

# **Test Report for Aircraft Repair Model Simulation**

Software version x.y.z

**CS 422 Software Engineering Principles**

**November 15, 2000**

## **Team 9: Goldrush**

**Nick Capron**

**Kyle Klicker**

**Carina Lansing**

**Jonathan McPherson**

**Andy Reynolds**

**Dave Stone**

**Russell Swannack**

**Miguel Vilchez**

**Kelly Yearout**

Version 1.2 (*DRAFT*)

## Abstract

This document will define and outline the test plan methodologies, and the resulting test report(s), applied by Team Goldrush on the Aircraft Repair Model Simulation (*ARMS*). All test results, regardless of the test, will have a detailed and compiled report.

The plan consists of testing the software against the Software Requirements Specification (SRS) and against the Users Manual (*UM*). Testing against the SRS is a validation process to ensure that the delivered program has met the customer's specified requirements and needs as identified in the Requirements Traceability Matrix (*RTM*). Testing against the UM is a means of verification that the delivered product performs and acts as stated.

The Test Report (*TR*) will have a primary focus on the finished ARMS software product. Each version of the ARMS software submitted for testing by the Design and Coding team will be evaluated according to the current stage of development for that version. Documentation, and its maintenance, occurs at each phase of the testing process. The ARMS application development, design, and coding is such as to allow for future extensibility, if so desired by the customer. Similar considerations were given to the GUI throughout design process as well. Any unimplemented ARMS extensibility options have not been considered for this test report, and subsequently not tested.

## Table of Contents

|          |   |                   |
|----------|---|-------------------|
| <b>1</b> | <b><i>Introduction.....</i></b>   | <b><i>iii</i></b> |
| 1.1      | Purpose .....   | 4                 |
| 1.2      | Scope .....   | 4                 |
| 1.3      | Definition, acronyms and abbreviations.....                             | 4                 |
| 1.4      | References.....   | 6                 |
| 1.5      | Overview.....   | 7                 |
| <b>2</b> | <b><i>Test-Design Specification .....</i></b>                           | <b><i>7</i></b>   |
| 2.1      | Purpose .....   | 7                 |
| 2.2      | Outline: TDS01 .....  | 7                 |
| 2.3      | Outline: TDS02.....   | 10                |
| 2.4      | Outline: TDS03.....   | 13                |
| 2.5      | Outline: TDS04.....   | 15                |
| 2.6      | Outline: TDS05.....   | 17                |
| <b>3</b> | <b><i>Test-Case Specification.....</i></b>                              | <b><i>19</i></b>  |
| 3.1      | Purpose .....   | 19                |
| 3.2      | Outline: TCS01 .....  | 20                |
| 3.3      | Outline: TCS02.....   | 23                |
| 3.4      | Outline: TCS03.....   | 26                |
| 3.5      | Outline: TCS04.....   | 28                |
| 3.6      | Outline: TCS05.....   | 32                |
| 3.7      | Outline: TCS06.....   | 36                |
| 3.8      | Outline: TCS07.....   | 40                |
| 3.9      | Outline: TCS08.....   | 43                |
| 3.10     | Outline: TCS09.....   | 46                |
| 3.11     | Outline: TCS10.....   | 49                |
| 3.12     | Outline: TCS11.....   | 52                |
| 3.13     | Outline: TCS12.....   | 55                |
| 3.14     | Outline: TCS13.....   | 58                |
| <b>4</b> | <b><i>Test-Summary Report .....</i></b>                                 | <b><i>60</i></b>  |
| 4.1      | Purpose .....   | 60                |
| 4.2      | Outline: TSR01 .....  | 60                |
|          | <b><i>APPENDIX A: The SRS Requirements Traceability Matrix.....</i></b> | <b><i>62</i></b>  |
|          | <b><i>APPENDIX B: The Individual Test Case Results.....</i></b>         | <b><i>65</i></b>  |
|          | <b><i>APPENDIX C: The Application Trace File (ATF).....</i></b>         | <b><i>66</i></b>  |

## Table of Figures

|  |    |
|--|----|
| Figure 1: Test Transmittal Database Entry Screen ..... | 64 |
| Figure 2: Test Incident Database Entry Screen .....    | 64 |

## List of Table

|  |    |
|--|----|
| Table 1: Software Requirements Specification - Requirements Traceability Matrix..... | 61 |
|--|----|

## **1 Introduction**

This section provides the basis for the Test Report (*TR*) and its importance in reference to the ARMS product. Any additional document artifacts, used as resources during the production of this document, are in the section for references.

### **1.1 Purpose**

The primary purpose of the TR is to define the processes and parameters used to verify and validate the ARMS project. The secondary, but of equal importance, purpose of the TR is to compile the results of all the tests applied to the ARMS program by the test team. Finally, the TR acts as the vehicle to demonstrate that the ARMS project has met the customer requirements.

### **1.2 Scope**

The specific intent of the test report is to provide descriptions, as utilized during the testing process, for each of the following:

- Any specialized tools and/or “test-only” software modules
- Code inspection techniques utilized
- GUI testing methodology
- Hardware systems, software environments, and facilities
- Module and system level approaches
- UM testing methodology

Each of the above items will generate documents and results whose criteria are strictly on a pass/fail basis. The design team will review and use these documents, in conjunction with the test engineers, to correct or modify program aspects in the event of a test failure. The ARMS project software, upon delivery to the customer, will have no test criteria resulting in a requirements’ failure.

### **1.3 Definition, acronyms and abbreviations**

This section includes explanations of terminology, abbreviations, and specialized acronyms used throughout the ARMS test report.

#### **1.3.1 Definitions**

This section is to define terminology that may be unfamiliar to the reader.

*Data Flow Diagram* - Provides an indication of how data transforms occur as the data moves through the system. They also depict the functions that transform the data.

*Design Level* - The design decomposition of the software item (e.g. - system, subsystem, program, or module).

*Pass/Fail Criteria* - Decision rules used to determine whether a software item or a software feature passes or fails a test.

*Product* - The end deliverable defined as the ARMS.

*Software Feature* - A distinguishing characteristic of a software item (e.g. - performance, portability, or functionality).

*Software Item* - Source code, object code, job control code, control data, or a collection of these items.

*Test* - (1) A set of one or more test cases, or (2) A set of one or more test procedures, or (3) A set of one or more test cases and procedures.

*Test Case Specification* - A document specifying inputs, predicted results, and a set of execution conditions for a test item.

*Test Design Specification* - A document specifying the details of the test approach for a software feature or combination of software features and identifying the associated tests.

*Test Incident Report* - A document reporting on any event that occurs during the testing process, which requires investigation.

*Test Item* - A software item, which is an object of testing.

*Test Log* - A chronological record of relevant details about the execution of tests.

*Test Plan* - A document describing the scope, approach, resources, and schedule of intended testing activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning.

*Test Procedure Specification* - A document specifying a sequence of actions for the execution of a test.

*Test Summary Report* - A document summarizing testing activities and results. It also contains an evaluation of the corresponding test items.

*Testing* - The process of analyzing a software item to detect the differences between existing and required condition (e.g.: bugs) and to evaluate the features of the software item.

### **1.3.2 Acronyms and Abbreviations**

This section is to define acronyms and abbreviations that may be unfamiliar to the reader.

*API* - Application Programming Interface

*ARMS* - Aircraft Repair Model Simulation

*ATF* - Application Trace File

*C* - The programming language ANSI-C

*DFD* - Data Flow Diagram

*ESEP* - Earth Science Enterprise Program

*GUI* - Graphical User Interface

*MB* - Megabyte, as in 1 million bytes of data storage

*PC* - Personal Computer

*RAM* - Random Access Memory

*RTM* - Requirements Traceability Matrix

*SRS* - Software Requirements Specification

*TR* - Test Report

## **1.4 References**

The following is a list of documents, texts, and web sites that were reviewed, or consulted throughout the activities of preparing the test plan and the test report.

1. IEEE-SA Standards Board, IEEE Standard for Software Test Documentation. IEEE Std 829-1983. USA, December 3, 1992
2. Law, Averill M. and David Kelton, Simulation Modeling and Analysis, 3rd Edition. New York: McGraw-Hill Companies, October 1999

3. Sheldon, Frederick T., “Project Requirements, CS422 Software Engineering Principles” <http://www.eecs.wsu.edu/~sheldon/cs422.html> (Items 5, 8 & 13) Fall 2000
4. Goldrush (Team 9): Software Requirements Specifications Document v3.6

## **1.5 Overview**

This TR document is divided into four sections and several appendixes. Section 1 includes the purpose and scope of the TR as applied to the ARMS application. Section 2 details the test design specifications. Section 3 defines the test case specifications for each test design specification. Section 4 provides a summary of results of the tests performed.

This document conforms to the formatting and structure standards as set forth by the IEEE Standards Board and Dr. F.T. Sheldon. Appendix A lists the specific requirements, as defined in the SRS, in an RTM. Appendix B contains the individual test-case results.

## **2 Test-Design Specification**

The test-design specifications will include the specific details of the test approach for a software feature or combination of features and identifying the associated tests. There are several test-designs that will encompass all of the items found in the ARMS SRS RTM.

### **2.1 Purpose**

This section will breakdown the specific test-designs and provide a detailed assignment of the SRS RTM items to be tested by each test-design in particular. Any requirements will generally be addressed in the order presented in the SRS RTM. All requirements to be tested will be referred to by their respective SRS RTM identifiers.

### **2.2 Outline: TDS01**

This test-design specification will be organized in the following manner: TDS Identifier, features to be tested, approach refinements, test-case identification list, and feature pass/fail criteria.

#### **2.2.1 Test-Design Specification Identifier**

The first general test-design, referred to as TDS01, will center on a testing combination of inspecting the source code and an application trace file to satisfy the specified features and requirements.

### **2.2.2 Features to be Tested**

The SRS RTM identifiers will be used as the reference for the features to be tested in this test design specification.

- 2.2.2.1** The final report screen will have the statistical summary results verified against RTM items: A01.21, A01.22, A01.23, A01.24, A01.25, B11.02, B11.03, B11.04, B11.05, B11.06, B11.07, B11.08, B11.09, and B11.10.
- 2.2.2.2** The manner in which the ARMS application handles the end-of-simulation event will be addressed against RTM item: B03.00.
- 2.2.2.3** The disbursement of arriving aircraft is to be tested against RTM items: B04.00, B04.02, B04.03, B04.05, and B04.06.
- 2.2.2.4** The engine inspection/repair cyclic process is to be tested against RTM items: B05.02, B05.02.1, B05.02.2, B05.02.3, B05.02.4, B05.02.5, B05.02.6, and B05.02.7.
- 2.2.2.5** The depletion of aircraft via departure events is to be tested against RTM items: B05.02.8, B05.03, B05.04, and B08.00.
- 2.2.2.6** The basic queuing options, as implemented in the code will be verified against RTM items: B06.01, B06.02, B06.02.1, B06.02.2, B06.03, B06.03.1, and B06.03.2.
- 2.2.2.7** The master configuration table input for the required aircraft types will be verified against RTM item: B07.00.

### **2.2.3 Approach Refinements**

The two primary methods used in verification are an examination/inspection of the source code and review/analysis of an application trace file (*ATF*).

An ATF will be produced as a result of an execution of the source code as detailed by the design/coding group. The produced ATF will be parsed into a spreadsheet format, as specified in Appendix C. The spreadsheet will have a variety of analysis applied to it as well as direct observations of the sequence of events surrounding the individual aircraft as the progress through the simulation.

The design/coding group will provide the source code listing. It will be contemporaneous with the version of the software under test. A detailed code walkthrough will not be conducted as an attribute of the test activity. The code will simply be reviewed to broadly determine

whether it supports the various specified requirements.

#### **2.2.4 Test Identification**

The test-cases that will support this test-design are as follows:

Test Case One [TCS01 section 3.2]: The ARMS application is allowed to run unhindered using queuing scenario one with acceptable user inputs.

Test Case Two [TCS02 section 3.3]: The ARMS application is allowed to run unhindered using queuing scenario two with acceptable user inputs.

Test Case Three [TCS03 section 3.4]: The ARMS application is allowed to run unhindered using queuing scenario three with acceptable user inputs.

#### **2.2.5 Feature Pass/Fail Criteria**

Each RTM item will be independently evaluated and given an individual pass/fail rating. If a test case only passes a partial number of RTM items and fails others, the test case will only have the necessary sub-sections retested to evaluate those areas that previously failed. Although, some discretion will be allowed to conduct retests on already passed RTM items if the Test Engineer so deems it necessary at the time.

**2.2.5.1** If an independently derived test summary statistical result is within 5% of the application generated test result, it will be considered to have passed.

**2.2.5.2** If the application ends when the end-of-simulation is processed it will have passed.

**2.2.5.3** If an ATF analysis demonstrates the aircraft arrivals are within 10% of the master configuration parameters specifications, it will have passed.

**2.2.5.4** If an ATF analysis demonstrates that all processed aircraft in the simulation have their arrival events handled as specified in the SRS, it will have passed.

**2.2.5.5** If an ATF analysis demonstrates that all aircraft engine inspection/repair events are handled in the correct order, it will have passed.

**2.2.5.6** If an ATF analysis demonstrates that all aircraft inspection times, repair decisions, and repair times are within 10% of the expected master configuration parameters and SRS description, it will have passed.

**2.2.5.7** A source code will be sufficient to satisfy RTM items that pertain to the general queuing option requirements as detailed in this test-design. At the Test Engineers discretion, a RTM item will have passed if it has properly met the technical components of the SRS. The Test Engineer must provide sufficient explanation or reasons why the requirements are meeting the criteria or not.

## **2.3 Outline: TDS02**

This test-design specification will be organized in the following manner: TDS Identifier, features to be tested, approach refinements, test identification list, and feature pass/fail criteria.

### **2.3.1 Test-Design Specification Identifier**

The second general test-design, referred to as TDS02, will focus on a visual inspection and observations of the application as it operates and reacts to the end user directed insertions and deletions that are enacted in a sensible/reasonable manner.

### **2.3.2 Features to be Tested**

The SRS RTM identifiers will be used as the reference for the features to be tested in this test design specification.

**2.3.2.1** The GUI *Input Screen* will be observed for its proper response and dispensing of the various initial user entries, this will specifically regard RTM items A01.00, A01.01, A01.02, A01.03, A01.05, A01.06, A01.07, A01.08, A01.09, and B06.04.

**2.3.2.2** The GUI *Progress Screen* will be observed for its operation and the manner in which it responds to the user intervention via the various insertion/deletion options, this will specifically regard RTM items A01.10, A01.11, A01.12, A01.13, A01.14, A01.15, A01.15.1, A01.15.2, A01.15.3, A01.15.4, A01.15.5, A01.15.6, and A01.16.

**2.3.2.3** The GUI *Report Screen* will be observed for how it responds to the user action buttons and whether or not a full set of statistical values are reported as described in the SRS, this will specifically regard RTM items A01.20, A01.21, A01.22, A01.23, A01.24, A01.25, and A01.26.

**2.3.2.4** The GUI end user entries and various initial settings will be observed as to how it is passed from one screen and how it is transferred to the next, this will specifically regard RTM items A02.00, A02.01, B01.05, B01.06, and B01.07.

**2.3.2.5** The aircraft arrivals, departures, and processing via the service stations and respective queues will be observed in of the GUI *Progress Screen* and the manner in which aircraft are moved from one list to another; this will specifically regard RTM items B04.03, B04.04, B04.05, B04.06, B04.07, and B05.04.

- 2.3.2.6** The representation of the queuing disciplines and service station designations will be observed in the *GUI Progress Screen* and the proper operation of moving designated aircraft from one list to another, regarding RTM items B06.00, B06.01.1, B06.01.2, B06.01.3, B06.02.1, B06.02.2, B06.02.3, B06.02.4, B06.02.5, B06.02.6, B06.03.1, B06.03.2, B06.03.3, B06.03.4, B06.03.5, B06.03.6, and B06.03.7.
- 2.3.2.7** The *GUI Progress Screen* and *GUI Report Screen* are to be observed to show that only the seven required aircrafts are available and reported upon, regarding RTM item B07.00.
- 2.3.2.8** The Test Engineer using reasonable entries and actions, regarding RTM items B09.00, B09.01, B09.02, B09.03, B10.00, B10.01, B10.02, and B10.03, will exercise the *GUI Progress Screen* insertion and deletion actions.
- 2.3.2.9** The Test Engineer will observe the values reported in the *GUI Report Screen* for all SRS defined statistical items, regarding RTM items B11.00, B11.01, B11.03, B11.04, B11.05, B11.06, B11.07, B11.08, B11.09, and B11.10.
- 2.3.2.10** Non-functional RTM items are to be noted by the Test Engineer during the testing procedure, specifically regarding RTM items C01.00 and C02.00.

### **2.3.3 Approach Refinements**

The primary method to be used for this test-design will be the visual observation of the application while it is in operation. The application will be operated using reasonable entries and sound actions with respect to the manipulation of the aircraft within the simulation process. It will be backed up where necessary with a supplementary review/analysis of the application trace file that will be created during the tests.

The Test Engineer will exercise all insertion and deletion operations of the progress screen. This will primarily be done using a very slow relative time scale (*simulation time to real-time*). The Test Engineer may even elect to minimize the total aircraft being processed by the system to make individual observations easier to note.

The design/coding group will provide the application to be tested. They will specify which modules, components, functions, and specific SRS RTM items are ready for testing in the form of a Test Transmittal Report via e-mail. The Test Engineer will create a response via a Test Incidence Report via e-mail to the Design/Coding group; it will detail the results of the testing process.

### 2.3.4 Test Identification

The test-cases that will support this test-design are as follows:

Test Case Four [TCS04 section 3.5]: The ARMS application is run using queuing scenario one with acceptable user inputs and the reasonable exercising of the various user operations.

Test Case Five [TCS05 section 3.6]: The ARMS application is run using queuing scenario two with acceptable user inputs and the reasonable exercising of the various user operations.

Test Case Six [TCS06 section 3.7]: The ARMS application is run using queuing scenario three with acceptable user inputs and the reasonable exercising of the various user operations.

### 2.3.5 Feature Pass/Fail Criteria

Each SRS RTM item will be independently evaluated and given an individual pass/fail rating. If a test case only passes a partial number of SRS RTM items and fails others, the test case will only have the necessary sub-sections redone to evaluate those areas that previously failed. Although, some discretion will be allowed to redo tests on already passed SRS RTM items if the Test Engineer so deems it necessary at the time.

**2.3.5.1** The GUI *Input Screen* will have its RTM items passed, if they simply exist and provide the basic functionality as described in the SRS.

**2.3.5.2** The GUI *Progress Screen* will have its RTM items passed, if they simply exist and provide the basic functionality as described in the SRS.

**2.3.5.3** The GUI *Report Screen* will have its RTM items passed, if they simply exist and provide the basic functionality as described in the SRS.

**2.3.5.4** The Test Engineer will pass the GUI end user entries and various initial settings based upon direct observation of the GUI.

**2.3.5.5** The aircraft arrivals, departures, and processing via the service stations and respective queues will have passed upon direct observation of the GUI *Progress Screen* and the manner in which it represents these events.

**2.3.5.6** The representation of the queuing disciplines and service station designations will have passed upon direct observation of the GUI *Progress Screen* and the proper operation of the aircraft from one list to another.

**2.3.5.7** If the *GUI Progress Screen* and *GUI Report Screen* show that the seven required aircrafts are available and reported upon, it will have passed.

**2.3.5.8** The Test Engineer will exercise the *GUI Progress Screen* insertion and deletion actions and if the actions are properly represented then they will have passed.

**2.3.5.9** If the Test Engineer observes values in the *GUI Report Screen* for all SRS defined statistical items, they will have passed.

**2.3.5.10** Non-functional RTM items are noted by the Test Engineer and will have passed if they are simply met as described in the SRS.

## **2.4 Outline: TDS03**

This test-design specification will be organized in the following manner: TDS Identifier, features to be tested, approach refinements, test identification list, and feature pass/fail criteria.

### **2.4.1 Test-Design Specification Identifier**

The third general test-design, referred to as TDS03, will focus on a visual inspection and observations of the application as it operates and reacts to the end user directed insertions and deletions that are enacted in an imprudent/reckless manner.

### **2.4.2 Features to be Tested**

The SRS RTM identifiers will be used as the reference for the features to be tested in this test design specification.

**2.4.2.1** The *GUI Input Screen* will be observed for its response and dispensing of the various initial improper user entries, this will specifically regard RTM items A01.02, A01.03, A01.05, A01.08, A01.09, and B06.04.

**2.4.2.2** The *GUI Progress Screen* will be observed for its response to the unreasonable user actions via a variety of insertions and deletions, this will specifically pertain to RTM items A01.15, A01.15.2, A01.15.3, A01.15.4, and A01.15.5.

**2.4.2.3** The GUI end user entries and various unreasonable initial settings will be observed as to how they are passed from one screen and to the next, this will specifically regard RTM items B01.05, B01.06, and B01.07.

**2.4.2.4** The aircraft arrivals, departures, and processing via the service stations and respective queues will be manipulated by the Test Engineer to purposefully attempt to make aircraft movements observed in the *GUI Progress Screen* respond incorrectly; this will

specifically regard RTM items B04.03, B04.07, B05.04, B06.01.1, B06.01.2, and B06.01.3.

**2.4.2.5** The representation of the queuing disciplines and service station designations will be observed in the GUI *Progress Screen* particularly in the applications response to reprehensible user actions, regarding RTM items B06.02.1, B06.02.2, B06.02.3, B06.02.4, B06.02.5, B06.02.6, B06.03.1, B06.03.2, B06.03.3, B06.03.4, B06.03.5, B06.03.6, and B06.03.7.

**2.4.2.6** The GUI *Progress Screen* insertion and deletion actions will be exercised by the Test Engineer using actions such as repetitiously trying to “break” the GUI by clicking rapidly or irrationally on the interface buttons, this is regarding RTM items B09.00, B09.01, B09.02, B09.03, B10.00, B10.01, B10.02, and B10.03.

### **2.4.3 Approach Refinements**

The primary method to be used for this test-design will be the visual observation of the application while it is in operation. The application will be operated using unreasonable entries and purposeful chaotic actions with respect to the manipulation of the aircraft within the simulation process. It will be backed up where necessary with a supplementary review/analysis of the application trace file that will be created during the tests.

The Test Engineer will exercise all insertion and deletion operations of the progress screen that may cause the simulation to run incorrectly. This will primarily be done using a very slow relative time scale (*simulation time to real-time*). The Test Engineer may even elect to minimize the total aircraft being processed by the system to make individual observations easier to note.

The design/coding group will provide the application to be tested. They will specify which modules, components, functions, and specific SRS RTM items are ready for testing in the form of a Test Transmittal Report via e-mail. The Test Engineer will create a response via a Test Incidence Report via e-mail to the Design/Coding group; it will detail the results of the testing process.

### **2.4.4 Test Identification**

The test-cases that will support this test-design are as follows:

Test Case Seven [TCS07 section 3.8]: The ARMS application is run using queuing scenario one with unacceptable user inputs and the unreasonable exercising of the various user operations.

Test Case Eight [TCS08 section 3.9]: The ARMS application is run using queuing scenario two with unacceptable user inputs and the unreasonable exercising of the various user operations.

Test Case Nine [TCS09 section 3.10]: The ARMS application is run using queuing scenario three with unacceptable user inputs and the unreasonable exercising of the various user operations.

#### **2.4.5 Feature Pass/Fail Criteria**

Each SRS RTM item will be independently evaluated and given an individual pass/fail rating. If a test case only passes a partial number of SRS RTM items and fails others, the test case will only have the necessary sub-sections redone to evaluate those areas that previously failed. Although, some discretion will be allowed to redo tests on already passed SRS RTM items if the Test Engineer so deems it necessary at the time.

**2.4.5.1** If the GUI *Input Screen* does not crash upon improper end user entries or it does so in a documented/predicted manner, then it will have passed.

**2.4.5.2** If the GUI *Progress Screen* does not allow the user to insert aircraft improperly and to not delete nonexistent aircraft or it crashes in a documented/predicted manner, then it will have passed.

**2.4.5.3** If the GUI *Input Screen* does not allow the end user to make unreasonable initial entries or it crashes in a documented or predicted manner, then it will have passed.

**2.4.5.4** If the GUI *Progress Screen* does not allow the user to manipulate aircraft arrivals, departures, and processing via the service stations and respective queues incorrectly or it crashes in a documented/predicted manner, then it will have passed.

**2.4.5.5** If the GUI *Progress Screen* does not allow the user to break the service station or queuing rules with improper insertions or deletions or it crashes in a documented/predicted manner, then it will have passed.

**2.4.5.6** If the GUI *Progress Screen* does not allow the user to have multiple insertions or deletes occurring simultaneously or it crashes in a documented/predicted manner, then it will have passed.

#### **2.5 Outline: TDS04**

This test-design specification will be organized in the following manner: TDS Identifier, features to be tested, approach refinements, test identification list, and feature pass/fail criteria.

### **2.5.1 Test-Design Specification Identifier**

The fourth general test-design, referred to as TDS04, will focus on the continuity between the ARMS User's Manual and the ARMS application itself. The primary test technique will be for the Test Engineer to make literal assumptions based on the Users Manual when testing the application.

### **2.5.2 Features to be Tested**

The Test Engineer will be focusing on those items listed in the SRS RTM that are specifically geared to the end user interactions with the system. Those items that are functionally observable to provide the user with visual queues will be referred to in the users manual. These are the specific RTM items to be addressed by this test-design.

**2.5.2.1** The GUI *Input Screen* that should be reviewed between the application and Users Manual are the RTM items A01.00, A01.01, A01.02, A01.03, A01.05, A01.06, A01.07, A01.08, and A01.09.

**2.5.2.2** The GUI *Progress Screen* is relatively complicated with respect to its explanation in the Users Manual and so the application should not have any discrepancies with regards to RTM items A01.10, A01.15.1, A01.15.2, A01.15.3, A01.15.4, A01.15.5, A01.15.6, A01.15, and A01.16.

**2.5.2.3** The GUI *Report Screen* has relatively few options for the User Manual that may cause significant discrepancies; the Test Engineer will make note with particular regard to RTM items A01.20, and A01.26.

**2.5.2.4** The manner in which the Users Manual and application have consistencies with regards to descriptions of the queuing system options and related random seed values, specifically RTM items B01.07, B06.00, B06.01, B06.02, and B06.03.

**2.5.2.5** The GUI *Progress Screen* explanations in the Users Manual with respect to the users ability to affect the application aircraft lists via the insertion and deletion functions is particularly complex. The Test Engineer must take special scrutiny to assure that the application and the Users Manual are synchronized and correct, in particular with regards to RTM items B09.00, B09.01, B09.02, B09.03, B10.00, B10.01, B10.02, and B10.03.

### **2.5.3 Approach Refinements**

The primary method to be used for this test-design will be literal interpretation of the users manual and the operation of the application. The Test Engineer will go through the users manual

and the application for discrepancies. A discrepancy will be defined as any item that is in one artifact but not the other.

#### **2.5.4 Test Identification**

The test-cases that will support this test-design are as follows:

Test Case Ten [TCS10 section 3.11]: The ARMS application is run using queuing scenario one with acceptable user inputs and the reasonable exercising of the various user operations.

Test Case Eleven [TCS11 section 3.12]: The ARMS application is run using queuing scenario two with acceptable user inputs and the reasonable exercising of the various user operations.

