunction with a uniform distribution to predict inspection times)*. This time will be halved for all non-initial engine inspections.

Inspection Time, Maximum .....The maximum amount of days it takes for this aircraft to have one engine inspected *(used in conjunction with a uniform distribution to predict inspection times).* This time will be halved for all non-initial engine inspections.

Repair Probability .....The individual probability that any single engine for this aircraft will require a repair of some kind. This probability will be halved for all non-initial engine inspections.

Repair Time.....The number of days that will be needed to complete a repair.

Daily Cost.....The cost per day for this aircraft to remain down. The aircraft is considered to be down the entire time it is in a queue or a service station. *(in* $10,000 *US Dollars).*

A.5 Queue Disciplines

Here is a brief description of the three standard queue disciplines.

Single - All waiting aircraft form a single, first-in/first-out (FIFO) queue. All stations (the number specified by the user) service all aircraft.

Dual - Wide-body and regular body aircraft wait in separate FIFO queues. The wide-body queue is given non-preemptive priority over the regular body queue. All stations (the number specified by the user) service all aircraft.

Dual assigned – Wide-body and regular body aircraft wait in separate FIFO queues.