Test Case Twelve [TCS12 section 3.13]: The ARMS application is run using queuing scenario three with acceptable user inputs and the reasonable exercising of the various user operations.

#### **2.5.5 Feature Pass/Fail Criteria**

If the users manual defines an item that is not in the application, a Test Incident Report will be e-mailed concerning this issue. If an item is in the application but not the Users Manual, then the application will be given a failure for that SRS RTM item.

If there is a discrepancy between the users manual and the software it will be noted by the Test Engineer and reported via Test Incident Report via e-mail. It will be the responsibility of the Design/Coding group to resolve User Manual and software application discrepancies.

### **2.6 Outline: TDS05**

This test-design specification will be organized in the following manner: TDS Identifier, features to be tested, approach refinements, test identification list, and feature pass/fail criteria.

#### **2.6.1 Test-Design Specification Identifier**

The fifth general test-design, referred to as TDS05, will use as its primary methodology a direct review of the source code listing and will be supplemented with interviews of the Design/Coding group. The reason behind needing annotations directly concerning the source code listing for the

testing team is to clarify specific SRS RTM items. Additionally, a few elements to be tested are strictly the Design groups' responsibility.

### **2.6.2 Features to be Tested**

The Test Engineer will receive a source code listing from the Design/Coding group contemporaneously with the version of the ARMS application currently under test.

- 2.6.2.1** The internal operation of the application with respect to the extensibility support, and clock events (*SimLib functions*) will be reviewed, specifically the RTM items A01.30, A03.00, and A03.01.
- 2.6.2.2** The application will initialize some of the internal data structures at the beginning, specifically the RTM items B01.00, B01.01, B01.02, B01.03, B01.04, and B01.07.
- 2.6.2.3** The application will utilize a time advancement technique specifically regarding RTM item B02.00.
- 2.6.2.4** The processing of aircraft arrival, departure and engine inspection/repair times will be made using SimLib functions, specifically regarding RTM items B04.01, B04.02, B04.04, B05.01, B05.02.2, and B05.02.3.
- 2.6.2.5** The application will support a specific number of queuing disciplines and aircraft parameters and the master configuration table, specifically regarding RTM items B06.00, B07.00, and B08.00.
- 2.6.2.6** The internal "streams" data structure will be utilized by the application for particular ARMS functions, specifically regarding RTM items D01.01, D01.02, D01.03, and D01.04.
- 2.6.2.7** The JAVA requirements will be reviewed with the Design/Coding group, specifically regarding the RTM items D02.00 and D02.01.
- 2.6.2.8** The application will be reviewed for evidence of usage of the SimLib functions as described in RTM items D03.00, D04.00, D05.00, and D06.00.
- 2.6.2.9** What the Design/Coding group used as sources, development system, and compilation process will be provided as affidavits referring specifically to the RTM items D06.01, D06.02, and D07.00.

### **2.6.3 Approach Refinements**

The primary method to be used for this test-design will be the direct visual review of the source code listing in conference with the Design/Coding group. Additionally, the Design/Coding group will supply evidence, via the Design Notebook to fulfill some of the non-functional requirements.

### **2.6.4 Test Identification**

The test-cases that will support this test-design are as follows:

Test Case Thirteen [TCS13 section 3.14]: The ARMS application source code listing will be reviewed concurrently with the release of software official versions as needed.

### **2.6.5 Feature Pass/Fail Criteria**

If the Design/Coding group is able to supply sufficient documentary evidence to demonstrate proof of a RTM item has met its descriptive criteria, then it will have passed.

If a criteria of this test-design has been met on an earlier version of the software, on subsequent versions, a re-iteration of confidence from the Design/Coding group that the previously attained “passing” test cases have not been altered. If the Design/Coding group feels they have sufficiently altered the software, then a reassessment of the source code listing may be required.

## **3 Test-Case Specification**

The test-case specifications will define the various inputs, predicted results, and a set of execution actions to be evaluated and deployed while conducting these processes.

### **3.1 Purpose**

This section is to refine the numerous test cases that were identified in the test-design specification section. Any requirements will generally be addressed in the order presented in the SRS RTM. All requirements to be tested will be referred to by their respective SRS RTM identifiers.

## 3.2 Outline: TCS01

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

### 3.2.1 Test-Case Specification Identifier

This test-case, referred to as TCS01, will focus on the inspection of the source code and the application trace file (*ATF*) to satisfy the specified requirements. The ARMS application is to be allowed to run unhindered using queuing scenario one with acceptable user inputs.

### 3.2.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.2.2.1** The numerical statistics in the final report summary will be verified to reflect:

- Average delay in queue for each aircraft type. .... [A01.21, B11.03]
- Overall average delay in queue for all aircraft types. .... [A01.22, B11.04]
- Time-average number of aircraft in the wait queue(s). .... [A01.23, B11.05]
- Time-average number of aircraft down for each aircraft type. .... [A01.24, B11.06]
- Total average daily downtime cost for all aircraft added together. .. [A01.25, B11.07]

**3.2.2.2** The costing statistics in the final report summary will be verified to reflect:

- Aircraft prematurely removed from the maintenance process. .... [B11.08]
- Manual insertion of aircraft into the maintenance process. .... [B11.09]
- Aircraft still in the maintenance process at the end of the simulation. .... [B11.10]

**3.2.2.3** Calculate the statistics using SimLib functions. .... [B11.02]

**3.2.2.4** Simulation end at “365 day” termination event processing. .... [B03.00]

**3.2.2.5** Aircraft arrival disbursement and service stations will be verified to reflect:

- Aircraft arrival event support. .... [B04.00]
- Times are determined probabilistically with SimLib. .... [B04.02]
- Events are handled in increasing time order. .... [B04.03]

- Arriving aircraft are put into empty (*not full*) service stations. .... [B04.05]
- Service stations are full (*not empty*) when they contain one aircraft. .... [B04.06]

**3.2.2.6** Aircraft departure dispensing and service stations will be verified to reflect:

- Aircraft departure event support. .... [B05.00]
- Scheduled using SimLib functions. .... [B05.01]
- Calculated are based on inspection/repair all aircraft engines. .... [B05.02, B05.03]
- The next queued aircraft shall occupy an empty service station. .... [B05.04]

**3.2.2.7** Each aircraft engine inspection/repair cycle will be verified to reflect:

- Has undergone an initial inspection. .... [B05.02.1]
- Inspection times determined probabilistically using SimLib functions. .... [B05.02.2]
- Inspection success/failure is determined using SimLib functions. .... [B05.02.3]
- A repair activity will be done, if it is needed. .... [B05.02.4]
- Initial repair times are determined probabilistically SimLib functions. .... [B05.02.5]
- Until an engine passes inspection, the inspection/repair cycle repeats. .... [B05.02.6]
- The inspection/repair process may be repeated for all engines. .... [B05.02.7]
- Departure time is the total of all engine inspection/repair cycles. .... [B05.02.8]

**3.2.2.8** The Master Configuration Table is employed for modeling aircraft arrival/departure events and is verified as being aircraft type specific. .... [B07.00, B08.00]

**3.2.2.9** This test-case focuses on the first queuing discipline option, and the manner in which it utilizes a single queue using a “first-in first-out” algorithm. .... [B06.01].

### **3.2.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = single FIFO queue
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will have the internal application trace file (*ATF*) generation flag set.

The ARMS application will be allowed to run unhindered once it has been started to its natural conclusion.

### **3.2.4 Output Specifications**

The primary output generated by the ARMS run is the application trace file (*ATF*). The ATF will be parsed and analyzed to meet the acceptance criteria as defined in **TDS01** [TR section 2.2.5].

Additionally, the ATF will be inspected to verify the user inputs are reflected in the application and that the Master Configuration Table was read in properly.

The ARMS application source code will also be inspected as needed to clarify the specific usage of any SimLib functions.

### **3.2.5 Environmental Needs**

This section will define the specific necessary hardware, software, and any unique facilities or personnel required to perform or execute the test procedures.

#### **3.2.5.1 Hardware.**

The testing will be done using a computer system that meets or exceeds all items referred to in the ARMS SRS sections 2.2.1, 2.2.2, 2.2.4, and 2.2.5.

#### **3.2.5.2 Software.**

The testing will be done using a computer system that meets or exceeds all items referred to in the ARMS SRS sections 2.2.1, 2.2.3, 2.2.4, and 2.2.7.

#### **3.2.5.3 Other.**

The computer systems utilized for testing will primarily be located at the Washington State University, Tri-Cities campus, West Building, room W151. The personnel to perform the testing shall be comprised of those students who comprise the Team 9: Goldrush group.

### **3.2.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### 3.2.7 Intercase Dependencies

The TCS01 results are not contingent upon the results from any other ARMS test-case specifications. The TCS01 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.3 Outline: TCS02

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.3.1 Test-Case Specification Identifier

This test-case, referred to as TCS02, will focus on the inspection of the source code and the application trace file (*ATF*) to satisfy the specified requirements. The ARMS application is to be allowed to run unhindered using queuing scenario two with acceptable user inputs.

#### 3.3.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.3.2.1** The numerical statistics in the final report summary will be verified to reflect:

- Average delay in queue for each aircraft type. .... [A01.21, B11.03]
- Overall average delay in queue for all aircraft types. .... [A01.22, B11.04]
- Time-average number of aircraft in the wait queue(s). .... [A01.23, B11.05]
- Time-average number of aircraft down for each aircraft type. .... [A01.24, B11.06]
- Total average daily downtime cost for all aircraft added together. .. [A01.25, B11.07]

**3.3.2.2** The costing statistics in the final report summary will be verified to reflect:

- Aircraft prematurely removed from the maintenance process. .... [B11.08]
- Manual insertion of aircraft into the maintenance process. .... [B11.09]
- Aircraft still in the maintenance process at the end of the simulation. .... [B11.10]

- 3.3.2.3 Calculate the statistics using SimLib functions..... [B11.02]
- 3.3.2.4 Simulation end at “365 day” termination event processing..... [B03.00]
- 3.3.2.5 Aircraft arrival disbursement and service stations will be verified to reflect:
- Aircraft arrival event support. .... [B04.00]
  - Times are determined probabilistically with SimLib. .... [B04.02]
  - Events are handled in increasing time order..... [B04.03]
  - Arriving aircraft are put into empty (*not full*) service stations. .... [B04.05]
  - Service stations are full (*not empty*) when they contain one aircraft..... [B04.06]
- 3.3.2.6 Aircraft departure dispensing and service stations will be verified to reflect:
- Aircraft departure event support..... [B05.00]
  - Scheduled using SimLib functions..... [B05.01]
  - Calculated are based on inspection/repair all aircraft engines.....[B05.02, B05.03]
  - The next queued aircraft shall occupy an empty service station. .... [B05.04]
- 3.3.2.7 Each aircraft engine inspection/repair cycle will be verified to reflect:
- Has undergone an initial inspection. .... [B05.02.1]
  - Inspection times determined probabilistically using SimLib functions. .... [B05.02.2]
  - Inspection success/failure is determined using SimLib functions..... [B05.02.3]
  - A repair activity will be done, if it is needed..... [B05.02.4]
  - Initial repair times are determined probabilistically SimLib functions. .... [B05.02.5]
  - Until an engine passes inspection, the inspection/repair cycle repeats. .... [B05.02.6]
  - The inspection/repair process may be repeated for all engines. .... [B05.02.7]
  - Departure time is the total of all engine inspection/repair cycles..... [B05.02.8]
- 3.3.2.8 The Master Configuration Table is employed for modeling aircraft arrival/departure events and is verified as being aircraft type specific. ....[B07.00, B08.00]
- 3.3.2.9 This test-case focuses on the second queuing discipline option (*regular & wide-body aircraft await in separate queues, with all stations servicing all aircraft*), and the manner in which it accomplishes the following:
- Dual independent queues..... [B06.02].
  - First queue tracks only the regular-body aircraft..... [B06.02.1]
  - Second queue tracks only the wide-body aircraft..... [B06.02.2]

### 3.3.3 Input Specifications

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Dual FIFO queue (*Non-preemptive wide-body*)
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will have the internal application trace file (ATF) generation flag set.

The ARMS application will be allowed to run unhindered once it has been started to its natural conclusion.

### **3.3.4 Output Specifications**

The primary output generated by the ARMS run is the application trace file (ATF). The ATF will be parsed and analyzed to meet the acceptance criteria as defined in TDS01 [TR section 2.2.5].

Additionally, the ATF will be inspected to verify the user inputs are reflected in the application and that the Master Configuration Table was read in properly.

The ARMS application source code will also be inspected as needed to clarify the specific usage of any SimLib functions.

### **3.3.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

### **3.3.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### **3.3.7 Intercase Dependencies**

The TCS02 results are not contingent upon the results from any other ARMS test-case specifications. The TCS02 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.4 Outline: TCS03

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.4.1 Test-Case Specification Identifier

This test-case, referred to as TCS03, will focus on the inspection of the source code and the application trace file (*ATF*) to satisfy the specified requirements. The ARMS application is to be allowed to run unhindered using queuing scenario two with acceptable user inputs.

#### 3.4.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.4.2.1** The numerical statistics in the final report summary will be verified to reflect:

- Average delay in queue for each aircraft type. .... [A01.21, B11.03]
- Overall average delay in queue for all aircraft types. .... [A01.22, B11.04]
- Time-average number of aircraft in the wait queue(s). .... [A01.23, B11.05]
- Time-average number of aircraft down for each aircraft type. .... [A01.24, B11.06]
- Total average daily downtime cost for all aircraft added together. .. [A01.25, B11.07]

**3.4.2.2** The costing statistics in the final report summary will be verified to reflect:

- Aircraft prematurely removed from the maintenance process. .... [B11.08]
- Manual insertion of aircraft into the maintenance process. .... [B11.09]
- Aircraft still in the maintenance process at the end of the simulation. .... [B11.10]

**3.4.2.3** Calculate the statistics using SimLib functions. .... [B11.02]

**3.4.2.4** Simulation end at “365 day” termination event processing..... [B03.00]

**3.4.2.5** Aircraft arrival disbursement and service stations will be verified to reflect:

- Aircraft arrival event support. .... [B04.00]
- Times are determined probabilistically with SimLib. .... [B04.02]
- Events are handled in increasing time order..... [B04.03]

- Arriving aircraft are put into empty (*not full*) service stations. .... [B04.05]
- Service stations are full (*not empty*) when they contain one aircraft. .... [B04.06]

**3.4.2.6** Aircraft departure dispensing and service stations will be verified to reflect:

- Aircraft departure event support. .... [B05.00]
- Scheduled using SimLib functions. .... [B05.01]
- Calculated are based on inspection/repair all aircraft engines. .... [B05.02, B05.03]
- The next queued aircraft shall occupy an empty service station. .... [B05.04]

**3.4.2.7** Each aircraft engine inspection/repair cycle will be verified to reflect:

- Has undergone an initial inspection. .... [B05.02.1]
- Inspection times determined probabilistically using SimLib functions. .... [B05.02.2]
- Inspection success/failure is determined using SimLib functions. .... [B05.02.3]
- A repair activity will be done, if it is needed. .... [B05.02.4]
- Initial repair times are determined probabilistically SimLib functions. .... [B05.02.5]
- Until an engine passes inspection, the inspection/repair cycle repeats. .... [B05.02.6]
- The inspection/repair process may be repeated for all engines. .... [B05.02.7]
- Departure time is the total of all engine inspection/repair cycles. .... [B05.02.8]

**3.4.2.8** The Master Configuration Table is employed for modeling aircraft arrival/departure events and is verified as being aircraft type specific. .... [B07.00, B08.00]

**3.4.2.9** This test-case focuses on the third queuing discipline option (*regular & wide-body aircraft await in separate queues, with stations servicing particular aircraft body types*), and the manner in which it accomplishes the following:

- Dual independent queues. .... [B06.03].
- First queue tracks only the regular-body aircraft. .... [B06.03.1]
- Second queue tracks only the wide-body aircraft. .... [B06.03.2]

### **3.4.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Separate FIFO queues (*separate service stations*)
- Number of Regular Service Stations = 6
- Number of Wide-Body Service Stations = 4
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will have the internal application trace file (ATF) generation flag set.

The ARMS application will be allowed to run unhindered once it has been started to its natural conclusion.

### **3.4.4 Output Specifications**

The primary output generated by the ARMS run is the application trace file (*ATF*). The ATF will be parsed and analyzed to meet the acceptance criteria as defined in **TDS01** [[TR section 2.2.5](#)].

Additionally, the ATF will be inspected to verify the user inputs are reflected in the application and that the Master Configuration Table was read in properly.

The ARMS application source code will also be inspected as needed to clarify the specific usage of any SimLib functions.

### **3.4.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section [3.2.5](#).

### **3.4.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### **3.4.7 Intercase Dependencies**

The [TCS03](#) results are not contingent upon the results from any other ARMS test-case specifications. The [TCS03](#) results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

## **3.5 Outline: TCS04**

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

### 3.5.1 Test-Case Specification Identifier

This test-case, referred to as TCS04, will focus on running the ARMS application with the Test Engineer making visual observations concerning its manner of operation using the first queuing discipline.

### 3.5.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.5.2.1** The ARMS application graphical user interface will be verified for the following:

- It provides a graphical user interface..... [A01.00]
- It provides an input screen..... [A01.01]
- It provides a progress screen ..... [A01.10]
- It provides a final report screen. .... [A01.20]

**3.5.2.2** The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02, B06.00]
- The number of service bays may be specified..... [A01.03]
- A random number seed may be specified..... [A01.05]
- The application may be quit from..... [A01.06]
- The simulation may be started..... [A01.07]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- All user data entries are made using the keyboard and mouse. .... [A02.00]
- The end user may specify the number of service stations. .... [B06.01.3]
- The maximum service stations number entry is ten (10)..... [B06.04]

**3.5.2.3** The progress screen, while it is “running” will be verified for the following:

- The current event list is displayed. .... [A01.11]
- The current simulation time is displayed..... [A01.12]
- The queue lists the aircraft waiting for repair..... [A01.13]
- The service station occupancy status is displayed. .... [A01.14]
- The simulation may be paused. .... [A01.15]
- The simulation is initially in the paused state. .... [A01.16]

**3.5.2.4** The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]

- Deletion of the first record of a list with a specific attribute. .... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list. .... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue. .... [A01.15.5, B10.02]
- Simulation resumption. .... [A01.15.6]
- Edit the end simulation event. .... [B09.03]
- Simulation termination, resulting in a final report. .... [A01.15.1]

**3.5.2.5** The progress screen will be observed for the following actions:

- The event execution speed was initialized with the user entry. .... [B01.05]
- The pseudo-random number seed was initialized with the user entry. .... [B01.06]
- A default random seed was provided if the user didn't enter one. .... [B01.07]
- Arrival events are handled in order based on increasing time. .... [B04.03]
- Simulation clock is updated after processing to an arrival event time. .... [B04.04]
- Arriving aircraft are put into empty (*not full*) service stations. .... [B04.05]
- Service stations are full (*not empty*) when they contain one aircraft. .... [B04.06]
- When service stations are all full, arriving aircraft are put in a queue. .... [B04.07]
- When aircraft depart from a service station, the next queued aircraft shall occupy the empty service station, beginning the repair process. .... [B05.04]
- All aircraft types are mixed into the queue and treated equally. .... [B06.01.1]
- All service stations are capable of inspecting/repairing all aircraft. .... [B06.01.2]

**3.5.2.6** The final report screen will be verified for the following:

- Application termination by user. .... [A01.26]
- Statistics generated at the conclusion of the simulation. .... [B11.00]
- All results shall be displayed on screen. .... [A02.01, B11.01]
- Average delay in queue for each aircraft type. .... [A01.21, B11.03]
- Overall average delay in queue for all aircraft types. .... [A01.22, B11.04]
- Time-average number of aircraft in the wait queue(s). .... [A01.23, B11.05]
- Time-average number of aircraft down for each aircraft type. .... [A01.24, B11.06]
- Total average daily downtime cost for all aircraft. .... [A01.25, B11.07]
- Costs from aircraft prematurely removed from the maintenance process. .... [B11.08]
- Costs from aircraft manually inserted into the maintenance process. .... [B11.09]
- Costs from aircraft in the maintenance process at simulation end. .... [B11.10]

**3.5.2.7** The ARMS application will be verified for the following:

- It supports seven distinct aircraft types. .... [B07.00]
- Is able to complete a full simulation in sixty (60) minutes. .... [C01.00]
- Is able to operate on a computer meeting the minimum specifications. .... [C02.00]

### 3.5.3 Input Specifications

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = single FIFO queue
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS02 section 2.3.5. These observations will cover the input, progress, and report screens.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (*Event, Queue, Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS02 section 2.3.5.

### 3.5.4 Output Specifications

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeated, if possible.

### **3.5.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

### **3.5.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### **3.5.7 Intercase Dependencies**

The TCS04 results are not contingent upon the results from any other ARMS test-case specifications. The TCS04 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

## **3.6 Outline: TCS05**

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

### **3.6.1 Test-Case Specification Identifier**

This test-case, referred to as TCS05, will focus on running the ARMS application with the Test Engineer making visual observations about its manner of operation using the second queuing discipline.

### **3.6.2 Test Items**

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

- 3.6.2.1** The ARMS application graphical user interface will be verified for the following:
- It provides a graphical user interface..... [A01.00]
  - It provides an input screen..... [A01.01]
  - It provides a progress screen ..... [A01.10]
  - It provides a final report screen. .... [A01.20]
- 3.6.2.2** The input screen will be verified for the following:
- Only one of three queuing disciplines is selectable..... [A01.02, B06.00]
  - The number of service bays may be specified. .... [A01.03]
  - A random number seed may be specified..... [A01.05]
  - The application may be quit from..... [A01.06]
  - The simulation may be started..... [A01.07]
  - The initial event execution speed may be set. .... [A01.08, A01.09]
  - All user data entries are made using the keyboard and mouse. .... [A02.00]
  - The end user may specify the number of service stations. .... [B06.02.6]
  - The maximum service stations number entry is ten (10)..... [B06.04]
- 3.6.2.3** The progress screen, while it is “running” will be verified for the following:
- The current event list is displayed. .... [A01.11]
  - The current simulation time is displayed..... [A01.12]
  - The queue list the proper aircraft waiting for repair..... [A01.13]
  - The service station occupancy status is displayed. .... [A01.14]
  - The simulation may be paused. .... [A01.15]
  - The simulation is initially in the paused state. .... [A01.16]
- 3.6.2.4** The progress screen, while it is “paused” verify that the user may do the following:
- Insertion of events into the event list..... [A01.15.2, B09.00]
  - Deletion of the first record of a list with a specific attribute. .... [B10.00]
  - Insert arrival events. .... [B09.01]
  - Edit departure events. .... [B09.02]
  - Removal/deletion of events from the event list..... [A01.15.3, B10.01]
  - Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
  - Aircraft removal/deletion from the wait queue(s). .... [A01.15.5, B10.02]
  - Simulation resumption. .... [A01.15.6]
  - Edit the end simulation event. .... [B09.03]
  - Simulation termination, resulting in a final report. .... [A01.15.1]
- 3.6.2.5** The progress screen will be observed for the following actions:
- The event execution speed was initialized with the user entry. .... [B01.05]
  - The pseudo-random number seed was initialized with the user entry..... [B01.06]
  - A default random seed was provided if the user didn’t enter one. .... [B01.07]
  - Arrival events are handled in order based on increasing time..... [B04.03]
  - Simulation clock is updated after processing to an arrival event time. .... [B04.04]

- Arriving aircraft are put into empty (*not full*) service stations. .... [B04.05]
- Service stations are full (*not empty*) when they contain one aircraft. .... [B04.06]
- When service stations are all full, arriving aircraft are put in a queue. .... [B04.07]
- When aircraft depart from a service station, the next queued aircraft shall occupy the empty service station, beginning the repair process. .... [B05.04]
- All stations are capable of inspecting/repairing all aircraft. .... [B06.02.5]
- The first queue tracks the regular-body aircraft progress. .... [B06.02.1]
- The second queue tracks the wide-body aircraft progress. .... [B06.02.2]
- Each queue has its own “first-in first-out” prioritization. .... [B06.02.3]
- Queue Two (*wide-body aircraft*) has non-preemptive priority over Queue One (*regular body aircraft*). .... [B06.02.4]

**3.6.2.6** The final report screen will be verified for the following:

- Application termination by user. .... [A01.26]
- Statistics generated at the conclusion of the simulation ..... [B11.00]
- All results shall be displayed on screen. .... [A02.01, B11.01]
- Average delay in queue for each aircraft type. .... [A01.21, B11.03]
- Overall average delay in queue for all aircraft types. .... [A01.22, B11.04]
- Time-average number of aircraft in the wait queue(s). .... [A01.23, B11.05]
- Time-average number of aircraft down for each aircraft type. .... [A01.24, B11.06]
- Total average daily downtime cost for all aircraft. .... [A01.25, B11.07]
- Costs from aircraft prematurely removed from the maintenance process. .... [B11.08]
- Costs from aircraft manually inserted into the maintenance process. .... [B11.09]
- Costs from aircraft in the maintenance process at simulation end. .... [B11.10]

**3.6.2.7** The ARMS application will be verified for the following:

- It supports seven distinct aircraft types. .... [B07.00]
- Is able to complete a full simulation in sixty (*60*) minutes. .... [C01.00]
- Is able to operate on a computer meeting the minimum specifications. .... [C02.00]

### **3.6.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Dual FIFO queue (*Non-preemptive wide-body*)
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (*7*) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible

(e.g.: items that do not require insertions/modifications/deletions to the various lists). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS02 section 2.3.5. These observations will cover the input, progress, and report screens. Special care will be done to observe the particular aircraft types and the manner in which they are moved between the event list, queues, and service stations. This is to assure that the “non-preemptive wide-body” rule is being accomplished correctly.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (*Event*, *Queue*, *Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS02 section 2.3.5.

### **3.6.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeatable, if it is so desired.

### **3.6.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

### **3.6.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the

Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### 3.6.7 Intercase Dependencies

The TCS05 results are not contingent upon the results from any other ARMS test-case specifications. The TCS05 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

## 3.7 Outline: TCS06

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

### 3.7.1 Test-Case Specification Identifier

This test-case, referred to as TCS06, will focus on running the ARMS application with the Test Engineer making visual observations about its manner of operation using the third queuing discipline.

### 3.7.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.7.2.1** The ARMS application graphical user interface will be verified for the following:

- It provides a graphical user interface..... [A01.00]
- It provides an input screen..... [A01.01]
- It provides a progress screen ..... [A01.10]
- It provides a final report screen. .... [A01.20]

**3.7.2.2** The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02, B06.00]
- The number of service bays may be specified..... [A01.03]

- A random number seed may be specified..... [A01.05]
- The application may be quit from..... [A01.06]
- The simulation may be started..... [A01.07]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- All user data entries are made using the keyboard and mouse. .... [A02.00]
- The end user may specify the number of service stations dedicated to wide-body and regular-body types of aircraft. .... [B06.03.6]
- The user must have at least one service station for each body type. .... [B06.03.7]
- The maximum service stations number entry is ten (10)..... [B06.04]

**3.7.2.3** The progress screen, while it is “running” will be verified for the following:

- The current event list is displayed. .... [A01.11]
- The current simulation time is displayed..... [A01.12]
- The queue list the proper aircraft waiting for repair..... [A01.13]
- The service station occupancy status is displayed. .... [A01.14]
- The simulation may be paused. .... [A01.15]
- The simulation is initially in the paused state. .... [A01.16]

**3.7.2.4** The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]
- Deletion of the first record of a list with a specific attribute..... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list..... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue(s). .... [A01.15.5, B10.02]
- Simulation resumption. .... [A01.15.6]
- Edit the end simulation event. .... [B09.03]
- Simulation termination, resulting in a final report. .... [A01.15.1]

**3.7.2.5** The progress screen will be observed for the following actions:

- The event execution speed was initialized with the user entry. .... [B01.05]
- The pseudo-random number seed was initialized with the user entry..... [B01.06]
- A default random seed was provided if the user didn’t enter one. .... [B01.07]
- Arrival events are handled in order based on increasing time..... [B04.03]
- Simulation clock is updated after processing to an arrival event time. .... [B04.04]
- Arriving aircraft are put into empty (*not full*) service stations. .... [B04.05]
- Service stations are full (*not empty*) when they contain one aircraft..... [B04.06]
- When service stations are all full, arriving aircraft are put in a queue. .... [B04.07]
- When aircraft depart from a service station, the next proper queued aircraft shall occupy the empty service station, beginning the repair process..... [B05.04]
- The first queue tracks the regular-body aircraft progress. .... [B06.03.1]
- The second queue tracks the wide-body aircraft progress..... [B06.03.2]

- Each queue has its own “first-in first-out” prioritization. .... [B06.03.3]
- One subset of service stations only supports wide-body aircraft..... [B06.03.4]
- One subset of service stations only supports regular-body aircraft. .... [B06.03.5]

**3.7.2.6** The final report screen will be verified for the following:

- Application termination by user. .... [A01.26]
- Statistics generated at the conclusion of the simulation ..... [B11.00]
- All results shall be displayed on screen..... [A02.01, B11.01]
- Average delay in queue for each aircraft type. .... [A01.21, B11.03]
- Overall average delay in queue for all aircraft types. .... [A01.22, B11.04]
- Time-average number of aircraft in the wait queue(s). .... [A01.23, B11.05]
- Time-average number of aircraft down for each aircraft type..... [A01.24, B11.06]
- Total average daily downtime cost for all aircraft..... [A01.25, B11.07]
- Costs from aircraft prematurely removed from the maintenance process. .... [B11.08]
- Costs from aircraft manually inserted into the maintenance process. .... [B11.09]
- Costs from aircraft in the maintenance process at simulation end. .... [B11.10]

**3.7.2.7** The ARMS application will be verified for the following:

- It supports seven distinct aircraft types. .... [B07.00]
- Is able to complete a full simulation in sixty (60) minutes. .... [C01.00]
- Is able to operate on a computer meeting the minimum specifications..... [C02.00]

### **3.7.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Separate FIFO queues (*separate service stations*)
- Number of Regular Service Stations = 6
- Number of Wide-Body Service Stations = 4
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS02 section 2.3.5. These observations will cover the input, progress, and report screens. Special care will be done to observe the particular aircraft types and the manner in which they are moved between the event list, queues, and service stations. This is to

assure that the “separate, but equal aircraft body type” rules are being accomplished correctly.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (*Event*, *Queue*, *Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS02 section 2.3.5.

### **3.7.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeatable, if it is so desired.

### **3.7.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

### **3.7.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### 3.7.7 Intercase Dependencies

The TCS06 results are not contingent upon the results from any other ARMS test-case specifications. The TCS06 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

## 3.8 Outline: TCS07

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

### 3.8.1 Test-Case Specification Identifier

This test-case, referred to as TCS07, will focus on running the ARMS application with the Test Engineer making visual observations concerning its manner of operation using the first queuing discipline with unreasonable actions or inputs.

### 3.8.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.8.2.1** The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02]
- The number of service bays may be specified..... [A01.03]
- Cannot input of more than 10 (*total*) service stations. .... [A01.04, B06.04]
- A random number seed may be specified..... [A01.05]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- The end user may specify the number of service stations. .... [B06.01.3]

**3.8.2.2** The progress screen, while it is “running” will be verified for the following:

- The simulation may be paused. .... [A01.15]

**3.8.2.3** The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]
- Deletion of the first record of a list with a specific attribute. .... [B10.00]

- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list. .... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue. .... [A01.15.5, B10.02]
- Edit the end simulation event. .... [B09.03]

**3.8.2.4** The progress screen will be observed for the following actions:

- The event execution speed was initialized with the user entry. .... [B01.05]
- The pseudo-random number seed was initialized with the user entry. .... [B01.06]
- A default random seed was provided if the user didn't enter one. .... [B01.07]
- Arrival events are handled in order based on increasing time. .... [B04.03]
- When service stations are all full, arriving aircraft are put in a queue. .... [B04.07]
- When aircraft depart from a service station, the next queued aircraft shall occupy the empty service station, beginning the repair process. .... [B05.04]
- All aircraft types are mixed into the queue and treated equally. .... [B06.01.1]
- All service stations are capable of inspecting/repairing all aircraft. .... [B06.01.2]

**3.8.2.5** The ARMS application will be verified for the following:

- Is able to complete a full simulation in sixty (60) minutes. .... [C01.00]

### **3.8.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = single FIFO queue
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS03 section 2.4.5. These observations will cover the input, progress, and report screens.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the

following per list (*Event, Queue, Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS03 section 2.4.5.

### **3.8.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeated, if possible.

### **3.8.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

### **3.8.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### **3.8.7 Intercase Dependencies**

The TCS07 results are not contingent upon the results from any other ARMS test-case specifications. The TCS07 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.9 Outline: TCS08

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.9.1 Test-Case Specification Identifier

This test-case, referred to as TCS08, will focus on running the ARMS application with the Test Engineer making visual observations about its manner of operation using the second queuing discipline with unreasonable actions or inputs.

#### 3.9.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.9.2.1** The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02]
- The number of service bays may be specified..... [A01.03]
- Cannot input of more than 10 (*total*) service stations. .... [A01.04, B06.04]
- A random number seed may be specified..... [A01.05]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- The end user may specify the number of service stations. .... [B06.02.6]

**3.9.2.2** The progress screen, while it is “running” will be verified for the following:

- The simulation may be paused. .... [A01.15]

**3.9.2.3** The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]
- Deletion of the first record of a list with a specific attribute..... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list..... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue(s). .... [A01.15.5, B10.02]
- Edit the end simulation event. .... [B09.03]

**3.9.2.4** The progress screen will be observed for the following actions:

- The event execution speed was initialized with the user entry. .... [B01.05]
- The pseudo-random number seed was initialized with the user entry..... [B01.06]
- A default random seed was provided if the user didn't enter one. .... [B01.07]
- Arrival events are handled in order based on increasing time..... [B04.03]
- When service stations are all full, arriving aircraft are put in a queue. .... [B04.07]
- When aircraft depart from a service station, the next queued aircraft shall occupy the empty service station, beginning the repair process. .... [B05.04]
- All stations are capable of inspecting/repairing all aircraft. .... [B06.02.5]
- The first queue tracks the regular-body aircraft progress. .... [B06.02.1]
- The second queue tracks the wide-body aircraft progress. .... [B06.02.2]
- Each queue has its own “first-in first-out” prioritization. .... [B06.02.3]
- Queue Two (*wide-body aircraft*) has non-preemptive priority over Queue One (*regular body aircraft*). .... [B06.02.4]

**3.9.2.5** The ARMS application will be verified for the following:

- Is able to complete a full simulation in sixty (60) minutes. .... [C01.00]

### **3.9.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Dual FIFO queue (*Non-preemptive wide-body*)
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS03 section 2.4.5. These observations will cover the input, progress, and report screens. Special care will be done to observe the particular aircraft types and the manner in which they are moved between the event list, queues, and service stations. This is to assure that the “non-preemptive wide-body” rule is being accomplished correctly.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the

following per list (*Event, Queue, Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS03 section 2.4.5.

### **3.9.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeated, if possible.

### **3.9.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

### **3.9.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

### **3.9.7 Intercase Dependencies**

The TCS08 results are not contingent upon the results from any other ARMS test-case specifications. The TCS08 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.10 Outline: TCS09

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.10.1 Test-Case Specification Identifier

This test-case, referred to as TCS09, will focus on running the ARMS application with the Test Engineer making visual observations about its manner of operation using the third queuing discipline with unreasonable actions or inputs.

#### 3.10.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

##### 3.10.2.1 The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02]
- The number of service bays may be specified..... [A01.03]
- Cannot input of more than 10 (*total*) service stations. .... [A01.04, B06.04]
- A random number seed may be specified..... [A01.05]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- The end user may specify the number of service stations dedicated to wide-body and regular-body types of aircraft. .... [B06.03.6]
- The user must have at least one service station for each body type. .... [B06.03.7]

##### 3.10.2.2 The progress screen, while it is “running” will be verified for the following:

- The simulation may be paused. .... [A01.15]

##### 3.10.2.3 The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]
- Deletion of the first record of a list with a specific attribute..... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list..... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]

- Aircraft removal/deletion from the wait queue(s). ..... [A01.15.5, B10.02]
- Edit the end simulation event. .... [B09.03]

**3.10.2.4** The progress screen will be observed for the following actions:

- The event execution speed was initialized with the user entry. .... [B01.05]
- The pseudo-random number seed was initialized with the user entry..... [B01.06]
- A default random seed was provided if the user didn't enter one. .... [B01.07]
- Arrival events are handled in order based on increasing time..... [B04.03]
- When service stations are all full, arriving aircraft are put in a queue. .... [B04.07]
- When aircraft depart from a service station, the next proper queued aircraft shall occupy the empty service station, beginning the repair process..... [B05.04]
- The first queue tracks the regular-body aircraft progress. .... [B06.03.1]
- The second queue tracks the wide-body aircraft progress..... [B06.03.2]
- Each queue has its own “first-in first-out” prioritization. .... [B06.03.3]
- One subset of service stations only supports wide-body aircraft..... [B06.03.4]
- One subset of service stations only supports regular-body aircraft. .... [B06.03.5]

**3.10.2.5** The ARMS application will be verified for the following:

- Is able to complete a full simulation in sixty (60) minutes. .... [C01.00]

### **3.10.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Separate FIFO queues (*separate service stations*)
- Number of Regular Service Stations = 6
- Number of Wide-Body Service Stations = 4
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (e.g.: items that do not require insertions/modifications/deletions to the various lists). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS04 section 2.4.5. These observations will cover the input, progress, and report screens. Special care will be done to observe the particular aircraft types and the manner in which they are moved between the event list, queues, and service stations. This is to assure that the “separate, but equal aircraft body type” rules are being accomplished correctly.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (*Event, Queue, Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS03 section 2.4.5.

#### **3.10.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeated, if possible.

#### **3.10.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

#### **3.10.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

#### **3.10.7 Intercase Dependencies**

The TCS09 results are not contingent upon the results from any other ARMS test-case specifications. The TCS09 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.11 Outline: TCS10

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.11.1 Test-Case Specification Identifier

This test-case, referred to as TCS10, will focus on running the ARMS application with the Test Engineer making visual observations concerning its manner of operation using the first queuing discipline, specifically checking for continuity with the ARMS User Manual (*UM*).

#### 3.11.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

**3.11.2.1** The ARMS application graphical user interface will be verified for the following:

- It provides a graphical user interface..... [A01.00]
- It provides an input screen..... [A01.01]
- It provides a progress screen ..... [A01.10]
- It provides a final report screen. .... [A01.20]

**3.11.2.2** The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02, B06.00]
- The number of service bays may be specified. .... [A01.03]
- A random number seed may be specified..... [A01.05]
- The application may be quit from..... [A01.06]
- The simulation may be started..... [A01.07]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- All user data entries are made using the keyboard and mouse. .... [A02.00]

**3.11.2.3** The progress screen, while it is “running” will be verified for the following:

- The simulation may be paused. .... [A01.15]

**3.11.2.4** The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]

- Deletion of the first record of a list with a specific attribute. .... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list. .... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue. .... [A01.15.5, B10.02]
- Simulation resumption. .... [A01.15.6]
- Edit the end simulation event. .... [B09.03]
- Simulation termination, resulting in a final report. .... [A01.15.1]

**3.11.2.5** The progress screen will be observed for the following actions:

- A default random seed was provided if the user didn't enter one. .... [B01.07]

**3.11.2.6** The final report screen will be verified for the following:

- Application termination by user. .... [A01.26]

### **3.11.3 Input Specifications**

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = single FIFO queue
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS04 section 2.5.5. These observations will cover the input, progress, and report screens.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (*Event, Queue, Service Stations*): insertion/adding, deletion/removal, and/or modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either

having “passed” or “failed” as per the criteria outlined in TDS04 section 2.5.5.

#### **3.11.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeated, if possible.

#### **3.11.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

#### **3.11.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

#### **3.11.7 Intercase Dependencies**

The TCS10 results are not contingent upon the results from any other ARMS test-case specifications. The TCS10 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.12 Outline: TCS11

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.12.1 Test-Case Specification Identifier

This test-case, referred to as TCS11, will focus on running the ARMS application with the Test Engineer making visual observations concerning its manner of operation using the second queuing discipline, specifically checking for continuity with the ARMS User Manual (*UM*).

#### 3.12.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

##### 3.12.2.1 The ARMS application graphical user interface will be verified for the following:

- It provides a graphical user interface..... [A01.00]
- It provides an input screen..... [A01.01]
- It provides a progress screen ..... [A01.10]
- It provides a final report screen. .... [A01.20]

##### 3.12.2.2 The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02, B06.00]
- The number of service bays may be specified. .... [A01.03]
- A random number seed may be specified..... [A01.05]
- The application may be quit from..... [A01.06]
- The simulation may be started..... [A01.07]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- All user data entries are made using the keyboard and mouse. .... [A02.00]

##### 3.12.2.3 The progress screen, while it is “running” will be verified for the following:

- The simulation may be paused. .... [A01.15]

##### 3.12.2.4 The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]

- Deletion of the first record of a list with a specific attribute. .... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list. .... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue(s). .... [A01.15.5, B10.02]
- Simulation resumption. .... [A01.15.6]
- Edit the end simulation event. .... [B09.03]
- Simulation termination, resulting in a final report. .... [A01.15.1]

3.12.2.5 The progress screen will be observed for the following actions:

- A default random seed was provided if the user didn't enter one. .... [B01.07]

3.12.2.6 The final report screen will be verified for the following:

- Application termination by user. .... [A01.26]

### 3.12.3 Input Specifications

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Dual FIFO queue (*Non-preemptive wide-body*)
- Number of Service Stations = 10
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS04 section 2.5.5. These observations will cover the input, progress, and report screens. Special care will be done to observe the particular aircraft types and the manner in which they are moved between the event list, queues, and service stations. This is to assure that the “non-preemptive wide-body” rule is being accomplished correctly.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (Event, Queue, Service Stations): insertion/adding, deletion/removal, and/or

modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS04 section 2.5.5.

#### **3.12.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeatable, if it is so desired.

#### **3.12.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

#### **3.12.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

#### **3.12.7 Intercase Dependencies**

The TCS11 results are not contingent upon the results from any other ARMS test-case specifications. The TCS11 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.13 Outline: TCS12

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.13.1 Test-Case Specification Identifier

This test-case, referred to as TCS12, will focus on running the ARMS application with the Test Engineer making visual observations concerning its manner of operation using the third queuing discipline, specifically checking for continuity with the ARMS User Manual (*UM*).

#### 3.13.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

3.13.2.1 The ARMS application graphical user interface will be verified for the following:

- It provides a graphical user interface..... [A01.00]
- It provides an input screen..... [A01.01]
- It provides a progress screen ..... [A01.10]
- It provides a final report screen. .... [A01.20]

3.13.2.2 The input screen will be verified for the following:

- Only one of three queuing disciplines is selectable..... [A01.02, B06.00]
- The number of service bays may be specified. .... [A01.03]
- A random number seed may be specified..... [A01.05]
- The application may be quit from..... [A01.06]
- The simulation may be started..... [A01.07]
- The initial event execution speed may be set. .... [A01.08, A01.09]
- All user data entries are made using the keyboard and mouse. .... [A02.00]

3.13.2.3 The progress screen, while it is “running” will be verified for the following:

- The simulation may be paused. .... [A01.15]

3.13.2.4 The progress screen, while it is “paused” verify that the user may do the following:

- Insertion of events into the event list..... [A01.15.2, B09.00]

- Deletion of the first record of a list with a specific attribute. .... [B10.00]
- Insert arrival events. .... [B09.01]
- Edit departure events. .... [B09.02]
- Removal/deletion of events from the event list. .... [A01.15.3, B10.01]
- Aircraft removal/deletion from service stations. .... [A01.15.4, B10.03]
- Aircraft removal/deletion from the wait queue(s). .... [A01.15.5, B10.02]
- Simulation resumption. .... [A01.15.6]
- Edit the end simulation event. .... [B09.03]
- Simulation termination, resulting in a final report. .... [A01.15.1]

3.13.2.5 The progress screen will be observed for the following actions:

- A default random seed was provided if the user didn't enter one. .... [B01.07]

3.13.2.6 The final report screen will be verified for the following:

- Application termination by user. .... [A01.26]

### 3.13.3 Input Specifications

The primary user inputs/option selections in the initial application screen are to be:

- Discipline = Separate FIFO queues (*separate service stations*)
- Number of Regular Service Stations = 6
- Number of Wide-Body Service Stations = 4
- Random Seed = 123456

The Master Configuration Table of seven (7) aircraft type specific parameters is an external file read in by the ARMS application at startup [B08.00].

The ARMS application will not have the internal application trace file (*ATF*) flag set.

The Test Engineer will initially make visual observations as to the overall operation of the ARMS application. The Test Engineer will verify as many “non-interfering” items possible (*e.g.: items that do not require insertions/modifications/deletions to the various lists*). The Test Engineer will make notations of all observations as to whether or not they met the specified criteria as outlined in TDS04 section 2.5.5. These observations will cover the input, progress, and report screens. Special care will be done to observe the particular aircraft types and the manner in which they are moved between the event list, queues, and service stations. This is to assure that the “separate, but equal aircraft body type” rules are being accomplished correctly.

The Test Engineer once having completed the above test observations will perform reasonable and rational actions on the various lists. This will include at least one action of the following per list (*Event, Queue, Service Stations*): insertion/adding, deletion/removal, and/or

modification. The Test Engineer will make detailed notes as to the results of the actions taken by the ARMS application from the user actions. The Test Engineer will mark each test as either having “passed” or “failed” as per the criteria outlined in TDS04 section 2.5.5.

#### **3.13.4 Output Specifications**

The primary output generated by the ARMS run is the GUI screens. A majority of the testing activities will focus on the progress screen alone. The input screen will be limited to the simple user entry choice verification. The report screen will primarily consist of whether or not all the desired outputs are provided.

The Test Engineer will detail in their Test Log and resulting Test Incident report what was done to test the ARMS application and the resulting screen activities that were observed. A sufficient level of detail must be provided by the Test Engineer to allow the test-case to be repeated, if possible.

#### **3.13.5 Environmental Needs**

This section will be considered identical for all intents and purposes to TR section 3.2.5.

#### **3.13.6 Special Procedural Requirements**

The computer system to be used under the test procedures will be configured and designated by the Team 9: Goldrush Design/Coding sub-team members. The individual(s) conducting this test-case are to record which machine they use as a part of the testing process.

Because not all SRS RTM requirements may be available simultaneously from the Design/Coding team, at the discretion of the testing staff, some portions of the test-cases may be repeated. If the repeated tests show no deviations from previous test-case executions, then the new observations for the repeated portions need not be recorded.

#### **3.13.7 Intercase Dependencies**

The TCS12 results are not contingent upon the results from any other ARMS test-case specifications. The TCS12 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

### 3.14 Outline: TCS13

This test-case specification will be organized in the following manner: TCS Identifier, test items, input specifications, output specifications, environmental needs, special procedural requirements, and intercase dependencies.

#### 3.14.1 Test-Case Specification Identifier

This test-case, referred to as TCS13, will focus on a direct inspection of the source code to satisfy the SRS RTM items. The ARMS application is not run as a function of this test case. Some SRS RTM items will be fulfilled via supplementary interviews with the Design/Coding team.

#### 3.14.2 Test Items

The SRS RTM identifiers will be used as the reference for the features to be tested in this test-case specification. The individual SRS RTM identifiers are only summarized here, the testing staff should carefully review the detailed descriptions available in the SRS. The SRS should be kept directly on-hand during the testing process to be directly referred to during the execution of the test-case.

##### 3.14.2.1 Verification regarding the utilization of SimLib functions for the following:

- The advancement of the simulation clock. .... [A03.01]
- All probabilistic time estimates. .... [A03.00]
- Probabilistic exponential distribution of arrival event times. .... [B04.02]
- Probabilistic uniform distributions of initial inspection times. .... [B05.02.2]
- Scheduling aircraft arrival events. .... [B04.01]
- Scheduling aircraft departure events. .... [B05.01]
- Aircraft engine inspection success or failure. .... [B05.02.3]

##### 3.14.2.2 Verification regarding the use of the SimLib “stream” structure:

- Source of the random numbers used in probability calculations. .... [D01.00]
- Numbers 01 - 07 shall be used for aircraft inter-arrival times. .... [D01.01]
- Numbers 08 - 14 shall be used for engine inspection time. .... [D01.02]
- Numbers 15 - 21 shall be used for determination of needed repairs. .... [D01.03]
- Numbers 22 - 28 shall be used for engine repair times. .... [D01.04]

##### 3.14.2.3 Other SimLib non-functional requirements to be verified are:

- The GUI can be extended to support more queues and service stations. .... [A01.30]
- List-structures are used for the event list, queues, and service stations. .... [D03.00]
- Its functions will be used wherever applicable. .... [D06.00]

- SimLib comes from the web-site for the Law & Kelton book. .... [D06.01]
- The C-library shall be compiled under a Linux OS..... [D06.02]

**3.14.2.4** Initial conditions to be verified upon the startup of the ARMS application are:

- An initialization procedure is executed. .... [B01.00]
- The queues are all set to empty. .... [B01.01]
- The service stations are all set to empty. .... [B01.02]
- The simulation clock is set to zero. .... [B01.03]
- The 365 days worth of aircraft arrival times are calculated. .... [B01.04]

**3.14.2.5** Operational/functional conditions for the ARMS application are:

- The program will supply a default random seed. .... [B01.07]
- A discreet next-event time advancement model is used..... [B02.00]
- The simulation clock is updated after processing an arrival event..... [B04.04]
- Three different queuing disciplines are supported. .... [B06.00]
- Seven distinct aircraft types are supported. .... [B07.00]
- The Master Configuration Table parameters are utilized..... [B08.00]
- Event list insertions use the "middle-pointer" algorithm..... [D04.00]
- The transfer array holds the deleted records..... [D05.00]

**3.14.2.6** Non-functional conditions for the ARMS application are:

- It is developed with the Java 1.2 (*Sun Sys*) language. .... [D02.00, D02.01]
- It functions on a PC using Linux OS..... [D07.00]

**3.14.3 Input Specifications**

There are no input specifications, as this test case does not involve running the ARMS application. It is purely a moderate code review and discussion of concepts, sources and functions used concerning SimLib and the background activities behind the ARMS application.

**3.14.4 Output Specifications**

The primary output generated by the ARMS run is the application source code listing. This is further supplemented by interviews/discussions with the Design/Coding group. This test case's items will have passed if the acceptance criteria as defined in **TDS05** [TR section 2.6.5] are met.

**3.14.5 Environmental Needs**

There are no environmental needs required by this test-case.

### **3.14.6 Special Procedural Requirements**

The Test Engineer who interviews the Design/Coding group must write a narrative of the discussion. It must provide sufficient detail to demonstrate the concepts were conveyed and understood by all parties involved, and who those individuals were.

### **3.14.7 Intercase Dependencies**

The TCS13 results are not contingent upon the results from any other ARMS test-case specifications. The TCS13 results (*pass/fail*) are not considered to be antecedent to any other test-case specifications.

## **4 Test-Summary Report**

This section is organized in two primary subsections, the purpose and the outline.

### **4.1 Purpose**

This section is to provide a singular area where an overall evaluation of the designated test-case results may be found.

### **4.2 Outline: TSR01**

This section will breakdown the specifics of the overall evaluation of the test-case results. It will do so by a TSR identifier, summary of evaluations performed, variances from the SRS, comprehensive assessment, summary of results, evaluation of the entire product status, and a summary of the testing activities encompassed by this report, and concluding with the approvals.

#### **4.2.1 Test-Summary-Report Identifier**

This test summary report will be identified as **TSR01**.

#### **4.2.2 Summary**

There were five test-design specifications supported by thirteen test-case specifications. At the time of this draft Test Report, none of the test-cases have been concluded. Several of the test-cases have been started, but the Testing group is primarily on hold while the Coding group produces a release product software version.

### 4.2.3 Variances

The Testing group has noted some deviations from the SRS to the Coding group with the preliminary software version released as a demonstration. Outside of that, there is nothing of significance to report at the time of this draft Test Report.

### 4.2.4 Comprehensive Assessment

At the time of the issuance of this draft Test Report the Testing group has not started the full battery of planned test-cases as we are awaiting the release of a testable software version for the application.

### 4.2.5 Summary of Results

There are no results of significance to be announced at this time as there has been no real progress of any substance at the time of drafting of this Test Report.

### 4.2.6 Evaluation

The primary item for evaluation here is the relevant schedule of events as put forth by the ARMS Project Plan. The Testing phase of the software is currently behind schedule.

### 4.2.7 Summary of Activities

So far the primary accomplishment of the Testing group has been drafting this Test Report. This is in addition to their regular duties and contributions to the Team 9: Goldrush deliverables that have been produced to date.

### 4.2.8 Approvals

|                                   |             |                             |             |
|-----------------------------------|-------------|-----------------------------|-------------|
| _____                             | _____       | _____                       | _____       |
| <i>Project Manager</i>            | <i>Date</i> | <i>ESEP Representative</i>  | <i>Date</i> |
| _____                             | _____       | _____                       | _____       |
| <i>Lead Requirements Engineer</i> | <i>Date</i> | <i>Lead Design Engineer</i> | <i>Date</i> |
| _____                             | _____       | _____                       | _____       |
| <i>Lead Test Engineer</i>         | <i>Date</i> | <i>Lead Programmer</i>      | <i>Date</i> |

## APPENDIX A: The SRS Requirements Traceability Matrix

Table 1: SRS RTM

| Req. ID<br>System Level | Req. ID<br>Sub-system Level | DFD<br>Identifier(s) | Module<br>Name(s)    | Verification<br>Method | Tested |
|-------------------------|-----------------------------|----------------------|----------------------|------------------------|--------|
| A01.00                  |                             | GUI                  |                      | D                      |        |
|                         | A01.01                      | GUI                  |                      | D                      |        |
|                         | A01.02                      | GUI                  |                      | D                      |        |
|                         | A01.03                      | GUI                  |                      | D                      |        |
|                         | A01.04                      | GUI, 1.2             | Evaluate # of Bays   | D                      |        |
|                         | A01.05                      | GUI, 1.3             | Eval. Random Seed    | D                      |        |
|                         | A01.06                      | GUI                  |                      | D                      |        |
|                         | A01.07                      | GUI                  |                      | D                      |        |
|                         | A01.08                      | GUI                  |                      | D                      |        |
|                         | A01.09                      | GUI                  |                      | A, D                   |        |
|                         | A01.10                      | GUI                  |                      | D                      |        |
|                         | A01.11                      | GUI, 3.3             | Update GUI           | I, D                   |        |
|                         | A01.12                      | GUI, 3.3             | Update GUI           | I, D                   |        |
|                         | A01.13                      | GUI, 3.3             | Update GUI           | D                      |        |
|                         | A01.14                      | GUI, 3.3             | Update GUI           | I, D                   |        |
|                         | A01.15                      | GUI                  |                      | D                      |        |
|                         | A01.15.1                    | GUI, 3.2.4           | Process Endsim       | D                      |        |
|                         | A01.15.2                    | GUI, 3.3             | Update GUI           | D                      |        |
|                         | A01.15.3                    | GUI, 3.3             | Update GUI           | D                      |        |
|                         | A01.15.4                    | GUI, 3.3             | Update GUI           | D                      |        |
|                         | A01.15.5                    | GUI, 3.3             | Update GUI           | D                      |        |
|                         | A01.15.6                    | GUI                  |                      | D                      |        |
|                         | A01.16                      | GUI, 2.3             | Init. Pause Flag     | D                      |        |
|                         | A01.20                      | GUI, 3.2.4           | Process Endsim       | D                      |        |
|                         | A01.21                      | GUI, 3.2.4           | Process Endsim       | I, D                   |        |
|                         | A01.22                      | GUI, 3.2.4           | Process Endsim       | I, D                   |        |
|                         | A01.23                      | GUI, 3.2.4           | Process Endsim       | I, D                   |        |
|                         | A01.24                      | GUI, 3.2.4           | Process Endsim       | I, D                   |        |
|                         | A01.25                      | GUI, 3.2.4           | Process Endsim       | I, D                   |        |
|                         | A01.26                      | GUI                  |                      | D                      |        |
|                         | A01.30                      | GUI                  |                      | I                      |        |
| A02.00                  |                             | GUI                  |                      | D                      |        |
|                         | A02.01                      | GUI                  |                      | D                      |        |
| A03.00                  |                             | 2.1                  | Initialize SimLib    | I                      |        |
|                         | A03.01                      | 3.2                  | Proc. Intern. Events | I                      |        |

**Key: A = Analysis, D = Demonstration, I = Inspection, K = Analogy**

## Appendix A: SRS Requirements Traceability Matrix *(continued)*

|        |          |              |                   |         |  |
|--------|----------|--------------|-------------------|---------|--|
| B01.00 |          | 2.           | Init. Simulation  | A, I    |  |
|        | B01.01   | 2.1          | Initialize SimLib | I       |  |
|        | B01.02   | 2.1          | Initialize SimLib | I       |  |
|        | B01.03   | 2.1          | Initialize SimLib | I       |  |
|        | B01.04   | 2.1          | Initialize SimLib | A, I    |  |
|        | B01.05   | 2.2          | Init. Delay Time  | A, D    |  |
|        | B01.06   | 2.1          | Initialize SimLib | A, D    |  |
|        | B01.07   | 2.1          | Initialize SimLib | A, D    |  |
| B02.00 |          | 2.           | Init. Simulation  | I       |  |
| B03.00 |          | 3.2.4        | Process Ends      | A, D    |  |
| B04.00 |          | 3.2.2        | Process Arrival   | I       |  |
|        | B04.01   | 3.2.2        | Process Arrival   | I       |  |
|        | B04.02   | 3.2.2        | Process Arrival   | I       |  |
|        | B04.03   | 3.2.2        | Process Arrival   | A, I, D |  |
|        | B04.04   | 3.2.2        | Process Arrival   | A, D    |  |
|        | B04.05   | 3.2.2        | Process Arrival   | D       |  |
|        | B04.06   | 3.2.2        | Process Arrival   | D       |  |
|        | B04.07   | 3.2.2        | Process Arrival   | D       |  |
| B05.00 |          | 3.2.3        | Process Departure | I       |  |
|        | B05.01   | 3.2.3        | Process Departure | I       |  |
|        | B05.02   | 3.2.3        | Process Departure | A, D    |  |
|        | B05.02.1 | 3.2.3        | Process Departure | I, D    |  |
|        | B05.02.2 | 3.2.3        | Process Departure | I       |  |
|        | B05.02.3 | 3.2.3        | Process Departure | I       |  |
|        | B05.02.4 | 3.2.3        | Process Departure | I, D    |  |
|        | B05.02.5 | 3.2.3        | Process Departure | I       |  |
|        | B05.02.6 | 3.2.3        | Process Departure | I, D    |  |
|        | B05.02.7 | 3.2.3        | Process Departure | I, D    |  |
|        | B05.02.8 | 3.2.3        | Process Departure | A, D    |  |
|        | B05.03   | 3.2.3        | Process Departure | A, D    |  |
|        | B05.04   | 3.2.3        | Process Departure | I, D    |  |
| B06.00 |          | 3.2.2, 3.3.3 | Arrival/Departure | I, D    |  |
|        | B06.01   | 3.2.2        | Process Arrival   | D       |  |
|        | B06.01.1 | 3.2.2        | Process Arrival   | I, D    |  |
|        | B06.01.2 | 3.2.3        | Process Departure | I, D    |  |
|        | B06.01.3 | 1.           | Evaluate Input    | D       |  |
|        | B06.02   | 3.2.2        | Process Arrival   | I, D    |  |
|        | B06.02.1 | 3.2.2        | Process Arrival   | I, D    |  |
|        | B06.02.2 | 3.2.2        | Process Arrival   | I, D    |  |
|        | B06.02.3 | 3.2.2        | Process Arrival   | I, D    |  |
|        | B06.02.4 | 3.2.3        | Process Departure | I, D    |  |
|        | B06.02.5 | 3.2.3        | Process Departure | I, D    |  |
|        | B06.02.6 | 1.           | Evaluate Input    | D       |  |

**Key:** A = Analysis, D = Demonstration, I = Inspection, K = Analogy

## Appendix A: SRS Requirements Traceability Matrix *(continued)*

|        |          |                 |                      |         |  |
|--------|----------|-----------------|----------------------|---------|--|
|        | B06.03   | 3.2.2           | Process Arrival      | D       |  |
|        | B06.03.1 | 3.2.2           | Process Arrival      | I, D    |  |
|        | B06.03.2 | 3.2.2           | Process Arrival      | I, D    |  |
|        | B06.03.2 | 3.2.2           | Process Arrival      | I, D    |  |
|        | B06.03.4 | 3.2.3           | Process Departure    | D       |  |
|        | B06.03.5 | 3.2.3           | Process Departure    | D       |  |
|        | B06.03.6 | 1.              | Evaluate Input       | D       |  |
|        | B06.03.7 | 3.2.3           | Process Departure    | I, D    |  |
|        | B06.04   | 1.              | Evaluate Input       | D       |  |
| B07.00 |          | 2.1             | Initialize SimLib    | I, D    |  |
| B08.00 |          | 2.1             | Initialize SimLib    | A, I, D |  |
| B09.00 |          | 3.1             | Proc. Ext. Events    | D       |  |
|        | B09.01   | 3.1.2           | Process Insert       | D       |  |
|        | B09.02   | 3.1.1, 3.1.2    | Proc. Insert/Delete  | D       |  |
|        | B09.03   | 3.1.1, 3.1.2    | Proc. Insert/Delete  | D       |  |
| B10.00 |          | 3.1, GUI        | Proc. Ext. Events    | I, D    |  |
|        | B10.01   | 3.1.1, GUI      | Process Delete       | D       |  |
|        | B10.02   | 3.1.1, GUI      | Process Delete       | D       |  |
|        | B10.03   | 3.1.1, GUI      | Process Delete       | D       |  |
| B11.00 |          | 3.2.4           | Process Endsim       | A, I, D |  |
|        | B11.01   | 3.2.4, GUI      | Process Endsim       | D       |  |
|        | B11.02   | 3.2.4           | Process Endsim       | I       |  |
|        | B11.03   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.04   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.05   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.06   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.07   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.08   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.09   | 3.2.4           | Process Endsim       | A, D    |  |
|        | B11.10   | 3.2.4           | Process Endsim       | A, D    |  |
| C01.00 |          | 3.2             | Proc. Intern. Events | D       |  |
| C02.00 |          | 3.2             | Proc. Intern. Events | D       |  |
| D01.00 |          | 2.1             | Initialize SimLib    | I       |  |
|        | D01.01   | 2.1             | Initialize SimLib    | I       |  |
|        | D01.02   | 2.1             | Initialize SimLib    | I       |  |
|        | D01.03   | 2.1             | Initialize SimLib    | I       |  |
|        | D01.04   | 2.1             | Initialize SimLib    | I       |  |
| D02.00 |          | 1., 2., 3., GUI |                      | I       |  |
|        | D02.01   | 1., 2., 3., GUI |                      | I       |  |
| D03.00 |          | 2.1             | Initialize SimLib    | I       |  |
| D04.00 |          | 3.1.2           | Process Insert       | I       |  |
| D05.00 |          | 3.1.1           | Process Delete       | I       |  |
| D06.00 |          | 2.1             | Initialize SimLib    | I       |  |
|        | D06.01   | 2.1             | Initialize SimLib    | I       |  |
|        | D06.02   | 2.1             | Initialize SimLib    | I       |  |
| D07.00 |          | 1., 2., 3., GUI |                      | D       |  |

**Key:** **A** = Analysis, **D** = Demonstration, **I** = Inspection, **K** = Analogy

## APPENDIX B: The Individual Test Case Results

When the application has reached a point in the coding where the programmers feel it is ready to be tested it will be communicated to the test staff via a Test Transmittal report via e-mail initiated via the following database.

Figure 1: Test Transmittal Database Entry Screen

The testing process as accomplished by the Test Engineer's will produce a Test Incident report via e-mail to communicate the test-case results to the Design/Coding team.

Figure 2: Test Incident Database Entry Screen

## APPENDIX C: The Application Trace File (ATF)

There are several message types in the application trace file (ATF). They different messages are based upon the first character of each record in the ATF.

Arrival Event Scheduling = “greater than” symbol (>)

- The first number shown at the left is the scheduled arrival day number.
- In parentheses is the stream random number used to calculate the scheduled arrival date.
- At the end is the unique Plane Identifier (**pid**) followed by a forward slash (/) and the aircraft type number.

Departure Event Scheduling = “greater than” symbol (>)

- The number shown at the left is the scheduled departure day number.

Inspection Sub-Cycle Calculation = “plus sign” symbol (+)

- The first number is the cumulative time in the service station.
- The second number is the time length devoted to this specific engine inspection.
- In parentheses is the stream random number used to calculate this inspection period.
- In brackets is the engine number being inspected.
- The word “inspect”.
- The word “passed” or the word “failed” (*relative to “inspection”*).
- Random number for repair decision (**r=#**).
- Probability of an engine needing to be repaired (**p=#**).

Repair Sub-Cycle Calculation = “plus sign” symbol (+)

- The first number is the cumulative time in the service station.
- The second number is the time length devoted to this specific engine repair.
- In parentheses is the stream random number used to calculate this repair period.
- In brackets is the engine number being repaired.
- The word “repair”.

Event Handler, Aircraft Arrives = “left square bracket” symbol ([)

- The first number is the simulation clock time for this event [**t=#**].
- The Plane Identifier (**pid**) followed by a forward slash (/) and the aircraft type number.
- Queue length status; number of planes waiting already.

Event Handler, Aircraft Enters Service Station = “left square bracket” symbol ([)

- The first number is the simulation clock time for this event [**t=#**].
- The Plane Identifier (**pid**) followed by a forward slash (/) and the aircraft type number.
- Service Station Number (**bay #**).
- Optional Service Station Status (*if previously “idle”*).

## APPENDIX C: The Application Trace File (ATF)

Event Handler, Aircraft Enters Queue = “asterisk” symbol (\*)

- The unique Plane Identifier (**pid**) followed by a forward slash (/) and the aircraft type number. (*preceded by the words “all bays busy”*)

Event Handler, Aircraft Leaves Service Station = “left square bracket” symbol ([)

- The first number is the simulation clock time for this event [**t=#**].
- The Plane Identifier (**pid**) followed by a forward slash (/) and the aircraft type number.
- Service Station Number (**bay #**).

Event Handler, End of Simulation = “left square bracket” symbol ([)

- The first number is the simulation clock time for this event [**t=#**].
- The words “close doors event”.

The following is a listing of the application trace file (ATF) that is the result for **TCS01**.

```
> 4.84 (0.189) : scheduled arrival for pid 0/1
> 1.88 (0.593) : scheduled arrival for pid 1/2
> 5.06 (0.548) : scheduled arrival for pid 2/3
> 6.96 (0.528) : scheduled arrival for pid 3/4
> 4.36 (0.522) : scheduled arrival for pid 4/5
> 2.56 (0.427) : scheduled arrival for pid 5/6
> 0.00 (0.495) : scheduled arrival for pid 6/7
[t: 0.000] pid 6/7 arrival event - queue len: 0
> 0.57 (0.826) : scheduled arrival for pid 7/7
[t: 0.000] handling pid 6/7 in bay 1 (was idle)
+ 1.925/1.925 (0.250) : [1] inspect (passed; r=0.460; p=0.210)
+ 3.865/1.940 (0.397) : [2] inspect (passed; r=0.781; p=0.210)
+ 5.803/1.938 (0.381) : [3] inspect (passed; r=0.920; p=0.210)
> 5.80 : final bay departure scheduled
[t: 0.573] pid 7/7 arrival event - queue len: 0
> 10.77 (0.033) : scheduled arrival for pid 8/7
[t: 0.573] handling pid 7/7 in bay 2 (was idle)
+ 1.981/1.981 (0.814) : [1] inspect (passed; r=0.785; p=0.210)
+ 3.890/1.908 (0.083) : [2] inspect (passed; r=0.737; p=0.210)
+ 5.871/1.981 (0.809) : [3] inspect (failed; r=0.154; p=0.210)
+ 7.809/1.939 (0.277) : [3] repair
+ 8.795/0.986 (0.711) : [3] inspect (passed; r=0.262; p=0.105)
> 9.37 : final bay departure scheduled
[t: 1.882] pid 1/2 arrival event - queue len: 0
> 5.79 (0.259) : scheduled arrival for pid 9/2
[t: 1.882] handling pid 1/2 in bay 3 (was idle)
+ 1.959/1.959 (0.471) : [1] inspect (passed; r=0.610; p=0.260)
+ 3.874/1.915 (0.615) : [2] inspect (passed; r=0.735; p=0.260)
+ 5.813/1.939 (0.536) : [3] inspect (passed; r=0.718; p=0.260)
> 7.70 : final bay departure scheduled
[t: 2.556] pid 5/6 arrival event - queue len: 0
> 8.33 (0.422) : scheduled arrival for pid 10/6
[t: 2.556] handling pid 5/6 in bay 4 (was idle)
+ 1.652/1.652 (0.522) : [1] inspect (passed; r=0.866; p=0.140)
+ 3.275/1.623 (0.233) : [2] inspect (passed; r=0.777; p=0.140)
> 5.83 : final bay departure scheduled
[t: 4.358] pid 4/5 arrival event - queue len: 0
> 4.45 (0.992) : scheduled arrival for pid 11/5
[t: 4.358] handling pid 4/5 in bay 5 (was idle)
+ 2.028/2.028 (0.877) : [1] inspect (failed; r=0.252; p=0.360)
+ 5.178/3.150 (0.287) : [1] repair
+ 6.269/1.091 (0.987) : [1] inspect (failed; r=0.145; p=0.180)
+ 6.948/0.679 (0.412) : [1] repair
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 7.654/0.706 (0.437) : [1] inspect (passed; r=0.325; p=0.180)
+ 8.735/1.081 (0.200) : [2] inspect (failed; r=0.022; p=0.360)
+12.278/3.544 (0.417) : [2] repair
+12.940/0.661 (0.373) : [2] inspect (passed; r=0.469; p=0.180)
+15.110/2.170 (0.979) : [3] inspect (passed; r=0.754; p=0.360)
+16.591/1.481 (0.487) : [4] inspect (failed; r=0.112; p=0.360)
+17.605/1.013 (0.969) : [4] repair
+18.338/0.734 (0.477) : [4] inspect (failed; r=0.056; p=0.180)
+18.933/0.595 (0.350) : [4] repair
+19.909/0.976 (0.823) : [4] inspect (failed; r=0.172; p=0.180)
+21.500/1.591 (0.228) : [4] repair
+22.339/0.839 (0.627) : [4] inspect (passed; r=0.728; p=0.180)
> 26.70 : final bay departure scheduled
[t: 4.446] pid 11/5 arrival event - queue len: 0
> 4.73 (0.974) : scheduled arrival for pid 12/5
[t: 4.446] handling pid 11/5 in bay 6 (was idle)
+ 1.011/1.011 (0.151) : [1] inspect (passed; r=0.403; p=0.360)
+ 2.901/1.890 (0.778) : [2] inspect (passed; r=0.655; p=0.360)
+ 4.114/1.214 (0.295) : [3] inspect (passed; r=0.485; p=0.360)
+ 5.097/0.983 (0.130) : [4] inspect (passed; r=0.754; p=0.360)
> 9.54 : final bay departure scheduled
[t: 4.733] pid 12/5 arrival event - queue len: 0
> 8.94 (0.680) : scheduled arrival for pid 13/5
* all bays busy; inserting pid 12/5 in queue
[t: 4.835] pid 0/1 arrival event - queue len: 1
> 12.25 (0.401) : scheduled arrival for pid 14/1
* all bays busy; inserting pid 0/1 in queue
[t: 5.058] pid 2/3 arrival event - queue len: 2
> 8.04 (0.436) : scheduled arrival for pid 15/3
* all bays busy; inserting pid 2/3 in queue
[t: 5.795] pid 9/2 arrival event - queue len: 3
> 7.38 (0.578) : scheduled arrival for pid 16/2
* all bays busy; inserting pid 9/2 in queue
[t: 5.803] pid 6/7 departure event (bay 1)
[t: 5.803] handling pid 12/5 in bay 1
+ 1.948/1.948 (0.482) : [1] inspect (passed; r=0.759; p=0.210)
+ 3.893/1.945 (0.448) : [2] inspect (passed; r=0.877; p=0.210)
+ 5.864/1.971 (0.708) : [3] inspect (failed; r=0.169; p=0.210)
+16.436/10.572 (0.004) : [3] repair
+17.429/0.993 (0.867) : [3] inspect (passed; r=0.778; p=0.105)
> 23.23 : final bay departure scheduled
[t: 5.832] pid 5/6 departure event (bay 4)
[t: 5.832] handling pid 0/1 in bay 4
+ 1.685/1.685 (0.853) : [1] inspect (passed; r=0.198; p=0.140)
+ 3.330/1.645 (0.446) : [2] inspect (passed; r=0.423; p=0.140)
> 9.16 : final bay departure scheduled
[t: 6.959] pid 3/4 arrival event - queue len: 2
> 20.06 (0.210) : scheduled arrival for pid 17/4
* all bays busy; inserting pid 3/4 in queue
[t: 7.384] pid 16/2 arrival event - queue len: 3
> 8.65 (0.645) : scheduled arrival for pid 18/2
* all bays busy; inserting pid 16/2 in queue
[t: 7.695] pid 1/2 departure event (bay 3)
[t: 7.695] handling pid 2/3 in bay 3
+ 1.921/1.921 (0.596) : [1] inspect (passed; r=0.786; p=0.260)
+ 3.900/1.978 (0.406) : [2] inspect (passed; r=0.577; p=0.260)
+ 5.810/1.911 (0.631) : [3] inspect (passed; r=0.299; p=0.260)
> 13.51 : final bay departure scheduled
[t: 8.042] pid 15/3 arrival event - queue len: 3
> 10.15 (0.557) : scheduled arrival for pid 19/3
* all bays busy; inserting pid 15/3 in queue
[t: 8.334] pid 10/6 arrival event - queue len: 4
> 15.43 (0.347) : scheduled arrival for pid 20/6
* all bays busy; inserting pid 10/6 in queue
[t: 8.654] pid 18/2 arrival event - queue len: 5
```

## APPENDIX C: The Application Trace File (ATF)

```
> 12.73 (0.245) : scheduled arrival for pid 21/2
* all bays busy; inserting pid 18/2 in queue
[t: 8.938] pid 13/5 arrival event - queue len: 6
> 9.50 (0.950) : scheduled arrival for pid 22/5
* all bays busy; inserting pid 13/5 in queue
[t: 9.162] pid 0/1 departure event (bay 4)
[t: 9.162] handling pid 9/2 in bay 4
+ 1.547/1.547 (0.605) : [1] inspect (failed; r=0.231; p=0.300)
+ 5.034/3.487 (0.668) : [1] repair
+ 5.669/0.635 (0.406) : [1] inspect (passed; r=0.938; p=0.150)
+ 6.803/1.134 (0.310) : [2] inspect (passed; r=0.663; p=0.300)
+ 7.844/1.041 (0.244) : [3] inspect (passed; r=0.404; p=0.300)
+ 9.353/1.509 (0.578) : [4] inspect (failed; r=0.212; p=0.300)
+12.169/2.815 (0.168) : [4] repair
+12.851/0.683 (0.475) : [4] inspect (passed; r=0.198; p=0.150)
> 22.01 : final bay departure scheduled
[t: 9.368] pid 7/7 departure event (bay 2)
[t: 9.368] handling pid 3/4 in bay 2
+ 1.952/1.952 (0.523) : [1] inspect (passed; r=0.449; p=0.210)
+ 3.863/1.910 (0.102) : [2] inspect (passed; r=0.453; p=0.210)
+ 5.839/1.977 (0.769) : [3] inspect (passed; r=0.295; p=0.210)
> 15.21 : final bay departure scheduled
[t: 9.502] pid 22/5 arrival event - queue len: 5
> 47.50 (0.031) : scheduled arrival for pid 23/5
* all bays busy; inserting pid 22/5 in queue
[t: 9.543] pid 11/5 departure event (bay 6)
[t: 9.543] handling pid 16/2 in bay 6
+ 1.005/1.005 (0.146) : [1] inspect (failed; r=0.236; p=0.360)
+ 3.477/2.472 (0.159) : [1] repair
+ 4.507/1.030 (0.901) : [1] inspect (passed; r=0.866; p=0.180)
+ 6.111/1.604 (0.574) : [2] inspect (passed; r=0.533; p=0.360)
+ 8.263/2.152 (0.965) : [3] inspect (failed; r=0.247; p=0.360)
+ 9.871/1.609 (0.260) : [3] repair
+10.693/0.822 (0.603) : [3] inspect (passed; r=0.227; p=0.180)
+12.047/1.354 (0.396) : [4] inspect (passed; r=0.975; p=0.360)
> 21.59 : final bay departure scheduled
[t: 10.152] pid 19/3 arrival event - queue len: 5
> 11.88 (0.618) : scheduled arrival for pid 24/3
* all bays busy; inserting pid 19/3 in queue
[t: 10.772] pid 8/7 arrival event - queue len: 6
> 11.86 (0.696) : scheduled arrival for pid 25/7
* all bays busy; inserting pid 8/7 in queue
[t: 11.860] pid 25/7 arrival event - queue len: 7
> 14.91 (0.362) : scheduled arrival for pid 26/7
* all bays busy; inserting pid 25/7 in queue
[t: 11.885] pid 24/3 arrival event - queue len: 8
> 12.83 (0.769) : scheduled arrival for pid 27/3
* all bays busy; inserting pid 24/3 in queue
[t: 12.247] pid 14/1 arrival event - queue len: 9
> 16.20 (0.614) : scheduled arrival for pid 28/1
* all bays busy; inserting pid 14/1 in queue
[t: 12.729] pid 21/2 arrival event - queue len: 10
> 13.18 (0.857) : scheduled arrival for pid 29/2
* all bays busy; inserting pid 21/2 in queue
[t: 12.830] pid 27/3 arrival event - queue len: 11
> 14.70 (0.595) : scheduled arrival for pid 30/3
* all bays busy; inserting pid 27/3 in queue
[t: 13.176] pid 29/2 arrival event - queue len: 12
> 13.28 (0.965) : scheduled arrival for pid 31/2
* all bays busy; inserting pid 29/2 in queue
[t: 13.279] pid 31/2 arrival event - queue len: 13
> 14.45 (0.669) : scheduled arrival for pid 32/2
* all bays busy; inserting pid 31/2 in queue
[t: 13.505] pid 2/3 departure event (bay 3)
[t: 13.505] handling pid 15/3 in bay 3
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 1.261/1.261 (0.516) : [1] inspect (passed; r=0.335; p=0.180)
+ 2.789/1.528 (0.897) : [2] inspect (passed; r=0.778; p=0.180)
> 16.29 : final bay departure scheduled
[t: 14.446] pid 32/2 arrival event - queue len: 13
> 18.51 (0.247) : scheduled arrival for pid 33/2
* all bays busy; inserting pid 32/2 in queue
[t: 14.698] pid 30/3 arrival event - queue len: 14
> 26.11 (0.042) : scheduled arrival for pid 34/3
* all bays busy; inserting pid 30/3 in queue
[t: 14.912] pid 26/7 arrival event - queue len: 15
> 16.08 (0.678) : scheduled arrival for pid 35/7
* all bays busy; inserting pid 26/7 in queue
[t: 15.208] pid 3/4 departure event (bay 2)
[t: 15.208] handling pid 10/6 in bay 2
+ 2.277/2.277 (0.477) : [1] inspect (passed; r=0.437; p=0.120)
+ 4.545/2.268 (0.468) : [2] inspect (passed; r=0.702; p=0.120)
+ 6.551/2.006 (0.206) : [3] inspect (passed; r=0.159; p=0.120)
+ 8.497/1.946 (0.146) : [4] inspect (passed; r=0.164; p=0.120)
> 23.70 : final bay departure scheduled
[t: 15.435] pid 20/6 arrival event - queue len: 15
> 57.77 (0.002) : scheduled arrival for pid 36/6
* all bays busy; inserting pid 20/6 in queue
[t: 16.077] pid 35/7 arrival event - queue len: 16
> 22.33 (0.124) : scheduled arrival for pid 37/7
* all bays busy; inserting pid 35/7 in queue
[t: 16.203] pid 28/1 arrival event - queue len: 17
> 22.97 (0.434) : scheduled arrival for pid 38/1
* all bays busy; inserting pid 28/1 in queue
[t: 16.295] pid 15/3 departure event (bay 3)
[t: 16.295] handling pid 18/2 in bay 3
+ 1.090/1.090 (0.271) : [1] inspect (passed; r=0.402; p=0.180)
+ 2.016/0.926 (0.038) : [2] inspect (passed; r=0.769; p=0.180)
> 18.31 : final bay departure scheduled
[t: 18.311] pid 18/2 departure event (bay 3)
[t: 18.311] handling pid 13/5 in bay 3
+ 1.893/1.893 (0.689) : [1] inspect (passed; r=0.750; p=0.260)
+ 3.708/1.815 (0.951) : [2] inspect (passed; r=0.319; p=0.260)
+ 5.700/1.991 (0.362) : [3] inspect (passed; r=0.845; p=0.260)
> 24.01 : final bay departure scheduled
[t: 18.506] pid 33/2 arrival event - queue len: 16
> 24.16 (0.142) : scheduled arrival for pid 39/2
* all bays busy; inserting pid 33/2 in queue
[t: 20.062] pid 17/4 arrival event - queue len: 17
> 20.44 (0.956) : scheduled arrival for pid 40/4
* all bays busy; inserting pid 17/4 in queue
[t: 20.436] pid 40/4 arrival event - queue len: 18
> 24.74 (0.599) : scheduled arrival for pid 41/4
* all bays busy; inserting pid 40/4 in queue
[t: 21.590] pid 16/2 departure event (bay 6)
[t: 21.590] handling pid 22/5 in bay 6
+ 2.033/2.033 (0.222) : [1] inspect (passed; r=0.943; p=0.260)
+ 3.922/1.889 (0.705) : [2] inspect (passed; r=0.381; p=0.260)
+ 5.933/2.011 (0.297) : [3] inspect (passed; r=0.966; p=0.260)
> 27.52 : final bay departure scheduled
[t: 22.013] pid 9/2 departure event (bay 4)
[t: 22.013] handling pid 19/3 in bay 4
+ 1.876/1.876 (0.748) : [1] inspect (passed; r=0.661; p=0.260)
+ 3.890/2.014 (0.285) : [2] inspect (passed; r=0.757; p=0.260)
+ 5.864/1.974 (0.421) : [3] inspect (passed; r=0.539; p=0.260)
> 27.88 : final bay departure scheduled
[t: 22.332] pid 37/7 arrival event - queue len: 17
> 22.86 (0.838) : scheduled arrival for pid 42/7
* all bays busy; inserting pid 37/7 in queue
[t: 22.863] pid 42/7 arrival event - queue len: 18
> 23.16 (0.905) : scheduled arrival for pid 43/7
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 42/7 in queue
[t: 22.972] pid 38/1 arrival event - queue len: 19
> 30.75 (0.383) : scheduled arrival for pid 44/1
* all bays busy; inserting pid 38/1 in queue
[t: 23.162] pid 43/7 arrival event - queue len: 20
> 31.99 (0.053) : scheduled arrival for pid 45/7
* all bays busy; inserting pid 43/7 in queue
[t: 23.232] pid 12/5 departure event (bay 1)
[t: 23.232] handling pid 8/7 in bay 1
+ 1.838/1.838 (0.741) : [1] inspect (passed; r=0.824; p=0.360)
+ 2.950/1.112 (0.223) : [2] inspect (passed; r=0.690; p=0.360)
+ 4.309/1.360 (0.400) : [3] inspect (failed; r=0.137; p=0.360)
+ 5.772/1.463 (0.453) : [3] repair
+ 6.782/1.010 (0.871) : [3] inspect (passed; r=0.480; p=0.180)
+ 8.463/1.681 (0.629) : [4] inspect (passed; r=0.968; p=0.360)
> 31.69 : final bay departure scheduled
[t: 23.704] pid 10/6 departure event (bay 2)
[t: 23.704] handling pid 25/7 in bay 2
+ 1.697/1.697 (0.974) : [1] inspect (passed; r=0.300; p=0.140)
+ 3.394/1.696 (0.964) : [2] inspect (failed; r=0.110; p=0.140)
+ 5.826/2.432 (0.761) : [2] repair
+ 6.654/0.828 (0.552) : [2] inspect (passed; r=0.088; p=0.070)
> 30.36 : final bay departure scheduled
[t: 24.010] pid 13/5 departure event (bay 3)
[t: 24.010] handling pid 24/3 in bay 3
+ 2.112/2.112 (0.937) : [1] inspect (failed; r=0.277; p=0.360)
+ 4.487/2.375 (0.266) : [1] repair
+ 5.285/0.798 (0.569) : [1] inspect (failed; r=0.141; p=0.180)
+ 6.482/1.198 (0.290) : [1] repair
+ 7.024/0.542 (0.202) : [1] inspect (passed; r=0.597; p=0.180)
+ 9.056/2.032 (0.880) : [2] inspect (failed; r=0.120; p=0.360)
+ 9.258/0.202 (0.950) : [2] repair
+ 9.823/0.565 (0.235) : [2] inspect (passed; r=0.724; p=0.180)
+11.025/1.202 (0.287) : [3] inspect (failed; r=0.213; p=0.360)
+13.163/2.137 (0.320) : [3] repair
+13.588/0.426 (0.037) : [3] inspect (failed; r=0.070; p=0.180)
+13.598/0.010 (0.995) : [3] repair
+14.562/0.964 (0.806) : [3] inspect (passed; r=0.308; p=0.180)
+15.949/1.387 (0.419) : [4] inspect (passed; r=0.924; p=0.360)
> 39.96 : final bay departure scheduled
[t: 24.163] pid 39/2 arrival event - queue len: 18
> 24.58 (0.866) : scheduled arrival for pid 46/2
* all bays busy; inserting pid 39/2 in queue
[t: 24.581] pid 46/2 arrival event - queue len: 19
> 30.03 (0.153) : scheduled arrival for pid 47/2
* all bays busy; inserting pid 46/2 in queue
[t: 24.737] pid 41/4 arrival event - queue len: 20
> 25.91 (0.869) : scheduled arrival for pid 48/4
* all bays busy; inserting pid 41/4 in queue
[t: 25.915] pid 48/4 arrival event - queue len: 21
> 49.40 (0.061) : scheduled arrival for pid 49/4
* all bays busy; inserting pid 48/4 in queue
[t: 26.111] pid 34/3 arrival event - queue len: 22
> 28.18 (0.563) : scheduled arrival for pid 50/3
* all bays busy; inserting pid 34/3 in queue
[t: 26.696] pid 4/5 departure event (bay 5)
[t: 26.696] handling pid 14/1 in bay 5
+ 1.024/1.024 (0.160) : [1] inspect (failed; r=0.226; p=0.360)
+ 3.456/2.432 (0.866) : [1] repair
+ 4.020/0.564 (0.235) : [1] inspect (passed; r=0.222; p=0.180)
+ 5.558/1.538 (0.527) : [2] inspect (failed; r=0.210; p=0.360)
+10.263/4.705 (0.436) : [2] repair
+11.072/0.809 (0.585) : [2] inspect (passed; r=0.602; p=0.180)
+12.658/1.586 (0.562) : [3] inspect (passed; r=0.621; p=0.360)
+13.719/1.060 (0.186) : [4] inspect (failed; r=0.336; p=0.360)
```

## APPENDIX C: The Application Trace File (ATF)

```
+15.735/2.016 (0.408) : [4] repair
+16.529/0.795 (0.564) : [4] inspect (passed; r=0.518; p=0.180)
> 43.23 : final bay departure scheduled
[t: 27.523] pid 22/5 departure event (bay 6)
[t: 27.523] handling pid 21/2 in bay 6
+ 1.240/1.240 (0.314) : [1] inspect (passed; r=0.509; p=0.360)
+ 2.327/1.088 (0.206) : [2] inspect (passed; r=0.999; p=0.360)
+ 4.385/2.058 (0.899) : [3] inspect (passed; r=0.573; p=0.360)
+ 6.033/1.648 (0.605) : [4] inspect (failed; r=0.139; p=0.360)
+11.009/4.975 (0.028) : [4] repair
+11.912/0.904 (0.720) : [4] inspect (passed; r=0.753; p=0.180)
> 39.44 : final bay departure scheduled
[t: 27.877] pid 19/3 departure event (bay 4)
[t: 27.877] handling pid 27/3 in bay 4
+ 1.379/1.379 (0.684) : [1] inspect (passed; r=0.775; p=0.180)
+ 2.465/1.086 (0.266) : [2] inspect (failed; r=0.062; p=0.180)
+ 3.073/0.607 (0.976) : [2] repair
+ 3.744/0.671 (0.631) : [2] inspect (passed; r=0.963; p=0.090)
> 31.62 : final bay departure scheduled
[t: 28.176] pid 50/3 arrival event - queue len: 20
> 34.93 (0.153) : scheduled arrival for pid 51/3
* all bays busy; inserting pid 50/3 in queue
[t: 30.027] pid 47/2 arrival event - queue len: 21
> 33.80 (0.273) : scheduled arrival for pid 52/2
* all bays busy; inserting pid 47/2 in queue
[t: 30.358] pid 25/7 departure event (bay 2)
[t: 30.358] handling pid 29/2 in bay 2
+ 1.984/1.984 (0.839) : [1] inspect (passed; r=0.733; p=0.210)
+ 3.923/1.939 (0.386) : [2] inspect (passed; r=0.558; p=0.210)
+ 5.831/1.908 (0.084) : [3] inspect (failed; r=0.185; p=0.210)
+ 8.002/2.171 (0.263) : [3] repair
+ 8.964/0.962 (0.245) : [3] inspect (passed; r=0.822; p=0.105)
> 39.32 : final bay departure scheduled
[t: 30.745] pid 44/1 arrival event - queue len: 21
> 36.26 (0.506) : scheduled arrival for pid 53/1
* all bays busy; inserting pid 44/1 in queue
[t: 31.620] pid 27/3 departure event (bay 4)
[t: 31.620] handling pid 31/2 in bay 4
+ 1.399/1.399 (0.712) : [1] inspect (passed; r=0.323; p=0.180)
+ 2.912/1.513 (0.876) : [2] inspect (passed; r=0.887; p=0.180)
> 34.53 : final bay departure scheduled
[t: 31.695] pid 8/7 departure event (bay 1)
[t: 31.695] handling pid 32/2 in bay 1
+ 1.946/1.946 (0.460) : [1] inspect (passed; r=0.315; p=0.210)
+ 3.934/1.988 (0.880) : [2] inspect (failed; r=0.180; p=0.210)
+ 5.542/1.608 (0.498) : [2] repair
+ 6.540/0.998 (0.961) : [2] inspect (passed; r=0.148; p=0.105)
+ 8.441/1.901 (0.013) : [3] inspect (passed; r=0.648; p=0.210)
> 40.14 : final bay departure scheduled
[t: 31.993] pid 45/7 arrival event - queue len: 20
> 36.92 (0.194) : scheduled arrival for pid 54/7
* all bays busy; inserting pid 45/7 in queue
[t: 33.797] pid 52/2 arrival event - queue len: 21
> 34.20 (0.869) : scheduled arrival for pid 55/2
* all bays busy; inserting pid 52/2 in queue
[t: 34.204] pid 55/2 arrival event - queue len: 22
> 38.04 (0.267) : scheduled arrival for pid 56/2
* all bays busy; inserting pid 55/2 in queue
[t: 34.532] pid 31/2 departure event (bay 4)
[t: 34.532] handling pid 30/3 in bay 4
+ 2.049/2.049 (0.169) : [1] inspect (passed; r=0.963; p=0.260)
+ 3.916/1.866 (0.779) : [2] inspect (passed; r=0.726; p=0.260)
+ 5.776/1.861 (0.797) : [3] inspect (failed; r=0.084; p=0.260)
+ 8.858/3.081 (0.124) : [3] repair
+ 9.779/0.921 (0.857) : [3] inspect (failed; r=0.090; p=0.130)
```

## APPENDIX C: The Application Trace File (ATF)

```
+11.433/1.654 (0.074) : [3] repair
+12.395/0.962 (0.586) : [3] inspect (passed; r=0.291; p=0.130)
> 46.93 : final bay departure scheduled
[t: 34.929] pid 51/3 arrival event - queue len: 22
> 36.25 (0.693) : scheduled arrival for pid 57/3
* all bays busy; inserting pid 51/3 in queue
[t: 36.247] pid 57/3 arrival event - queue len: 23
> 39.53 (0.401) : scheduled arrival for pid 58/3
* all bays busy; inserting pid 57/3 in queue
[t: 36.261] pid 53/1 arrival event - queue len: 24
> 39.05 (0.709) : scheduled arrival for pid 59/1
* all bays busy; inserting pid 53/1 in queue
[t: 36.918] pid 54/7 arrival event - queue len: 25
> 37.79 (0.748) : scheduled arrival for pid 60/7
* all bays busy; inserting pid 54/7 in queue
[t: 37.789] pid 60/7 arrival event - queue len: 26
> 40.50 (0.405) : scheduled arrival for pid 61/7
* all bays busy; inserting pid 60/7 in queue
[t: 38.037] pid 56/2 arrival event - queue len: 27
> 38.59 (0.826) : scheduled arrival for pid 62/2
* all bays busy; inserting pid 56/2 in queue
[t: 38.590] pid 62/2 arrival event - queue len: 28
> 38.73 (0.953) : scheduled arrival for pid 63/2
* all bays busy; inserting pid 62/2 in queue
[t: 38.730] pid 63/2 arrival event - queue len: 29
> 42.13 (0.309) : scheduled arrival for pid 64/2
* all bays busy; inserting pid 63/2 in queue
[t: 39.045] pid 59/1 arrival event - queue len: 30
> 52.71 (0.185) : scheduled arrival for pid 65/1
* all bays busy; inserting pid 59/1 in queue
[t: 39.322] pid 29/2 departure event (bay 2)
[t: 39.322] handling pid 26/7 in bay 2
+ 1.997/1.997 (0.345) : [1] inspect (passed; r=0.674; p=0.260)
+ 3.995/1.999 (0.338) : [2] inspect (passed; r=0.644; p=0.260)
+ 5.946/1.951 (0.498) : [3] inspect (passed; r=0.933; p=0.260)
> 45.27 : final bay departure scheduled
[t: 39.435] pid 21/2 departure event (bay 6)
[t: 39.435] handling pid 20/6 in bay 6
+ 1.997/1.997 (0.344) : [1] inspect (passed; r=0.475; p=0.260)
+ 3.872/1.875 (0.750) : [2] inspect (passed; r=0.327; p=0.260)
+ 5.761/1.889 (0.705) : [3] inspect (failed; r=0.227; p=0.260)
+ 6.175/0.415 (0.756) : [3] repair
+ 7.179/1.004 (0.309) : [3] inspect (passed; r=0.386; p=0.130)
> 46.61 : final bay departure scheduled
[t: 39.534] pid 58/3 arrival event - queue len: 29
> 40.49 (0.767) : scheduled arrival for pid 66/3
* all bays busy; inserting pid 58/3 in queue
[t: 39.959] pid 24/3 departure event (bay 3)
[t: 39.959] handling pid 35/7 in bay 3
+ 1.381/1.381 (0.688) : [1] inspect (passed; r=0.399; p=0.180)
+ 2.817/1.435 (0.765) : [2] inspect (passed; r=0.539; p=0.180)
> 42.78 : final bay departure scheduled
[t: 40.136] pid 32/2 departure event (bay 1)
[t: 40.136] handling pid 28/1 in bay 1
+ 1.906/1.906 (0.646) : [1] inspect (failed; r=0.097; p=0.260)
+ 2.965/1.058 (0.419) : [1] repair
+ 3.871/0.906 (0.959) : [1] inspect (passed; r=0.730; p=0.130)
+ 5.792/1.921 (0.598) : [2] inspect (passed; r=0.480; p=0.260)
+ 7.851/2.060 (0.134) : [3] inspect (passed; r=0.558; p=0.260)
> 47.99 : final bay departure scheduled
[t: 40.487] pid 66/3 arrival event - queue len: 28
> 42.33 (0.599) : scheduled arrival for pid 67/3
* all bays busy; inserting pid 66/3 in queue
[t: 40.499] pid 61/7 arrival event - queue len: 29
> 47.05 (0.112) : scheduled arrival for pid 68/7
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 61/7 in queue
[t: 42.133] pid 64/2 arrival event - queue len: 30
> 48.00 (0.132) : scheduled arrival for pid 69/2
* all bays busy; inserting pid 64/2 in queue
[t: 42.333] pid 67/3 arrival event - queue len: 31
> 44.86 (0.495) : scheduled arrival for pid 70/3
* all bays busy; inserting pid 67/3 in queue
[t: 42.776] pid 35/7 departure event (bay 3)
[t: 42.776] handling pid 33/2 in bay 3
+ 1.950/1.950 (0.504) : [1] inspect (passed; r=0.297; p=0.210)
+ 3.871/1.920 (0.204) : [2] inspect (passed; r=0.782; p=0.210)
+ 5.773/1.902 (0.019) : [3] inspect (passed; r=0.760; p=0.210)
> 48.55 : final bay departure scheduled
[t: 43.226] pid 14/1 departure event (bay 5)
[t: 43.226] handling pid 17/4 in bay 5
+ 1.066/1.066 (0.261) : [1] inspect (failed; r=0.178; p=0.300)
+ 8.750/7.684 (0.838) : [1] repair
+ 9.705/0.955 (0.865) : [1] inspect (passed; r=0.391; p=0.150)
+10.805/1.100 (0.286) : [2] inspect (passed; r=0.770; p=0.300)
+12.498/1.693 (0.709) : [3] inspect (passed; r=0.852; p=0.300)
+14.046/1.548 (0.606) : [4] inspect (passed; r=0.560; p=0.300)
> 57.27 : final bay departure scheduled
[t: 44.862] pid 70/3 arrival event - queue len: 30
> 45.54 (0.828) : scheduled arrival for pid 71/3
* all bays busy; inserting pid 70/3 in queue
[t: 45.268] pid 26/7 departure event (bay 2)
[t: 45.268] handling pid 40/4 in bay 2
+ 1.973/1.973 (0.726) : [1] inspect (passed; r=0.609; p=0.210)
+ 3.912/1.939 (0.395) : [2] inspect (failed; r=0.040; p=0.210)
+ 9.381/5.469 (0.988) : [2] repair
+10.342/0.961 (0.219) : [2] inspect (passed; r=0.394; p=0.105)
+12.269/1.927 (0.270) : [3] inspect (failed; r=0.106; p=0.210)
+17.370/5.101 (0.229) : [3] repair
+18.320/0.950 (0.006) : [3] inspect (passed; r=0.690; p=0.105)
> 63.59 : final bay departure scheduled
[t: 45.542] pid 71/3 arrival event - queue len: 30
> 48.70 (0.416) : scheduled arrival for pid 72/3
* all bays busy; inserting pid 71/3 in queue
[t: 46.614] pid 20/6 departure event (bay 6)
[t: 46.614] handling pid 37/7 in bay 6
+ 1.694/1.694 (0.938) : [1] inspect (passed; r=0.179; p=0.140)
+ 3.357/1.664 (0.635) : [2] inspect (passed; r=0.563; p=0.140)
> 49.97 : final bay departure scheduled
[t: 46.928] pid 30/3 departure event (bay 4)
[t: 46.928] handling pid 42/7 in bay 4
+ 1.021/1.021 (0.172) : [1] inspect (passed; r=0.208; p=0.180)
+ 2.111/1.090 (0.271) : [2] inspect (passed; r=0.896; p=0.180)
> 49.04 : final bay departure scheduled
[t: 47.054] pid 68/7 arrival event - queue len: 29
> 50.96 (0.272) : scheduled arrival for pid 73/7
* all bays busy; inserting pid 68/7 in queue
[t: 47.499] pid 23/5 arrival event - queue len: 30
> 60.04 (0.316) : scheduled arrival for pid 74/5
* all bays busy; inserting pid 23/5 in queue
[t: 47.987] pid 28/1 departure event (bay 1)
[t: 47.987] handling pid 38/1 in bay 1
+ 1.044/1.044 (0.245) : [1] inspect (passed; r=0.338; p=0.300)
+ 1.771/0.727 (0.019) : [2] inspect (passed; r=0.317; p=0.300)
+ 3.809/2.038 (0.956) : [3] inspect (passed; r=0.343; p=0.300)
+ 5.853/2.044 (0.960) : [4] inspect (passed; r=0.708; p=0.300)
> 53.84 : final bay departure scheduled
[t: 47.999] pid 69/2 arrival event - queue len: 30
> 53.92 (0.130) : scheduled arrival for pid 75/2
* all bays busy; inserting pid 69/2 in queue
[t: 48.548] pid 33/2 departure event (bay 3)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 48.548] handling pid 43/7 in bay 3
+ 1.845/1.845 (0.849) : [1] inspect (failed; r=0.237; p=0.260)
+ 2.306/0.461 (0.755) : [1] repair
+ 3.243/0.936 (0.759) : [1] inspect (passed; r=0.825; p=0.130)
+ 5.142/1.899 (0.669) : [2] inspect (failed; r=0.062; p=0.260)
+ 5.253/0.111 (0.927) : [2] repair
+ 6.265/1.013 (0.248) : [2] inspect (passed; r=0.293; p=0.130)
+ 8.319/2.053 (0.156) : [3] inspect (failed; r=0.042; p=0.260)
+ 9.462/1.144 (0.694) : [3] repair
+10.370/0.908 (0.948) : [3] inspect (passed; r=0.176; p=0.130)
> 58.92 : final bay departure scheduled
[t: 48.698] pid 72/3 arrival event - queue len: 30
> 49.83 (0.730) : scheduled arrival for pid 76/3
* all bays busy; inserting pid 72/3 in queue
[t: 49.038] pid 42/7 departure event (bay 4)
[t: 49.038] handling pid 39/2 in bay 4
+ 1.992/1.992 (0.917) : [1] inspect (passed; r=0.663; p=0.210)
+ 3.949/1.957 (0.572) : [2] inspect (failed; r=0.186; p=0.210)
+ 7.249/3.300 (0.432) : [2] repair
+ 8.240/0.991 (0.811) : [2] inspect (failed; r=0.105; p=0.105)
+10.512/2.272 (0.408) : [2] repair
+11.496/0.984 (0.677) : [2] inspect (passed; r=0.940; p=0.105)
+13.493/1.997 (0.969) : [3] inspect (passed; r=0.644; p=0.210)
> 62.53 : final bay departure scheduled
[t: 49.402] pid 49/4 arrival event - queue len: 30
> 50.84 (0.843) : scheduled arrival for pid 77/4
* all bays busy; inserting pid 49/4 in queue
[t: 49.833] pid 76/3 arrival event - queue len: 31
> 50.65 (0.796) : scheduled arrival for pid 78/3
* all bays busy; inserting pid 76/3 in queue
[t: 49.972] pid 37/7 departure event (bay 6)
[t: 49.972] handling pid 46/2 in bay 6
+ 1.960/1.960 (0.596) : [1] inspect (passed; r=0.236; p=0.210)
+ 3.902/1.942 (0.423) : [2] inspect (passed; r=0.579; p=0.210)
+ 5.838/1.937 (0.365) : [3] inspect (failed; r=0.011; p=0.210)
+ 7.206/1.368 (0.386) : [3] repair
+ 8.190/0.984 (0.676) : [3] inspect (passed; r=0.980; p=0.105)
> 58.16 : final bay departure scheduled
[t: 50.654] pid 78/3 arrival event - queue len: 31
> 51.44 (0.804) : scheduled arrival for pid 79/3
* all bays busy; inserting pid 78/3 in queue
[t: 50.835] pid 77/4 arrival event - queue len: 32
> 56.91 (0.485) : scheduled arrival for pid 80/4
* all bays busy; inserting pid 77/4 in queue
[t: 50.959] pid 73/7 arrival event - queue len: 33
> 54.03 (0.359) : scheduled arrival for pid 81/7
* all bays busy; inserting pid 73/7 in queue
[t: 51.437] pid 79/3 arrival event - queue len: 34
> 52.44 (0.757) : scheduled arrival for pid 82/3
* all bays busy; inserting pid 79/3 in queue
[t: 52.437] pid 82/3 arrival event - queue len: 35
> 57.49 (0.246) : scheduled arrival for pid 83/3
* all bays busy; inserting pid 82/3 in queue
[t: 52.709] pid 65/1 arrival event - queue len: 36
> 54.18 (0.834) : scheduled arrival for pid 84/1
* all bays busy; inserting pid 65/1 in queue
[t: 53.840] pid 38/1 departure event (bay 1)
[t: 53.840] handling pid 41/4 in bay 1
+ 0.702/0.702 (0.001) : [1] inspect (passed; r=0.872; p=0.300)
+ 1.569/0.867 (0.120) : [2] inspect (passed; r=0.494; p=0.300)
+ 3.417/1.847 (0.820) : [3] inspect (failed; r=0.182; p=0.300)
+ 6.922/3.506 (0.090) : [3] repair
+ 7.378/0.456 (0.151) : [3] inspect (passed; r=0.953; p=0.150)
+ 8.673/1.294 (0.425) : [4] inspect (passed; r=0.319; p=0.300)
> 62.51 : final bay departure scheduled
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 53.923] pid 75/2 arrival event - queue len: 36
> 54.66 (0.776) : scheduled arrival for pid 85/2
* all bays busy; inserting pid 75/2 in queue
[t: 54.031] pid 81/7 arrival event - queue len: 37
> 56.27 (0.474) : scheduled arrival for pid 86/7
* all bays busy; inserting pid 81/7 in queue
[t: 54.183] pid 84/1 arrival event - queue len: 38
> 57.72 (0.646) : scheduled arrival for pid 87/1
* all bays busy; inserting pid 84/1 in queue
[t: 54.658] pid 85/2 arrival event - queue len: 39
> 58.05 (0.311) : scheduled arrival for pid 88/2
* all bays busy; inserting pid 85/2 in queue
[t: 56.274] pid 86/7 arrival event - queue len: 40
> 56.42 (0.953) : scheduled arrival for pid 89/7
* all bays busy; inserting pid 86/7 in queue
[t: 56.419] pid 89/7 arrival event - queue len: 41
> 57.52 (0.692) : scheduled arrival for pid 90/7
* all bays busy; inserting pid 89/7 in queue
[t: 56.910] pid 80/4 arrival event - queue len: 42
> 67.62 (0.279) : scheduled arrival for pid 91/4
* all bays busy; inserting pid 80/4 in queue
[t: 57.272] pid 17/4 departure event (bay 5)
[t: 57.272] handling pid 48/4 in bay 5
+ 2.751/2.751 (0.951) : [1] inspect (passed; r=0.289; p=0.120)
+ 5.211/2.460 (0.660) : [2] inspect (passed; r=0.377; p=0.120)
+ 7.051/1.840 (0.040) : [3] inspect (passed; r=0.598; p=0.120)
+ 9.423/2.372 (0.572) : [4] inspect (passed; r=0.228; p=0.120)
> 66.69 : final bay departure scheduled
[t: 57.490] pid 83/3 arrival event - queue len: 42
> 59.24 (0.615) : scheduled arrival for pid 92/3
* all bays busy; inserting pid 83/3 in queue
[t: 57.523] pid 90/7 arrival event - queue len: 43
> 60.18 (0.413) : scheduled arrival for pid 93/7
* all bays busy; inserting pid 90/7 in queue
[t: 57.720] pid 87/1 arrival event - queue len: 44
> 65.65 (0.376) : scheduled arrival for pid 94/1
* all bays busy; inserting pid 87/1 in queue
[t: 57.767] pid 36/6 arrival event - queue len: 45
> 60.19 (0.697) : scheduled arrival for pid 95/6
* all bays busy; inserting pid 36/6 in queue
[t: 58.047] pid 88/2 arrival event - queue len: 46
> 58.25 (0.933) : scheduled arrival for pid 96/2
* all bays busy; inserting pid 88/2 in queue
[t: 58.162] pid 46/2 departure event (bay 6)
[t: 58.162] handling pid 34/3 in bay 6
+ 2.079/2.079 (0.069) : [1] inspect (passed; r=0.702; p=0.260)
+ 4.080/2.001 (0.331) : [2] inspect (passed; r=0.558; p=0.260)
+ 6.101/2.021 (0.265) : [3] inspect (passed; r=0.672; p=0.260)
> 64.26 : final bay departure scheduled
[t: 58.248] pid 96/2 arrival event - queue len: 46
> 59.47 (0.655) : scheduled arrival for pid 97/2
* all bays busy; inserting pid 96/2 in queue
[t: 58.918] pid 43/7 departure event (bay 3)
[t: 58.918] handling pid 50/3 in bay 3
+ 1.912/1.912 (0.125) : [1] inspect (passed; r=0.325; p=0.210)
+ 3.832/1.920 (0.196) : [2] inspect (failed; r=0.002; p=0.210)
+ 6.843/3.011 (0.359) : [2] repair
+ 7.818/0.975 (0.497) : [2] inspect (passed; r=0.368; p=0.105)
+ 9.793/1.975 (0.752) : [3] inspect (failed; r=0.151; p=0.210)
+12.261/2.468 (0.450) : [3] repair
+13.231/0.971 (0.412) : [3] inspect (passed; r=0.954; p=0.105)
> 72.15 : final bay departure scheduled
[t: 59.239] pid 92/3 arrival event - queue len: 46
> 64.67 (0.221) : scheduled arrival for pid 98/3
* all bays busy; inserting pid 92/3 in queue
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 59.475] pid 97/2 arrival event - queue len: 47
> 62.09 (0.406) : scheduled arrival for pid 99/2
* all bays busy; inserting pid 97/2 in queue
[t: 60.043] pid 74/5 arrival event - queue len: 48
> 74.13 (0.275) : scheduled arrival for pid 100/5
* all bays busy; inserting pid 74/5 in queue
[t: 60.179] pid 93/7 arrival event - queue len: 49
> 61.57 (0.629) : scheduled arrival for pid 101/7
* all bays busy; inserting pid 93/7 in queue
[t: 60.190] pid 95/6 arrival event - queue len: 50
> 64.92 (0.493) : scheduled arrival for pid 102/6
* all bays busy; inserting pid 95/6 in queue
[t: 61.569] pid 101/7 arrival event - queue len: 51
> 67.64 (0.132) : scheduled arrival for pid 103/7
* all bays busy; inserting pid 101/7 in queue
[t: 62.087] pid 99/2 arrival event - queue len: 52
> 64.45 (0.443) : scheduled arrival for pid 104/2
* all bays busy; inserting pid 99/2 in queue
[t: 62.513] pid 41/4 departure event (bay 1)
[t: 62.513] handling pid 47/2 in bay 1
+ 2.054/2.054 (0.254) : [1] inspect (passed; r=0.461; p=0.120)
+ 4.509/2.455 (0.655) : [2] inspect (passed; r=0.617; p=0.120)
+ 6.405/1.896 (0.096) : [3] inspect (passed; r=0.844; p=0.120)
+ 8.583/2.178 (0.378) : [4] inspect (passed; r=0.252; p=0.120)
> 71.10 : final bay departure scheduled
[t: 62.531] pid 39/2 departure event (bay 4)
[t: 62.531] handling pid 44/1 in bay 4
+ 1.859/1.859 (0.804) : [1] inspect (passed; r=0.609; p=0.260)
+ 3.936/2.077 (0.077) : [2] inspect (passed; r=0.617; p=0.260)
+ 5.888/1.952 (0.492) : [3] inspect (passed; r=0.909; p=0.260)
> 68.42 : final bay departure scheduled
[t: 63.589] pid 40/4 departure event (bay 2)
[t: 63.589] handling pid 45/7 in bay 2
+ 2.421/2.421 (0.621) : [1] inspect (passed; r=0.260; p=0.120)
+ 4.246/1.826 (0.026) : [2] inspect (passed; r=0.424; p=0.120)
+ 6.646/2.399 (0.599) : [3] inspect (passed; r=0.297; p=0.120)
+ 9.292/2.646 (0.846) : [4] inspect (passed; r=0.353; p=0.120)
> 72.88 : final bay departure scheduled
[t: 64.262] pid 34/3 departure event (bay 6)
[t: 64.262] handling pid 52/2 in bay 6
+ 1.536/1.536 (0.909) : [1] inspect (passed; r=0.793; p=0.180)
+ 2.592/1.056 (0.223) : [2] inspect (failed; r=0.141; p=0.180)
+ 4.103/1.511 (0.341) : [2] repair
+ 4.785/0.683 (0.664) : [2] inspect (passed; r=0.692; p=0.090)
> 69.05 : final bay departure scheduled
[t: 64.449] pid 104/2 arrival event - queue len: 49
> 72.56 (0.061) : scheduled arrival for pid 105/2
* all bays busy; inserting pid 104/2 in queue
[t: 64.672] pid 98/3 arrival event - queue len: 50
> 65.80 (0.731) : scheduled arrival for pid 106/3
* all bays busy; inserting pid 98/3 in queue
[t: 64.923] pid 102/6 arrival event - queue len: 51
> 75.33 (0.212) : scheduled arrival for pid 107/6
* all bays busy; inserting pid 102/6 in queue
[t: 65.652] pid 94/1 arrival event - queue len: 52
> 74.20 (0.348) : scheduled arrival for pid 108/1
* all bays busy; inserting pid 94/1 in queue
[t: 65.800] pid 106/3 arrival event - queue len: 53
> 65.90 (0.974) : scheduled arrival for pid 109/3
* all bays busy; inserting pid 106/3 in queue
[t: 65.895] pid 109/3 arrival event - queue len: 54
> 67.39 (0.661) : scheduled arrival for pid 110/3
* all bays busy; inserting pid 109/3 in queue
[t: 66.695] pid 48/4 departure event (bay 5)
[t: 66.695] handling pid 55/2 in bay 5
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 2.149/2.149 (0.349) : [1] inspect (passed; r=0.877; p=0.120)
+ 4.254/2.105 (0.305) : [2] inspect (passed; r=0.405; p=0.120)
+ 6.624/2.370 (0.570) : [3] inspect (passed; r=0.208; p=0.120)
+ 8.650/2.027 (0.227) : [4] inspect (passed; r=0.944; p=0.120)
> 75.34 : final bay departure scheduled
[t: 67.386] pid 110/3 arrival event - queue len: 54
> 68.97 (0.643) : scheduled arrival for pid 111/3
* all bays busy; inserting pid 110/3 in queue
[t: 67.622] pid 91/4 arrival event - queue len: 55
> 75.84 (0.376) : scheduled arrival for pid 112/4
* all bays busy; inserting pid 91/4 in queue
[t: 67.636] pid 103/7 arrival event - queue len: 56
> 76.79 (0.047) : scheduled arrival for pid 113/7
* all bays busy; inserting pid 103/7 in queue
[t: 68.419] pid 44/1 departure event (bay 4)
[t: 68.419] handling pid 51/3 in bay 4
+ 1.524/1.524 (0.589) : [1] inspect (passed; r=0.724; p=0.300)
+ 2.985/1.461 (0.543) : [2] inspect (passed; r=0.756; p=0.300)
+ 4.949/1.964 (0.903) : [3] inspect (passed; r=0.688; p=0.300)
+ 6.106/1.157 (0.327) : [4] inspect (passed; r=0.581; p=0.300)
> 74.53 : final bay departure scheduled
[t: 68.974] pid 111/3 arrival event - queue len: 56
> 70.88 (0.590) : scheduled arrival for pid 114/3
* all bays busy; inserting pid 111/3 in queue
[t: 69.048] pid 52/2 departure event (bay 6)
[t: 69.048] handling pid 57/3 in bay 6
+ 2.027/2.027 (0.243) : [1] inspect (passed; r=0.652; p=0.260)
+ 4.081/2.054 (0.153) : [2] inspect (passed; r=0.680; p=0.260)
+ 5.983/1.901 (0.662) : [3] inspect (passed; r=0.566; p=0.260)
> 75.03 : final bay departure scheduled
[t: 70.876] pid 114/3 arrival event - queue len: 56
> 71.72 (0.790) : scheduled arrival for pid 115/3
* all bays busy; inserting pid 114/3 in queue
[t: 71.096] pid 47/2 departure event (bay 1)
[t: 71.096] handling pid 53/1 in bay 1
+ 1.844/1.844 (0.852) : [1] inspect (passed; r=0.475; p=0.260)
+ 3.823/1.979 (0.404) : [2] inspect (passed; r=0.552; p=0.260)
+ 5.774/1.951 (0.497) : [3] inspect (failed; r=0.075; p=0.260)
+ 5.951/0.177 (0.865) : [3] repair
+ 6.928/0.976 (0.490) : [3] inspect (passed; r=0.263; p=0.130)
> 78.02 : final bay departure scheduled
[t: 71.724] pid 115/3 arrival event - queue len: 56
> 80.01 (0.100) : scheduled arrival for pid 116/3
* all bays busy; inserting pid 115/3 in queue
[t: 72.150] pid 50/3 departure event (bay 3)
[t: 72.150] handling pid 54/7 in bay 3
+ 1.312/1.312 (0.588) : [1] inspect (failed; r=0.139; p=0.180)
+ 3.016/1.704 (0.305) : [1] repair
+ 3.711/0.694 (0.698) : [1] inspect (passed; r=0.732; p=0.090)
+ 4.952/1.242 (0.488) : [2] inspect (failed; r=0.038; p=0.180)
+ 6.128/1.176 (0.279) : [2] repair
+ 6.580/0.452 (0.006) : [2] inspect (passed; r=0.821; p=0.090)
> 78.73 : final bay departure scheduled
[t: 72.564] pid 105/2 arrival event - queue len: 56
> 73.90 (0.632) : scheduled arrival for pid 117/2
* all bays busy; inserting pid 105/2 in queue
[t: 72.881] pid 45/7 departure event (bay 2)
[t: 72.881] handling pid 60/7 in bay 2
+ 1.968/1.968 (0.677) : [1] inspect (failed; r=0.145; p=0.210)
+ 7.366/5.398 (0.413) : [1] repair
+ 8.359/0.993 (0.851) : [1] inspect (passed; r=0.467; p=0.105)
+10.259/1.900 (0.004) : [2] inspect (failed; r=0.029; p=0.210)
+13.832/3.573 (0.990) : [2] repair
+14.800/0.968 (0.369) : [2] inspect (passed; r=0.129; p=0.105)
+16.791/1.991 (0.911) : [3] inspect (passed; r=0.649; p=0.210)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 89.67 : final bay departure scheduled
[t: 73.895] pid 117/2 arrival event - queue len: 56
> 77.10 (0.331) : scheduled arrival for pid 118/2
* all bays busy; inserting pid 117/2 in queue
[t: 74.130] pid 100/5 arrival event - queue len: 57
> 89.09 (0.253) : scheduled arrival for pid 119/5
* all bays busy; inserting pid 100/5 in queue
[t: 74.197] pid 108/1 arrival event - queue len: 58
> 76.38 (0.764) : scheduled arrival for pid 120/1
* all bays busy; inserting pid 108/1 in queue
[t: 74.525] pid 51/3 departure event (bay 4)
[t: 74.525] handling pid 56/2 in bay 4
+ 1.576/1.576 (0.966) : [1] inspect (passed; r=0.436; p=0.180)
+ 3.084/1.508 (0.868) : [2] inspect (passed; r=0.573; p=0.180)
> 77.61 : final bay departure scheduled
[t: 75.031] pid 57/3 departure event (bay 6)
[t: 75.031] handling pid 62/2 in bay 6
+ 1.020/1.020 (0.172) : [1] inspect (passed; r=0.932; p=0.180)
+ 2.425/1.405 (0.722) : [2] inspect (passed; r=0.554; p=0.180)
> 77.46 : final bay departure scheduled
[t: 75.327] pid 107/6 arrival event - queue len: 57
> 78.44 (0.628) : scheduled arrival for pid 121/6
* all bays busy; inserting pid 107/6 in queue
[t: 75.345] pid 55/2 departure event (bay 5)
[t: 75.345] handling pid 63/2 in bay 5
+ 1.946/1.946 (0.512) : [1] inspect (passed; r=0.412; p=0.260)
+ 3.938/1.992 (0.361) : [2] inspect (failed; r=0.237; p=0.260)
+ 6.367/2.429 (0.754) : [2] repair
+ 7.407/1.040 (0.067) : [2] inspect (passed; r=0.237; p=0.130)
+ 9.316/1.909 (0.635) : [3] inspect (failed; r=0.021; p=0.260)
+13.531/4.215 (0.036) : [3] repair
+14.435/0.904 (0.972) : [3] inspect (passed; r=0.627; p=0.130)
> 89.78 : final bay departure scheduled
[t: 75.840] pid 112/4 arrival event - queue len: 57
> 106.19 (0.027) : scheduled arrival for pid 122/4
* all bays busy; inserting pid 112/4 in queue
[t: 76.382] pid 120/1 arrival event - queue len: 58
> 82.92 (0.446) : scheduled arrival for pid 123/1
* all bays busy; inserting pid 120/1 in queue
[t: 76.791] pid 113/7 arrival event - queue len: 59
> 80.17 (0.324) : scheduled arrival for pid 124/7
* all bays busy; inserting pid 113/7 in queue
[t: 77.100] pid 118/2 arrival event - queue len: 60
> 78.66 (0.583) : scheduled arrival for pid 125/2
* all bays busy; inserting pid 118/2 in queue
[t: 77.456] pid 62/2 departure event (bay 6)
[t: 77.456] handling pid 59/1 in bay 6
+ 2.056/2.056 (0.148) : [1] inspect (passed; r=0.379; p=0.260)
+ 3.964/1.909 (0.638) : [2] inspect (passed; r=0.872; p=0.260)
+ 5.988/2.024 (0.254) : [3] inspect (passed; r=0.572; p=0.260)
> 83.44 : final bay departure scheduled
[t: 77.609] pid 56/2 departure event (bay 4)
[t: 77.609] handling pid 58/3 in bay 4
+ 1.897/1.897 (0.676) : [1] inspect (passed; r=0.613; p=0.260)
+ 3.912/2.014 (0.286) : [2] inspect (passed; r=0.757; p=0.260)
+ 6.007/2.096 (0.014) : [3] inspect (passed; r=0.685; p=0.260)
> 83.62 : final bay departure scheduled
[t: 78.024] pid 53/1 departure event (bay 1)
[t: 78.024] handling pid 66/3 in bay 1
+ 1.635/1.635 (0.668) : [1] inspect (passed; r=0.703; p=0.300)
+ 2.831/1.196 (0.354) : [2] inspect (passed; r=0.760; p=0.300)
+ 4.391/1.560 (0.614) : [3] inspect (passed; r=0.490; p=0.300)
+ 5.849/1.458 (0.541) : [4] inspect (failed; r=0.238; p=0.300)
+ 7.641/1.793 (0.355) : [4] repair
+ 8.012/0.370 (0.029) : [4] inspect (passed; r=0.520; p=0.150)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 86.04 : final bay departure scheduled
[t: 78.443] pid 121/6 arrival event - queue len: 58
> 89.93 (0.180) : scheduled arrival for pid 126/6
* all bays busy; inserting pid 121/6 in queue
[t: 78.664] pid 125/2 arrival event - queue len: 59
> 80.51 (0.529) : scheduled arrival for pid 127/2
* all bays busy; inserting pid 125/2 in queue
[t: 78.730] pid 54/7 departure event (bay 3)
[t: 78.730] handling pid 61/7 in bay 3
+ 1.918/1.918 (0.184) : [1] inspect (passed; r=0.364; p=0.210)
+ 3.841/1.922 (0.222) : [2] inspect (passed; r=0.721; p=0.210)
+ 5.743/1.902 (0.020) : [3] inspect (passed; r=0.675; p=0.210)
> 84.47 : final bay departure scheduled
[t: 80.007] pid 116/3 arrival event - queue len: 59
> 80.30 (0.923) : scheduled arrival for pid 128/3
* all bays busy; inserting pid 116/3 in queue
[t: 80.173] pid 124/7 arrival event - queue len: 60
> 83.07 (0.380) : scheduled arrival for pid 129/7
* all bays busy; inserting pid 124/7 in queue
[t: 80.297] pid 128/3 arrival event - queue len: 61
> 86.90 (0.160) : scheduled arrival for pid 130/3
* all bays busy; inserting pid 128/3 in queue
[t: 80.509] pid 127/2 arrival event - queue len: 62
> 88.88 (0.056) : scheduled arrival for pid 131/2
* all bays busy; inserting pid 127/2 in queue
[t: 82.924] pid 123/1 arrival event - queue len: 63
> 83.69 (0.910) : scheduled arrival for pid 132/1
* all bays busy; inserting pid 123/1 in queue
[t: 83.073] pid 129/7 arrival event - queue len: 64
> 97.46 (0.008) : scheduled arrival for pid 133/7
* all bays busy; inserting pid 129/7 in queue
[t: 83.444] pid 59/1 departure event (bay 6)
[t: 83.444] handling pid 64/2 in bay 6
+ 1.967/1.967 (0.905) : [1] inspect (failed; r=0.151; p=0.300)
+ 2.839/0.872 (0.522) : [1] repair
+ 3.189/0.351 (0.001) : [1] inspect (passed; r=0.440; p=0.150)
+ 4.087/0.897 (0.141) : [2] inspect (passed; r=0.766; p=0.300)
+ 5.986/1.899 (0.857) : [3] inspect (failed; r=0.149; p=0.300)
+ 7.121/1.134 (0.887) : [3] repair
+ 7.929/0.809 (0.655) : [3] inspect (failed; r=0.004; p=0.150)
+ 8.417/0.488 (0.603) : [3] repair
+ 9.396/0.980 (0.899) : [3] inspect (passed; r=0.376; p=0.150)
+11.267/1.871 (0.836) : [4] inspect (passed; r=0.433; p=0.300)
> 94.71 : final bay departure scheduled
[t: 83.617] pid 58/3 departure event (bay 4)
[t: 83.617] handling pid 67/3 in bay 4
+ 1.363/1.363 (0.662) : [1] inspect (passed; r=0.810; p=0.180)
+ 2.419/1.056 (0.223) : [2] inspect (passed; r=0.240; p=0.180)
> 86.04 : final bay departure scheduled
[t: 83.686] pid 132/1 arrival event - queue len: 63
> 89.73 (0.474) : scheduled arrival for pid 134/1
* all bays busy; inserting pid 132/1 in queue
[t: 84.472] pid 61/7 departure event (bay 3)
[t: 84.472] handling pid 70/3 in bay 3
+ 1.901/1.901 (0.014) : [1] inspect (passed; r=0.432; p=0.210)
+ 3.844/1.943 (0.425) : [2] inspect (failed; r=0.035; p=0.210)
+ 8.835/4.991 (0.285) : [2] repair
+ 9.812/0.977 (0.536) : [2] inspect (passed; r=0.650; p=0.105)
+11.796/1.984 (0.841) : [3] inspect (passed; r=0.622; p=0.210)
> 96.27 : final bay departure scheduled
[t: 86.036] pid 67/3 departure event (bay 4)
[t: 86.036] handling pid 71/3 in bay 4
+ 1.314/1.314 (0.592) : [1] inspect (passed; r=0.777; p=0.180)
+ 2.906/1.592 (0.988) : [2] inspect (failed; r=0.004; p=0.180)
+ 3.845/0.938 (0.688) : [2] repair
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 4.561/0.717 (0.762) : [2] inspect (failed; r=0.019; p=0.090)
+ 5.294/0.732 (0.635) : [2] repair
+ 5.845/0.552 (0.290) : [2] inspect (passed; r=0.887; p=0.090)
> 91.88 : final bay departure scheduled
[t: 86.036] pid 66/3 departure event (bay 1)
[t: 86.036] handling pid 68/7 in bay 1
+ 1.195/1.195 (0.421) : [1] inspect (passed; r=0.265; p=0.180)
+ 2.408/1.214 (0.448) : [2] inspect (passed; r=0.627; p=0.180)
> 88.44 : final bay departure scheduled
[t: 86.902] pid 130/3 arrival event - queue len: 61
> 87.97 (0.744) : scheduled arrival for pid 135/3
* all bays busy; inserting pid 130/3 in queue
[t: 87.967] pid 135/3 arrival event - queue len: 62
> 99.54 (0.040) : scheduled arrival for pid 136/3
* all bays busy; inserting pid 135/3 in queue
[t: 88.444] pid 68/7 departure event (bay 1)
[t: 88.444] handling pid 23/5 in bay 1
+ 1.936/1.936 (0.357) : [1] inspect (passed; r=0.960; p=0.210)
+ 3.862/1.927 (0.267) : [2] inspect (failed; r=0.064; p=0.210)
+ 6.774/2.911 (0.191) : [2] repair
+ 7.726/0.953 (0.056) : [2] inspect (passed; r=0.224; p=0.105)
+ 9.638/1.911 (0.114) : [3] inspect (passed; r=0.857; p=0.210)
> 98.08 : final bay departure scheduled
[t: 88.882] pid 131/2 arrival event - queue len: 62
> 91.40 (0.419) : scheduled arrival for pid 137/2
* all bays busy; inserting pid 131/2 in queue
[t: 89.093] pid 119/5 arrival event - queue len: 63
> 100.67 (0.346) : scheduled arrival for pid 138/5
* all bays busy; inserting pid 119/5 in queue
[t: 89.672] pid 60/7 departure event (bay 2)
[t: 89.672] handling pid 69/2 in bay 2
+ 1.969/1.969 (0.688) : [1] inspect (failed; r=0.140; p=0.210)
+ 3.156/1.188 (0.445) : [1] repair
+ 4.146/0.989 (0.790) : [1] inspect (passed; r=0.927; p=0.105)
+ 6.069/1.923 (0.234) : [2] inspect (passed; r=0.950; p=0.210)
+ 7.997/1.928 (0.282) : [3] inspect (passed; r=0.465; p=0.210)
> 97.67 : final bay departure scheduled
[t: 89.729] pid 134/1 arrival event - queue len: 63
> 95.77 (0.475) : scheduled arrival for pid 139/1
* all bays busy; inserting pid 134/1 in queue
[t: 89.780] pid 63/2 departure event (bay 5)
[t: 89.780] handling pid 72/3 in bay 5
+ 1.987/1.987 (0.377) : [1] inspect (failed; r=0.107; p=0.260)
+ 5.093/3.106 (0.092) : [1] repair
+ 6.103/1.010 (0.265) : [1] inspect (failed; r=0.009; p=0.130)
+ 8.663/2.560 (0.352) : [1] repair
+ 9.679/1.016 (0.228) : [1] inspect (failed; r=0.014; p=0.130)
+10.850/1.171 (0.634) : [1] repair
+11.849/0.999 (0.339) : [1] inspect (passed; r=0.870; p=0.130)
+13.753/1.904 (0.652) : [2] inspect (passed; r=0.949; p=0.260)
+15.767/2.014 (0.286) : [3] inspect (failed; r=0.247; p=0.260)
+19.294/3.527 (0.666) : [3] repair
+20.335/1.041 (0.063) : [3] inspect (passed; r=0.256; p=0.130)
> 110.12 : final bay departure scheduled
[t: 89.928] pid 126/6 arrival event - queue len: 63
> 91.25 (0.821) : scheduled arrival for pid 140/6
* all bays busy; inserting pid 126/6 in queue
[t: 91.251] pid 140/6 arrival event - queue len: 64
> 110.62 (0.056) : scheduled arrival for pid 141/6
* all bays busy; inserting pid 140/6 in queue
[t: 91.403] pid 137/2 arrival event - queue len: 65
> 92.65 (0.651) : scheduled arrival for pid 142/2
* all bays busy; inserting pid 137/2 in queue
[t: 91.881] pid 71/3 departure event (bay 4)
[t: 91.881] handling pid 49/4 in bay 4
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 1.498/1.498 (0.854) : [1] inspect (passed; r=0.738; p=0.180)
+ 2.972/1.474 (0.819) : [2] inspect (passed; r=0.510; p=0.180)
> 94.85 : final bay departure scheduled
[t: 92.648] pid 142/2 arrival event - queue len: 65
> 93.80 (0.672) : scheduled arrival for pid 143/2
* all bays busy; inserting pid 142/2 in queue
[t: 93.800] pid 143/2 arrival event - queue len: 66
> 96.42 (0.406) : scheduled arrival for pid 144/2
* all bays busy; inserting pid 143/2 in queue
[t: 94.711] pid 64/2 departure event (bay 6)
[t: 94.711] handling pid 76/3 in bay 6
+ 2.027/2.027 (0.242) : [1] inspect (failed; r=0.157; p=0.260)
+ 3.318/1.291 (0.505) : [1] repair
+ 4.356/1.038 (0.077) : [1] inspect (passed; r=0.940; p=0.130)
+ 6.416/2.060 (0.134) : [2] inspect (passed; r=0.702; p=0.260)
+ 8.354/1.938 (0.541) : [3] inspect (passed; r=0.738; p=0.260)
> 103.06 : final bay departure scheduled
[t: 94.853] pid 49/4 departure event (bay 4)
[t: 94.853] handling pid 78/3 in bay 4
+ 2.353/2.353 (0.553) : [1] inspect (passed; r=0.402; p=0.120)
+ 4.207/1.854 (0.054) : [2] inspect (failed; r=0.050; p=0.120)
+ 7.440/3.233 (0.207) : [2] repair
+ 8.665/1.225 (0.650) : [2] inspect (passed; r=0.971; p=0.060)
+11.297/2.632 (0.832) : [3] inspect (passed; r=0.967; p=0.120)
+14.040/2.743 (0.943) : [4] inspect (passed; r=0.275; p=0.120)
> 108.89 : final bay departure scheduled
[t: 95.766] pid 139/1 arrival event - queue len: 65
> 97.06 (0.852) : scheduled arrival for pid 145/1
* all bays busy; inserting pid 139/1 in queue
[t: 96.269] pid 70/3 departure event (bay 3)
[t: 96.269] handling pid 77/4 in bay 3
+ 1.519/1.519 (0.884) : [1] inspect (passed; r=0.446; p=0.180)
+ 2.454/0.935 (0.050) : [2] inspect (passed; r=0.462; p=0.180)
> 98.72 : final bay departure scheduled
[t: 96.417] pid 144/2 arrival event - queue len: 65
> 100.47 (0.248) : scheduled arrival for pid 146/2
* all bays busy; inserting pid 144/2 in queue
[t: 97.062] pid 145/1 arrival event - queue len: 66
> 98.83 (0.804) : scheduled arrival for pid 147/1
* all bays busy; inserting pid 145/1 in queue
[t: 97.462] pid 133/7 arrival event - queue len: 67
> 109.64 (0.017) : scheduled arrival for pid 148/7
* all bays busy; inserting pid 133/7 in queue
[t: 97.669] pid 69/2 departure event (bay 2)
[t: 97.669] handling pid 73/7 in bay 2
+ 1.924/1.924 (0.588) : [1] inspect (passed; r=0.934; p=0.260)
+ 3.818/1.895 (0.685) : [2] inspect (passed; r=0.426; p=0.260)
+ 5.827/2.008 (0.306) : [3] inspect (passed; r=0.821; p=0.260)
> 103.50 : final bay departure scheduled
[t: 98.082] pid 23/5 departure event (bay 1)
[t: 98.082] handling pid 79/3 in bay 1
+ 1.169/1.169 (0.263) : [1] inspect (failed; r=0.319; p=0.360)
+ 2.096/0.927 (0.874) : [1] repair
+ 2.681/0.585 (0.265) : [1] inspect (failed; r=0.052; p=0.180)
+ 3.399/0.718 (0.437) : [1] repair
+ 3.957/0.558 (0.225) : [1] inspect (passed; r=0.239; p=0.180)
+ 5.415/1.459 (0.470) : [2] inspect (failed; r=0.093; p=0.360)
+ 8.119/2.704 (0.345) : [2] repair
+ 8.521/0.402 (0.002) : [2] inspect (passed; r=0.304; p=0.180)
+10.256/1.735 (0.668) : [3] inspect (failed; r=0.223; p=0.360)
+12.796/2.540 (0.233) : [3] repair
+13.640/0.844 (0.634) : [3] inspect (failed; r=0.157; p=0.180)
+15.169/1.528 (0.074) : [3] repair
+15.877/0.708 (0.440) : [3] inspect (passed; r=0.251; p=0.180)
+17.667/1.790 (0.707) : [4] inspect (failed; r=0.287; p=0.360)
```

## APPENDIX C: The Application Trace File (ATF)

```
+21.599/3.932 (0.327) : [4] repair
+22.664/1.065 (0.950) : [4] inspect (passed; r=0.922; p=0.180)
> 120.75 : final bay departure scheduled
[t: 98.723] pid 77/4 departure event (bay 3)
[t: 98.723] handling pid 82/3 in bay 3
+ 2.625/2.625 (0.825) : [1] inspect (passed; r=0.260; p=0.120)
+ 5.077/2.452 (0.652) : [2] inspect (failed; r=0.061; p=0.120)
+ 8.805/3.727 (0.274) : [2] repair
+ 9.994/1.189 (0.579) : [2] inspect (passed; r=0.722; p=0.060)
+12.078/2.085 (0.285) : [3] inspect (passed; r=0.657; p=0.120)
+14.522/2.443 (0.643) : [4] inspect (failed; r=0.087; p=0.120)
+14.961/0.439 (0.937) : [4] repair
+16.293/1.332 (0.863) : [4] inspect (passed; r=0.187; p=0.060)
> 115.02 : final bay departure scheduled
[t: 98.827] pid 147/1 arrival event - queue len: 65
> 101.45 (0.723) : scheduled arrival for pid 149/1
* all bays busy; inserting pid 147/1 in queue
[t: 99.539] pid 136/3 arrival event - queue len: 66
> 101.07 (0.654) : scheduled arrival for pid 150/3
* all bays busy; inserting pid 136/3 in queue
[t: 100.465] pid 146/2 arrival event - queue len: 67
> 104.41 (0.256) : scheduled arrival for pid 151/2
* all bays busy; inserting pid 146/2 in queue
[t: 100.669] pid 138/5 arrival event - queue len: 68
> 101.35 (0.940) : scheduled arrival for pid 152/5
* all bays busy; inserting pid 138/5 in queue
[t: 101.068] pid 150/3 arrival event - queue len: 69
> 102.03 (0.766) : scheduled arrival for pid 153/3
* all bays busy; inserting pid 150/3 in queue
[t: 101.346] pid 152/5 arrival event - queue len: 70
> 108.30 (0.528) : scheduled arrival for pid 154/5
* all bays busy; inserting pid 152/5 in queue
[t: 101.454] pid 149/1 arrival event - queue len: 71
> 104.35 (0.700) : scheduled arrival for pid 155/1
* all bays busy; inserting pid 149/1 in queue
[t: 102.026] pid 153/3 arrival event - queue len: 72
> 103.24 (0.714) : scheduled arrival for pid 156/3
* all bays busy; inserting pid 153/3 in queue
[t: 103.065] pid 76/3 departure event (bay 6)
[t: 103.065] handling pid 65/1 in bay 6
+ 0.966/0.966 (0.094) : [1] inspect (passed; r=0.847; p=0.180)
+ 2.253/1.287 (0.552) : [2] inspect (passed; r=0.266; p=0.180)
> 105.32 : final bay departure scheduled
[t: 103.237] pid 156/3 arrival event - queue len: 72
> 104.52 (0.699) : scheduled arrival for pid 157/3
* all bays busy; inserting pid 156/3 in queue
[t: 103.496] pid 73/7 departure event (bay 2)
[t: 103.496] handling pid 75/2 in bay 2
+ 1.938/1.938 (0.383) : [1] inspect (passed; r=0.960; p=0.210)
+ 3.897/1.959 (0.592) : [2] inspect (passed; r=0.497; p=0.210)
+ 5.799/1.901 (0.011) : [3] inspect (passed; r=0.742; p=0.210)
> 109.29 : final bay departure scheduled
[t: 104.347] pid 155/1 arrival event - queue len: 72
> 114.63 (0.281) : scheduled arrival for pid 158/1
* all bays busy; inserting pid 155/1 in queue
[t: 104.414] pid 151/2 arrival event - queue len: 73
> 104.78 (0.880) : scheduled arrival for pid 159/2
* all bays busy; inserting pid 151/2 in queue
[t: 104.524] pid 157/3 arrival event - queue len: 74
> 106.88 (0.520) : scheduled arrival for pid 160/3
* all bays busy; inserting pid 157/3 in queue
[t: 104.785] pid 159/2 arrival event - queue len: 75
> 105.90 (0.682) : scheduled arrival for pid 161/2
* all bays busy; inserting pid 159/2 in queue
[t: 105.317] pid 65/1 departure event (bay 6)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 105.317] handling pid 81/7 in bay 6
+ 1.082/1.082 (0.273) : [1] inspect (passed; r=0.696; p=0.300)
+ 3.144/2.062 (0.973) : [2] inspect (passed; r=0.831; p=0.300)
+ 4.471/1.327 (0.448) : [3] inspect (passed; r=0.463; p=0.300)
+ 6.310/1.839 (0.813) : [4] inspect (passed; r=0.466; p=0.300)
> 111.63 : final bay departure scheduled
[t: 105.895] pid 161/2 arrival event - queue len: 75
> 107.30 (0.616) : scheduled arrival for pid 162/2
* all bays busy; inserting pid 161/2 in queue
[t: 106.189] pid 122/4 arrival event - queue len: 76
> 121.64 (0.159) : scheduled arrival for pid 163/4
* all bays busy; inserting pid 122/4 in queue
[t: 106.877] pid 160/3 arrival event - queue len: 77
> 110.65 (0.350) : scheduled arrival for pid 164/3
* all bays busy; inserting pid 160/3 in queue
[t: 107.300] pid 162/2 arrival event - queue len: 78
> 109.39 (0.487) : scheduled arrival for pid 165/2
* all bays busy; inserting pid 162/2 in queue
[t: 108.299] pid 154/5 arrival event - queue len: 79
> 110.01 (0.855) : scheduled arrival for pid 166/5
* all bays busy; inserting pid 154/5 in queue
[t: 108.892] pid 78/3 departure event (bay 4)
[t: 108.892] handling pid 84/1 in bay 4
+ 1.250/1.250 (0.499) : [1] inspect (passed; r=0.319; p=0.180)
+ 2.215/0.965 (0.093) : [2] inspect (passed; r=0.663; p=0.180)
> 111.11 : final bay departure scheduled
[t: 109.294] pid 75/2 departure event (bay 2)
[t: 109.294] handling pid 85/2 in bay 2
+ 1.859/1.859 (0.802) : [1] inspect (passed; r=0.713; p=0.260)
+ 3.769/1.910 (0.634) : [2] inspect (passed; r=0.309; p=0.260)
+ 5.619/1.850 (0.834) : [3] inspect (failed; r=0.148; p=0.260)
+ 8.343/2.724 (0.174) : [3] repair
+ 9.270/0.927 (0.818) : [3] inspect (passed; r=0.132; p=0.130)
> 118.56 : final bay departure scheduled
[t: 109.388] pid 165/2 arrival event - queue len: 78
> 110.76 (0.624) : scheduled arrival for pid 167/2
* all bays busy; inserting pid 165/2 in queue
[t: 109.637] pid 148/7 arrival event - queue len: 79
> 110.02 (0.879) : scheduled arrival for pid 168/7
* all bays busy; inserting pid 148/7 in queue
[t: 110.006] pid 166/5 arrival event - queue len: 80
> 115.57 (0.600) : scheduled arrival for pid 169/5
* all bays busy; inserting pid 166/5 in queue
[t: 110.025] pid 168/7 arrival event - queue len: 81
> 110.55 (0.839) : scheduled arrival for pid 170/7
* all bays busy; inserting pid 168/7 in queue
[t: 110.115] pid 72/3 departure event (bay 5)
[t: 110.115] handling pid 86/7 in bay 5
+ 0.980/0.980 (0.114) : [1] inspect (passed; r=0.476; p=0.180)
+ 2.332/1.352 (0.646) : [2] inspect (passed; r=0.875; p=0.180)
> 112.45 : final bay departure scheduled
[t: 110.550] pid 170/7 arrival event - queue len: 81
> 112.70 (0.488) : scheduled arrival for pid 171/7
* all bays busy; inserting pid 170/7 in queue
[t: 110.622] pid 141/6 arrival event - queue len: 82
> 112.30 (0.779) : scheduled arrival for pid 172/6
* all bays busy; inserting pid 141/6 in queue
[t: 110.651] pid 164/3 arrival event - queue len: 83
> 111.04 (0.898) : scheduled arrival for pid 173/3
* all bays busy; inserting pid 164/3 in queue
[t: 110.757] pid 167/2 arrival event - queue len: 84
> 112.66 (0.519) : scheduled arrival for pid 174/2
* all bays busy; inserting pid 167/2 in queue
[t: 111.039] pid 173/3 arrival event - queue len: 85
> 117.67 (0.158) : scheduled arrival for pid 175/3
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 173/3 in queue
[t: 111.107] pid 84/1 departure event (bay 4)
[t: 111.107] handling pid 89/7 in bay 4
+ 1.105/1.105 (0.289) : [1] inspect (passed; r=0.977; p=0.300)
+ 2.219/1.115 (0.296) : [2] inspect (passed; r=0.475; p=0.300)
+ 3.435/1.216 (0.369) : [3] inspect (passed; r=0.619; p=0.300)
+ 4.189/0.754 (0.038) : [4] inspect (passed; r=0.715; p=0.300)
> 115.30 : final bay departure scheduled
[t: 111.627] pid 81/7 departure event (bay 6)
[t: 111.627] handling pid 80/4 in bay 6
+ 1.931/1.931 (0.309) : [1] inspect (passed; r=0.562; p=0.210)
+ 3.909/1.978 (0.778) : [2] inspect (passed; r=0.747; p=0.210)
+ 5.888/1.979 (0.789) : [3] inspect (passed; r=0.512; p=0.210)
> 117.51 : final bay departure scheduled
[t: 112.297] pid 172/6 arrival event - queue len: 84
> 116.74 (0.515) : scheduled arrival for pid 176/6
* all bays busy; inserting pid 172/6 in queue
[t: 112.448] pid 86/7 departure event (bay 5)
[t: 112.448] handling pid 83/3 in bay 5
+ 1.920/1.920 (0.199) : [1] inspect (failed; r=0.154; p=0.210)
+ 5.423/3.503 (0.702) : [1] repair
+ 6.397/0.974 (0.481) : [1] inspect (passed; r=0.487; p=0.105)
+ 8.328/1.931 (0.308) : [2] inspect (passed; r=0.722; p=0.210)
+10.281/1.954 (0.537) : [3] inspect (passed; r=0.657; p=0.210)
> 122.73 : final bay departure scheduled
[t: 112.660] pid 174/2 arrival event - queue len: 84
> 113.16 (0.843) : scheduled arrival for pid 177/2
* all bays busy; inserting pid 174/2 in queue
[t: 112.702] pid 171/7 arrival event - queue len: 85
> 114.57 (0.536) : scheduled arrival for pid 178/7
* all bays busy; inserting pid 171/7 in queue
[t: 113.157] pid 177/2 arrival event - queue len: 86
> 115.51 (0.445) : scheduled arrival for pid 179/2
* all bays busy; inserting pid 177/2 in queue
[t: 114.572] pid 178/7 arrival event - queue len: 87
> 114.83 (0.918) : scheduled arrival for pid 180/7
* all bays busy; inserting pid 178/7 in queue
[t: 114.634] pid 158/1 arrival event - queue len: 88
> 130.34 (0.144) : scheduled arrival for pid 181/1
* all bays busy; inserting pid 158/1 in queue
[t: 114.829] pid 180/7 arrival event - queue len: 89
> 119.90 (0.184) : scheduled arrival for pid 182/7
* all bays busy; inserting pid 180/7 in queue
[t: 115.016] pid 82/3 departure event (bay 3)
[t: 115.016] handling pid 90/7 in bay 3
+ 1.040/1.040 (0.200) : [1] inspect (passed; r=0.917; p=0.180)
+ 2.393/1.353 (0.648) : [2] inspect (failed; r=0.008; p=0.180)
+ 3.218/0.824 (0.696) : [2] repair
+ 3.698/0.480 (0.087) : [2] inspect (passed; r=0.976; p=0.090)
> 118.71 : final bay departure scheduled
[t: 115.296] pid 89/7 departure event (bay 4)
[t: 115.296] handling pid 87/1 in bay 4
+ 1.941/1.941 (0.410) : [1] inspect (passed; r=0.515; p=0.210)
+ 3.865/1.924 (0.236) : [2] inspect (passed; r=0.766; p=0.210)
+ 5.849/1.984 (0.840) : [3] inspect (passed; r=0.878; p=0.210)
> 121.14 : final bay departure scheduled
[t: 115.508] pid 179/2 arrival event - queue len: 88
> 116.26 (0.772) : scheduled arrival for pid 183/2
* all bays busy; inserting pid 179/2 in queue
[t: 115.572] pid 169/5 arrival event - queue len: 89
> 132.49 (0.212) : scheduled arrival for pid 184/5
* all bays busy; inserting pid 169/5 in queue
[t: 116.258] pid 183/2 arrival event - queue len: 90
> 119.09 (0.377) : scheduled arrival for pid 185/2
* all bays busy; inserting pid 183/2 in queue
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 116.739] pid 176/6 arrival event - queue len: 91
> 119.49 (0.663) : scheduled arrival for pid 186/6
* all bays busy; inserting pid 176/6 in queue
[t: 117.515] pid 80/4 departure event (bay 6)
[t: 117.515] handling pid 36/6 in bay 6
+ 2.628/2.628 (0.828) : [1] inspect (passed; r=0.744; p=0.120)
+ 5.183/2.555 (0.755) : [2] inspect (passed; r=0.353; p=0.120)
+ 7.685/2.503 (0.703) : [3] inspect (passed; r=0.543; p=0.120)
+ 9.549/1.864 (0.064) : [4] inspect (passed; r=0.249; p=0.120)
> 127.06 : final bay departure scheduled
[t: 117.671] pid 175/3 arrival event - queue len: 91
> 125.31 (0.120) : scheduled arrival for pid 187/3
* all bays busy; inserting pid 175/3 in queue
[t: 118.565] pid 85/2 departure event (bay 2)
[t: 118.565] handling pid 88/2 in bay 2
+ 2.067/2.067 (0.109) : [1] inspect (passed; r=0.977; p=0.260)
+ 3.965/1.898 (0.674) : [2] inspect (passed; r=0.928; p=0.260)
+ 6.034/2.069 (0.104) : [3] inspect (passed; r=0.824; p=0.260)
> 124.60 : final bay departure scheduled
[t: 118.714] pid 90/7 departure event (bay 3)
[t: 118.714] handling pid 96/2 in bay 3
+ 1.974/1.974 (0.743) : [1] inspect (passed; r=0.684; p=0.210)
+ 3.908/1.933 (0.333) : [2] inspect (passed; r=0.461; p=0.210)
+ 5.865/1.958 (0.575) : [3] inspect (passed; r=0.659; p=0.210)
> 124.58 : final bay departure scheduled
[t: 119.090] pid 185/2 arrival event - queue len: 90
> 120.57 (0.600) : scheduled arrival for pid 188/2
* all bays busy; inserting pid 185/2 in queue
[t: 119.492] pid 186/6 arrival event - queue len: 91
> 123.11 (0.583) : scheduled arrival for pid 189/6
* all bays busy; inserting pid 186/6 in queue
[t: 119.903] pid 182/7 arrival event - queue len: 92
> 123.20 (0.333) : scheduled arrival for pid 190/7
* all bays busy; inserting pid 182/7 in queue
[t: 120.571] pid 188/2 arrival event - queue len: 93
> 120.77 (0.934) : scheduled arrival for pid 191/2
* all bays busy; inserting pid 188/2 in queue
[t: 120.746] pid 79/3 departure event (bay 1)
[t: 120.746] handling pid 92/3 in bay 1
+ 1.530/1.530 (0.900) : [1] inspect (failed; r=0.046; p=0.180)
+ 2.769/1.239 (0.435) : [1] repair
+ 3.312/0.542 (0.264) : [1] inspect (passed; r=0.523; p=0.090)
+ 4.364/1.052 (0.217) : [2] inspect (failed; r=0.085; p=0.180)
+ 5.727/1.363 (0.253) : [2] repair
+ 6.219/0.492 (0.120) : [2] inspect (passed; r=0.280; p=0.090)
> 126.97 : final bay departure scheduled
[t: 120.770] pid 191/2 arrival event - queue len: 93
> 124.48 (0.278) : scheduled arrival for pid 192/2
* all bays busy; inserting pid 191/2 in queue
[t: 121.145] pid 87/1 departure event (bay 4)
[t: 121.145] handling pid 97/2 in bay 4
+ 1.124/1.124 (0.303) : [1] inspect (passed; r=0.601; p=0.300)
+ 1.890/0.766 (0.047) : [2] inspect (passed; r=0.527; p=0.300)
+ 3.442/1.552 (0.608) : [3] inspect (failed; r=0.276; p=0.300)
+ 3.875/0.433 (0.787) : [3] repair
+ 4.625/0.751 (0.572) : [3] inspect (passed; r=0.404; p=0.150)
+ 6.696/2.071 (0.979) : [4] inspect (passed; r=0.932; p=0.300)
> 127.84 : final bay departure scheduled
[t: 121.645] pid 163/4 arrival event - queue len: 93
> 137.48 (0.152) : scheduled arrival for pid 193/4
* all bays busy; inserting pid 163/4 in queue
[t: 122.729] pid 83/3 departure event (bay 5)
[t: 122.729] handling pid 74/5 in bay 5
+ 0.950/0.950 (0.071) : [1] inspect (passed; r=0.331; p=0.180)
+ 1.977/1.027 (0.182) : [2] inspect (passed; r=0.756; p=0.180)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 124.71 : final bay departure scheduled
[t: 123.105] pid 189/6 arrival event - queue len: 93
> 124.85 (0.770) : scheduled arrival for pid 194/6
* all bays busy; inserting pid 189/6 in queue
[t: 123.204] pid 190/7 arrival event - queue len: 94
> 124.24 (0.709) : scheduled arrival for pid 195/7
* all bays busy; inserting pid 190/7 in queue
[t: 124.236] pid 195/7 arrival event - queue len: 95
> 126.52 (0.467) : scheduled arrival for pid 196/7
* all bays busy; inserting pid 195/7 in queue
[t: 124.479] pid 192/2 arrival event - queue len: 96
> 126.61 (0.479) : scheduled arrival for pid 197/2
* all bays busy; inserting pid 192/2 in queue
[t: 124.579] pid 96/2 departure event (bay 3)
[t: 124.579] handling pid 93/7 in bay 3
+ 1.856/1.856 (0.815) : [1] inspect (passed; r=0.361; p=0.260)
+ 3.847/1.991 (0.363) : [2] inspect (passed; r=0.620; p=0.260)
+ 5.932/2.085 (0.050) : [3] inspect (passed; r=0.502; p=0.260)
> 130.51 : final bay departure scheduled
[t: 124.599] pid 88/2 departure event (bay 2)
[t: 124.599] handling pid 95/6 in bay 2
+ 1.872/1.872 (0.761) : [1] inspect (passed; r=0.948; p=0.260)
+ 3.726/1.854 (0.818) : [2] inspect (passed; r=0.741; p=0.260)
+ 5.713/1.987 (0.376) : [3] inspect (failed; r=0.074; p=0.260)
+ 6.633/0.920 (0.492) : [3] repair
+ 7.584/0.951 (0.659) : [3] inspect (passed; r=0.571; p=0.130)
> 132.18 : final bay departure scheduled
[t: 124.706] pid 74/5 departure event (bay 5)
[t: 124.706] handling pid 101/7 in bay 5
+ 1.767/1.767 (0.691) : [1] inspect (passed; r=0.927; p=0.360)
+ 3.763/1.996 (0.854) : [2] inspect (passed; r=0.725; p=0.360)
+ 5.899/2.136 (0.955) : [3] inspect (failed; r=0.271; p=0.360)
+ 6.993/1.094 (0.714) : [3] repair
+ 7.564/0.571 (0.244) : [3] inspect (passed; r=0.933; p=0.180)
+ 8.975/1.411 (0.436) : [4] inspect (failed; r=0.046; p=0.360)
+ 9.901/0.925 (0.792) : [4] repair
+10.715/0.815 (0.593) : [4] inspect (passed; r=0.915; p=0.180)
> 135.42 : final bay departure scheduled
[t: 124.855] pid 194/6 arrival event - queue len: 94
> 125.90 (0.856) : scheduled arrival for pid 198/6
* all bays busy; inserting pid 194/6 in queue
[t: 125.305] pid 187/3 arrival event - queue len: 95
> 128.93 (0.365) : scheduled arrival for pid 199/3
* all bays busy; inserting pid 187/3 in queue
[t: 125.898] pid 198/6 arrival event - queue len: 96
> 132.87 (0.353) : scheduled arrival for pid 200/6
* all bays busy; inserting pid 198/6 in queue
[t: 126.523] pid 196/7 arrival event - queue len: 97
> 127.12 (0.820) : scheduled arrival for pid 201/7
* all bays busy; inserting pid 196/7 in queue
[t: 126.615] pid 197/2 arrival event - queue len: 98
> 133.61 (0.090) : scheduled arrival for pid 202/2
* all bays busy; inserting pid 197/2 in queue
[t: 126.965] pid 92/3 departure event (bay 1)
[t: 126.965] handling pid 99/2 in bay 1
+ 0.973/0.973 (0.104) : [1] inspect (passed; r=0.562; p=0.180)
+ 1.984/1.012 (0.160) : [2] inspect (failed; r=0.140; p=0.180)
+ 2.513/0.528 (0.883) : [2] repair
+ 3.085/0.572 (0.349) : [2] inspect (passed; r=0.475; p=0.090)
> 130.05 : final bay departure scheduled
[t: 127.064] pid 36/6 departure event (bay 6)
[t: 127.064] handling pid 104/2 in bay 6
+ 1.605/1.605 (0.046) : [1] inspect (passed; r=0.921; p=0.140)
+ 3.261/1.657 (0.565) : [2] inspect (passed; r=0.311; p=0.140)
> 130.32 : final bay departure scheduled
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 127.118] pid 201/7 arrival event - queue len: 97
> 134.74 (0.079) : scheduled arrival for pid 203/7
* all bays busy; inserting pid 201/7 in queue
[t: 127.841] pid 97/2 departure event (bay 4)
[t: 127.841] handling pid 98/3 in bay 4
+ 1.801/1.801 (0.996) : [1] inspect (passed; r=0.853; p=0.260)
+ 3.824/2.022 (0.259) : [2] inspect (passed; r=0.784; p=0.260)
+ 5.778/1.954 (0.485) : [3] inspect (failed; r=0.160; p=0.260)
+ 6.869/1.091 (0.698) : [3] repair
+ 7.843/0.974 (0.503) : [3] inspect (passed; r=0.766; p=0.130)
> 135.68 : final bay departure scheduled
[t: 128.930] pid 199/3 arrival event - queue len: 97
> 134.60 (0.207) : scheduled arrival for pid 204/3
* all bays busy; inserting pid 199/3 in queue
[t: 130.050] pid 99/2 departure event (bay 1)
[t: 130.050] handling pid 102/6 in bay 1
+ 1.804/1.804 (0.987) : [1] inspect (passed; r=0.562; p=0.260)
+ 3.774/1.970 (0.433) : [2] inspect (passed; r=0.302; p=0.260)
+ 5.812/2.038 (0.208) : [3] inspect (passed; r=0.604; p=0.260)
> 135.86 : final bay departure scheduled
[t: 130.325] pid 104/2 departure event (bay 6)
[t: 130.325] handling pid 94/1 in bay 6
+ 1.989/1.989 (0.369) : [1] inspect (passed; r=0.845; p=0.260)
+ 4.050/2.061 (0.129) : [2] inspect (passed; r=0.731; p=0.260)
+ 6.098/2.047 (0.175) : [3] inspect (passed; r=0.885; p=0.260)
> 136.42 : final bay departure scheduled
[t: 130.344] pid 181/1 arrival event - queue len: 96
> 132.61 (0.756) : scheduled arrival for pid 205/1
* all bays busy; inserting pid 181/1 in queue
[t: 130.510] pid 93/7 departure event (bay 3)
[t: 130.510] handling pid 106/3 in bay 3
+ 1.978/1.978 (0.781) : [1] inspect (passed; r=0.849; p=0.210)
+ 3.934/1.956 (0.560) : [2] inspect (passed; r=0.996; p=0.210)
+ 5.887/1.953 (0.525) : [3] inspect (passed; r=0.384; p=0.210)
> 136.40 : final bay departure scheduled
[t: 132.183] pid 95/6 departure event (bay 2)
[t: 132.183] handling pid 109/3 in bay 2
+ 1.642/1.642 (0.415) : [1] inspect (passed; r=0.592; p=0.140)
+ 3.326/1.684 (0.843) : [2] inspect (passed; r=0.638; p=0.140)
> 135.51 : final bay departure scheduled
[t: 132.495] pid 184/5 arrival event - queue len: 95
> 139.31 (0.535) : scheduled arrival for pid 206/5
* all bays busy; inserting pid 184/5 in queue
[t: 132.608] pid 205/1 arrival event - queue len: 96
> 145.86 (0.195) : scheduled arrival for pid 207/1
* all bays busy; inserting pid 205/1 in queue
[t: 132.872] pid 200/6 arrival event - queue len: 97
> 133.24 (0.946) : scheduled arrival for pid 208/6
* all bays busy; inserting pid 200/6 in queue
[t: 133.244] pid 208/6 arrival event - queue len: 98
> 135.17 (0.750) : scheduled arrival for pid 209/6
* all bays busy; inserting pid 208/6 in queue
[t: 133.613] pid 202/2 arrival event - queue len: 99
> 140.22 (0.103) : scheduled arrival for pid 210/2
* all bays busy; inserting pid 202/2 in queue
[t: 134.598] pid 204/3 arrival event - queue len: 100
> 142.38 (0.115) : scheduled arrival for pid 211/3
* all bays busy; inserting pid 204/3 in queue
[t: 134.741] pid 203/7 arrival event - queue len: 101
> 135.70 (0.725) : scheduled arrival for pid 212/7
* all bays busy; inserting pid 203/7 in queue
[t: 135.173] pid 209/6 arrival event - queue len: 102
> 135.45 (0.960) : scheduled arrival for pid 213/6
* all bays busy; inserting pid 209/6 in queue
[t: 135.421] pid 101/7 departure event (bay 5)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 135.421] handling pid 110/3 in bay 5
+ 1.935/1.935 (0.354) : [1] inspect (passed; r=0.677; p=0.210)
+ 3.929/1.994 (0.941) : [2] inspect (passed; r=0.937; p=0.210)
+ 5.885/1.955 (0.554) : [3] inspect (passed; r=0.478; p=0.210)
> 141.31 : final bay departure scheduled
[t: 135.446] pid 213/6 arrival event - queue len: 102
> 144.92 (0.243) : scheduled arrival for pid 214/6
* all bays busy; inserting pid 213/6 in queue
[t: 135.509] pid 109/3 departure event (bay 2)
[t: 135.509] handling pid 91/4 in bay 2
+ 1.385/1.385 (0.693) : [1] inspect (failed; r=0.061; p=0.180)
+ 7.053/5.668 (0.067) : [1] repair
+ 7.718/0.665 (0.613) : [1] inspect (passed; r=0.689; p=0.090)
+ 8.676/0.958 (0.083) : [2] inspect (passed; r=0.596; p=0.180)
> 144.18 : final bay departure scheduled
[t: 135.684] pid 98/3 departure event (bay 4)
[t: 135.684] handling pid 103/7 in bay 4
+ 1.227/1.227 (0.467) : [1] inspect (passed; r=0.653; p=0.180)
+ 2.398/1.171 (0.388) : [2] inspect (passed; r=0.567; p=0.180)
> 138.08 : final bay departure scheduled
[t: 135.705] pid 212/7 arrival event - queue len: 101
> 136.15 (0.863) : scheduled arrival for pid 215/7
* all bays busy; inserting pid 212/7 in queue
[t: 135.861] pid 102/6 departure event (bay 1)
[t: 135.861] handling pid 111/3 in bay 1
+ 1.637/1.637 (0.371) : [1] inspect (failed; r=0.088; p=0.140)
+ 4.425/2.788 (0.963) : [1] repair
+ 5.239/0.815 (0.291) : [1] inspect (passed; r=0.681; p=0.070)
+ 6.862/1.622 (0.223) : [2] inspect (passed; r=0.735; p=0.140)
> 142.72 : final bay departure scheduled
[t: 136.145] pid 215/7 arrival event - queue len: 101
> 142.20 (0.133) : scheduled arrival for pid 216/7
* all bays busy; inserting pid 215/7 in queue
[t: 136.397] pid 106/3 departure event (bay 3)
[t: 136.397] handling pid 114/3 in bay 3
+ 1.239/1.239 (0.484) : [1] inspect (passed; r=0.194; p=0.180)
+ 2.361/1.122 (0.318) : [2] inspect (failed; r=0.019; p=0.180)
+ 3.965/1.604 (0.709) : [2] repair
+ 4.541/0.576 (0.359) : [2] inspect (passed; r=0.099; p=0.090)
> 140.94 : final bay departure scheduled
[t: 136.423] pid 94/1 departure event (bay 6)
[t: 136.423] handling pid 115/3 in bay 6
+ 1.664/1.664 (0.689) : [1] inspect (failed; r=0.120; p=0.300)
+ 2.446/0.782 (0.765) : [1] repair
+ 3.377/0.930 (0.829) : [1] inspect (passed; r=0.183; p=0.150)
+ 5.240/1.863 (0.831) : [2] inspect (passed; r=0.480; p=0.300)
+ 5.987/0.747 (0.033) : [3] inspect (passed; r=0.645; p=0.300)
+ 6.879/0.892 (0.137) : [4] inspect (passed; r=0.713; p=0.300)
> 143.30 : final bay departure scheduled
[t: 137.477] pid 193/4 arrival event - queue len: 100
> 148.99 (0.254) : scheduled arrival for pid 217/4
* all bays busy; inserting pid 193/4 in queue
[t: 138.082] pid 103/7 departure event (bay 4)
[t: 138.082] handling pid 105/2 in bay 4
+ 1.934/1.934 (0.335) : [1] inspect (passed; r=0.928; p=0.210)
+ 3.915/1.982 (0.816) : [2] inspect (passed; r=0.334; p=0.210)
+ 5.868/1.952 (0.524) : [3] inspect (passed; r=0.326; p=0.210)
> 143.95 : final bay departure scheduled
[t: 139.307] pid 206/5 arrival event - queue len: 100
> 154.32 (0.252) : scheduled arrival for pid 218/5
* all bays busy; inserting pid 206/5 in queue
[t: 140.217] pid 210/2 arrival event - queue len: 101
> 149.33 (0.043) : scheduled arrival for pid 219/2
* all bays busy; inserting pid 210/2 in queue
[t: 140.938] pid 114/3 departure event (bay 3)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 140.938] handling pid 117/2 in bay 3
+ 1.066/1.066 (0.238) : [1] inspect (passed; r=0.238; p=0.180)
+ 2.084/1.017 (0.168) : [2] inspect (passed; r=0.383; p=0.180)
> 143.02 : final bay departure scheduled
[t: 141.306] pid 110/3 departure event (bay 5)
[t: 141.306] handling pid 100/5 in bay 5
+ 1.561/1.561 (0.944) : [1] inspect (passed; r=0.767; p=0.180)
+ 2.927/1.366 (0.666) : [2] inspect (passed; r=0.710; p=0.180)
> 144.23 : final bay departure scheduled
[t: 142.197] pid 216/7 arrival event - queue len: 100
> 144.74 (0.428) : scheduled arrival for pid 220/7
* all bays busy; inserting pid 216/7 in queue
[t: 142.383] pid 211/3 arrival event - queue len: 101
> 142.77 (0.899) : scheduled arrival for pid 221/3
* all bays busy; inserting pid 211/3 in queue
[t: 142.723] pid 111/3 departure event (bay 1)
[t: 142.723] handling pid 108/1 in bay 1
+ 1.241/1.241 (0.487) : [1] inspect (passed; r=0.918; p=0.180)
+ 2.356/1.115 (0.307) : [2] inspect (failed; r=0.056; p=0.180)
+ 3.593/1.237 (0.692) : [2] repair
+ 4.256/0.663 (0.608) : [2] inspect (passed; r=0.731; p=0.090)
> 146.98 : final bay departure scheduled
[t: 142.766] pid 221/3 arrival event - queue len: 101
> 144.16 (0.680) : scheduled arrival for pid 222/3
* all bays busy; inserting pid 221/3 in queue
[t: 143.021] pid 117/2 departure event (bay 3)
[t: 143.021] handling pid 107/6 in bay 3
+ 2.049/2.049 (0.170) : [1] inspect (passed; r=0.655; p=0.260)
+ 3.859/1.810 (0.966) : [2] inspect (failed; r=0.011; p=0.260)
+ 4.624/0.765 (0.542) : [2] repair
+ 5.602/0.978 (0.481) : [2] inspect (failed; r=0.078; p=0.130)
+ 6.039/0.436 (0.577) : [2] repair
+ 6.940/0.902 (0.990) : [2] inspect (failed; r=0.063; p=0.130)
+ 7.216/0.275 (0.802) : [2] repair
+ 8.171/0.955 (0.631) : [2] inspect (passed; r=0.410; p=0.130)
+10.123/1.952 (0.495) : [3] inspect (passed; r=0.910; p=0.260)
> 153.14 : final bay departure scheduled
[t: 143.302] pid 115/3 departure event (bay 6)
[t: 143.302] handling pid 112/4 in bay 6
+ 1.285/1.285 (0.550) : [1] inspect (passed; r=0.713; p=0.180)
+ 2.196/0.911 (0.016) : [2] inspect (passed; r=0.399; p=0.180)
> 145.50 : final bay departure scheduled
[t: 143.950] pid 105/2 departure event (bay 4)
[t: 143.950] handling pid 120/1 in bay 4
+ 1.989/1.989 (0.370) : [1] inspect (passed; r=0.458; p=0.260)
+ 4.041/2.052 (0.161) : [2] inspect (passed; r=0.635; p=0.260)
+ 5.969/1.928 (0.573) : [3] inspect (failed; r=0.132; p=0.260)
+ 7.195/1.226 (0.590) : [3] repair
+ 8.137/0.942 (0.717) : [3] inspect (passed; r=0.528; p=0.130)
> 152.09 : final bay departure scheduled
[t: 144.156] pid 222/3 arrival event - queue len: 99
> 146.67 (0.497) : scheduled arrival for pid 223/3
* all bays busy; inserting pid 222/3 in queue
[t: 144.185] pid 91/4 departure event (bay 2)
[t: 144.185] handling pid 113/7 in bay 2
+ 2.143/2.143 (0.343) : [1] inspect (passed; r=0.636; p=0.120)
+ 4.940/2.798 (0.998) : [2] inspect (passed; r=0.153; p=0.120)
+ 7.639/2.699 (0.899) : [3] inspect (passed; r=0.359; p=0.120)
+10.146/2.507 (0.707) : [4] inspect (passed; r=0.998; p=0.120)
> 154.33 : final bay departure scheduled
[t: 144.233] pid 100/5 departure event (bay 5)
[t: 144.233] handling pid 118/2 in bay 5
+ 1.181/1.181 (0.272) : [1] inspect (passed; r=0.861; p=0.360)
+ 2.334/1.154 (0.253) : [2] inspect (passed; r=0.453; p=0.360)
+ 3.947/1.613 (0.580) : [3] inspect (failed; r=0.349; p=0.360)
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 7.792/3.845 (0.657) : [3] repair
+ 8.728/0.936 (0.766) : [3] inspect (passed; r=0.454; p=0.180)
+10.315/1.587 (0.562) : [4] inspect (passed; r=0.631; p=0.360)
> 154.55 : final bay departure scheduled
[t: 144.743] pid 220/7 arrival event - queue len: 98
> 145.88 (0.685) : scheduled arrival for pid 224/7
* all bays busy; inserting pid 220/7 in queue
[t: 144.923] pid 214/6 arrival event - queue len: 99
> 145.05 (0.981) : scheduled arrival for pid 225/6
* all bays busy; inserting pid 214/6 in queue
[t: 145.051] pid 225/6 arrival event - queue len: 100
> 147.17 (0.729) : scheduled arrival for pid 226/6
* all bays busy; inserting pid 225/6 in queue
[t: 145.498] pid 112/4 departure event (bay 6)
[t: 145.498] handling pid 121/6 in bay 6
+ 2.505/2.505 (0.705) : [1] inspect (passed; r=0.648; p=0.120)
+ 4.872/2.367 (0.567) : [2] inspect (passed; r=0.997; p=0.120)
+ 6.828/1.956 (0.156) : [3] inspect (passed; r=0.435; p=0.120)
+ 9.512/2.684 (0.884) : [4] inspect (passed; r=0.863; p=0.120)
> 155.01 : final bay departure scheduled
[t: 145.858] pid 207/1 arrival event - queue len: 100
> 152.82 (0.424) : scheduled arrival for pid 227/1
* all bays busy; inserting pid 207/1 in queue
[t: 145.879] pid 224/7 arrival event - queue len: 101
> 149.75 (0.275) : scheduled arrival for pid 228/7
* all bays busy; inserting pid 224/7 in queue
[t: 146.670] pid 223/3 arrival event - queue len: 102
> 146.95 (0.925) : scheduled arrival for pid 229/3
* all bays busy; inserting pid 223/3 in queue
[t: 146.951] pid 229/3 arrival event - queue len: 103
> 150.54 (0.369) : scheduled arrival for pid 230/3
* all bays busy; inserting pid 229/3 in queue
[t: 146.979] pid 108/1 departure event (bay 1)
[t: 146.979] handling pid 125/2 in bay 1
+ 1.420/1.420 (0.514) : [1] inspect (passed; r=0.950; p=0.300)
+ 2.627/1.208 (0.363) : [2] inspect (passed; r=0.814; p=0.300)
+ 4.437/1.809 (0.792) : [3] inspect (passed; r=0.890; p=0.300)
+ 5.290/0.853 (0.110) : [4] inspect (passed; r=0.587; p=0.300)
> 152.27 : final bay departure scheduled
[t: 147.168] pid 226/6 arrival event - queue len: 103
> 149.34 (0.723) : scheduled arrival for pid 231/6
* all bays busy; inserting pid 226/6 in queue
[t: 148.993] pid 217/4 arrival event - queue len: 104
> 153.04 (0.618) : scheduled arrival for pid 232/4
* all bays busy; inserting pid 217/4 in queue
[t: 149.329] pid 219/2 arrival event - queue len: 105
> 149.80 (0.850) : scheduled arrival for pid 233/2
* all bays busy; inserting pid 219/2 in queue
[t: 149.343] pid 231/6 arrival event - queue len: 106
> 165.41 (0.091) : scheduled arrival for pid 234/6
* all bays busy; inserting pid 231/6 in queue
[t: 149.749] pid 228/7 arrival event - queue len: 107
> 150.15 (0.876) : scheduled arrival for pid 235/7
* all bays busy; inserting pid 228/7 in queue
[t: 149.802] pid 233/2 arrival event - queue len: 108
> 157.18 (0.079) : scheduled arrival for pid 236/2
* all bays busy; inserting pid 233/2 in queue
[t: 150.148] pid 235/7 arrival event - queue len: 109
> 157.26 (0.093) : scheduled arrival for pid 237/7
* all bays busy; inserting pid 235/7 in queue
[t: 150.539] pid 230/3 arrival event - queue len: 110
> 150.67 (0.964) : scheduled arrival for pid 238/3
* all bays busy; inserting pid 230/3 in queue
[t: 150.670] pid 238/3 arrival event - queue len: 111
> 152.35 (0.626) : scheduled arrival for pid 239/3
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 238/3 in queue
[t: 152.087] pid 120/1 departure event (bay 4)
[t: 152.087] handling pid 116/3 in bay 4
+ 1.741/1.741 (0.743) : [1] inspect (passed; r=0.493; p=0.300)
+ 3.280/1.540 (0.600) : [2] inspect (passed; r=0.609; p=0.300)
+ 4.648/1.368 (0.477) : [3] inspect (failed; r=0.050; p=0.300)
+ 6.992/2.344 (0.342) : [3] repair
+ 7.534/0.543 (0.275) : [3] inspect (passed; r=0.370; p=0.150)
+ 8.236/0.702 (0.001) : [4] inspect (failed; r=0.202; p=0.300)
+10.022/1.786 (0.193) : [4] repair
+10.707/0.685 (0.478) : [4] inspect (passed; r=0.493; p=0.150)
> 162.79 : final bay departure scheduled
[t: 152.269] pid 125/2 departure event (bay 1)
[t: 152.269] handling pid 124/7 in bay 1
+ 1.849/1.849 (0.838) : [1] inspect (failed; r=0.179; p=0.260)
+ 3.299/1.451 (0.322) : [1] repair
+ 4.325/1.026 (0.159) : [1] inspect (passed; r=0.139; p=0.130)
+ 6.238/1.913 (0.623) : [2] inspect (failed; r=0.098; p=0.260)
+ 7.178/0.940 (0.377) : [2] repair
+ 8.193/1.015 (0.235) : [2] inspect (passed; r=0.213; p=0.130)
+10.006/1.813 (0.956) : [3] inspect (passed; r=0.784; p=0.260)
> 162.27 : final bay departure scheduled
[t: 152.354] pid 239/3 arrival event - queue len: 110
> 162.47 (0.060) : scheduled arrival for pid 240/3
* all bays busy; inserting pid 239/3 in queue
[t: 152.817] pid 227/1 arrival event - queue len: 111
> 183.24 (0.023) : scheduled arrival for pid 241/1
* all bays busy; inserting pid 227/1 in queue
[t: 153.036] pid 232/4 arrival event - queue len: 112
> 156.13 (0.692) : scheduled arrival for pid 242/4
* all bays busy; inserting pid 232/4 in queue
[t: 153.144] pid 107/6 departure event (bay 3)
[t: 153.144] handling pid 128/3 in bay 3
+ 1.694/1.694 (0.935) : [1] inspect (passed; r=0.734; p=0.140)
+ 3.338/1.645 (0.446) : [2] inspect (passed; r=0.777; p=0.140)
> 156.48 : final bay departure scheduled
[t: 154.325] pid 218/5 arrival event - queue len: 112
> 173.77 (0.168) : scheduled arrival for pid 243/5
* all bays busy; inserting pid 218/5 in queue
[t: 154.331] pid 113/7 departure event (bay 2)
[t: 154.331] handling pid 127/2 in bay 2
+ 1.904/1.904 (0.041) : [1] inspect (passed; r=0.332; p=0.210)
+ 3.832/1.928 (0.277) : [2] inspect (passed; r=0.919; p=0.210)
+ 5.744/1.912 (0.121) : [3] inspect (passed; r=0.550; p=0.210)
> 160.07 : final bay departure scheduled
[t: 154.548] pid 118/2 departure event (bay 5)
[t: 154.548] handling pid 123/1 in bay 5
+ 1.965/1.965 (0.450) : [1] inspect (passed; r=0.993; p=0.260)
+ 4.000/2.034 (0.219) : [2] inspect (passed; r=0.412; p=0.260)
+ 5.825/1.826 (0.915) : [3] inspect (passed; r=0.436; p=0.260)
> 160.37 : final bay departure scheduled
[t: 155.010] pid 121/6 departure event (bay 6)
[t: 155.010] handling pid 129/7 in bay 6
+ 1.647/1.647 (0.468) : [1] inspect (failed; r=0.112; p=0.140)
+ 2.108/0.461 (0.602) : [1] repair
+ 2.937/0.829 (0.589) : [1] inspect (passed; r=0.805; p=0.070)
+ 4.588/1.651 (0.509) : [2] inspect (passed; r=0.621; p=0.140)
> 159.60 : final bay departure scheduled
[t: 156.133] pid 242/4 arrival event - queue len: 110
> 171.64 (0.158) : scheduled arrival for pid 244/4
* all bays busy; inserting pid 242/4 in queue
[t: 156.482] pid 128/3 departure event (bay 3)
[t: 156.482] handling pid 132/1 in bay 3
+ 1.410/1.410 (0.728) : [1] inspect (failed; r=0.040; p=0.180)
+ 2.585/1.175 (0.255) : [1] repair
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 3.203/0.618 (0.480) : [1] inspect (passed; r=0.613; p=0.090)
+ 4.555/1.352 (0.646) : [2] inspect (passed; r=0.672; p=0.180)
> 161.04 : final bay departure scheduled
[t: 157.176] pid 236/2 arrival event - queue len: 110
> 157.40 (0.925) : scheduled arrival for pid 245/2
* all bays busy; inserting pid 236/2 in queue
[t: 157.264] pid 237/7 arrival event - queue len: 111
> 161.21 (0.269) : scheduled arrival for pid 246/7
* all bays busy; inserting pid 237/7 in queue
[t: 157.401] pid 245/2 arrival event - queue len: 112
> 159.69 (0.454) : scheduled arrival for pid 247/2
* all bays busy; inserting pid 245/2 in queue
[t: 159.598] pid 129/7 departure event (bay 6)
[t: 159.598] handling pid 130/3 in bay 6
+ 1.947/1.947 (0.475) : [1] inspect (passed; r=0.868; p=0.210)
+ 3.857/1.909 (0.091) : [2] inspect (passed; r=0.780; p=0.210)
+ 5.853/1.996 (0.961) : [3] inspect (failed; r=0.086; p=0.210)
+10.384/4.532 (0.374) : [3] repair
+11.345/0.961 (0.219) : [3] inspect (passed; r=0.824; p=0.105)
> 170.94 : final bay departure scheduled
[t: 159.691] pid 247/2 arrival event - queue len: 112
> 161.31 (0.573) : scheduled arrival for pid 248/2
* all bays busy; inserting pid 247/2 in queue
[t: 160.075] pid 127/2 departure event (bay 2)
[t: 160.075] handling pid 135/3 in bay 2
+ 1.935/1.935 (0.548) : [1] inspect (passed; r=0.802; p=0.260)
+ 3.803/1.867 (0.775) : [2] inspect (passed; r=0.621; p=0.260)
+ 5.603/1.800 (1.000) : [3] inspect (passed; r=0.525; p=0.260)
> 165.68 : final bay departure scheduled
[t: 160.373] pid 123/1 departure event (bay 5)
[t: 160.373] handling pid 131/2 in bay 5
+ 0.785/0.785 (0.061) : [1] inspect (passed; r=0.308; p=0.300)
+ 1.805/1.020 (0.229) : [2] inspect (passed; r=0.866; p=0.300)
+ 3.291/1.486 (0.561) : [3] inspect (passed; r=0.701; p=0.300)
+ 5.093/1.802 (0.787) : [4] inspect (passed; r=0.373; p=0.300)
> 165.47 : final bay departure scheduled
[t: 161.037] pid 132/1 departure event (bay 3)
[t: 161.037] handling pid 119/5 in bay 3
+ 2.048/2.048 (0.963) : [1] inspect (passed; r=0.923; p=0.300)
+ 3.567/1.518 (0.584) : [2] inspect (passed; r=0.974; p=0.300)
+ 5.394/1.828 (0.805) : [3] inspect (failed; r=0.211; p=0.300)
+ 8.742/3.348 (0.058) : [3] repair
+ 9.718/0.976 (0.894) : [3] inspect (passed; r=0.391; p=0.150)
+11.161/1.443 (0.531) : [4] inspect (passed; r=0.308; p=0.300)
> 172.20 : final bay departure scheduled
[t: 161.206] pid 246/7 arrival event - queue len: 110
> 162.77 (0.594) : scheduled arrival for pid 249/7
* all bays busy; inserting pid 246/7 in queue
[t: 161.308] pid 248/2 arrival event - queue len: 111
> 161.76 (0.857) : scheduled arrival for pid 250/2
* all bays busy; inserting pid 248/2 in queue
[t: 161.757] pid 250/2 arrival event - queue len: 112
> 165.38 (0.287) : scheduled arrival for pid 251/2
* all bays busy; inserting pid 250/2 in queue
[t: 162.275] pid 124/7 departure event (bay 1)
[t: 162.275] handling pid 134/1 in bay 1
+ 1.909/1.909 (0.087) : [1] inspect (passed; r=0.211; p=0.210)
+ 3.849/1.941 (0.407) : [2] inspect (passed; r=0.789; p=0.210)
+ 5.776/1.927 (0.270) : [3] inspect (passed; r=0.233; p=0.210)
> 168.05 : final bay departure scheduled
[t: 162.469] pid 240/3 arrival event - queue len: 112
> 162.56 (0.976) : scheduled arrival for pid 252/3
* all bays busy; inserting pid 240/3 in queue
[t: 162.557] pid 252/3 arrival event - queue len: 113
> 166.42 (0.342) : scheduled arrival for pid 253/3
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 252/3 in queue
[t: 162.768] pid 249/7 arrival event - queue len: 114
> 167.11 (0.235) : scheduled arrival for pid 254/7
* all bays busy; inserting pid 249/7 in queue
[t: 162.794] pid 116/3 departure event (bay 4)
[t: 162.794] handling pid 126/6 in bay 4
+ 1.515/1.515 (0.879) : [1] inspect (passed; r=0.215; p=0.180)
+ 2.657/1.142 (0.346) : [2] inspect (passed; r=0.369; p=0.180)
> 165.45 : final bay departure scheduled
[t: 165.375] pid 251/2 arrival event - queue len: 114
> 166.47 (0.685) : scheduled arrival for pid 255/2
* all bays busy; inserting pid 251/2 in queue
[t: 165.410] pid 234/6 arrival event - queue len: 115
> 175.34 (0.227) : scheduled arrival for pid 256/6
* all bays busy; inserting pid 234/6 in queue
[t: 165.451] pid 126/6 departure event (bay 4)
[t: 165.451] handling pid 140/6 in bay 4
+ 1.645/1.645 (0.445) : [1] inspect (passed; r=0.563; p=0.140)
+ 3.279/1.634 (0.341) : [2] inspect (passed; r=0.478; p=0.140)
> 168.73 : final bay departure scheduled
[t: 165.466] pid 131/2 departure event (bay 5)
[t: 165.466] handling pid 137/2 in bay 5
+ 2.038/2.038 (0.207) : [1] inspect (passed; r=0.361; p=0.260)
+ 4.109/2.071 (0.097) : [2] inspect (passed; r=0.687; p=0.260)
+ 5.985/1.876 (0.746) : [3] inspect (passed; r=0.803; p=0.260)
> 171.45 : final bay departure scheduled
[t: 165.678] pid 135/3 departure event (bay 2)
[t: 165.678] handling pid 142/2 in bay 2
+ 0.914/0.914 (0.020) : [1] inspect (failed; r=0.069; p=0.180)
+ 1.403/0.489 (0.773) : [1] repair
+ 1.862/0.459 (0.025) : [1] inspect (passed; r=0.616; p=0.090)
+ 3.012/1.150 (0.357) : [2] inspect (failed; r=0.030; p=0.180)
+ 7.511/4.499 (0.178) : [2] repair
+ 8.123/0.612 (0.463) : [2] inspect (passed; r=0.949; p=0.090)
> 173.80 : final bay departure scheduled
[t: 166.417] pid 253/3 arrival event - queue len: 113
> 171.95 (0.215) : scheduled arrival for pid 257/3
* all bays busy; inserting pid 253/3 in queue
[t: 166.472] pid 255/2 arrival event - queue len: 114
> 170.74 (0.230) : scheduled arrival for pid 258/2
* all bays busy; inserting pid 255/2 in queue
[t: 167.109] pid 254/7 arrival event - queue len: 115
> 169.82 (0.405) : scheduled arrival for pid 259/7
* all bays busy; inserting pid 254/7 in queue
[t: 168.051] pid 134/1 departure event (bay 1)
[t: 168.051] handling pid 143/2 in bay 1
+ 1.218/1.218 (0.370) : [1] inspect (passed; r=0.444; p=0.300)
+ 3.274/2.056 (0.969) : [2] inspect (passed; r=0.388; p=0.300)
+ 4.459/1.185 (0.346) : [3] inspect (failed; r=0.014; p=0.300)
+ 6.213/1.754 (0.214) : [3] repair
+ 6.584/0.371 (0.031) : [3] inspect (failed; r=0.103; p=0.150)
+ 8.174/1.590 (0.424) : [3] repair
+ 9.154/0.980 (0.900) : [3] inspect (passed; r=0.374; p=0.150)
+ 11.047/1.893 (0.852) : [4] inspect (passed; r=0.576; p=0.300)
> 179.10 : final bay departure scheduled
[t: 168.730] pid 140/6 departure event (bay 4)
[t: 168.730] handling pid 139/1 in bay 4
+ 1.624/1.624 (0.241) : [1] inspect (failed; r=0.075; p=0.140)
+ 2.421/0.797 (0.698) : [1] repair
+ 3.239/0.819 (0.371) : [1] inspect (passed; r=0.434; p=0.070)
+ 4.937/1.698 (0.975) : [2] inspect (passed; r=0.943; p=0.140)
> 173.67 : final bay departure scheduled
[t: 169.817] pid 259/7 arrival event - queue len: 114
> 178.55 (0.054) : scheduled arrival for pid 260/7
* all bays busy; inserting pid 259/7 in queue
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 170.738] pid 258/2 arrival event - queue len: 115
> 171.94 (0.661) : scheduled arrival for pid 261/2
* all bays busy; inserting pid 258/2 in queue
[t: 170.944] pid 130/3 departure event (bay 6)
[t: 170.944] handling pid 144/2 in bay 6
+ 1.169/1.169 (0.384) : [1] inspect (passed; r=0.830; p=0.180)
+ 2.189/1.021 (0.172) : [2] inspect (failed; r=0.152; p=0.180)
+ 3.476/1.287 (0.297) : [2] repair
+ 3.987/0.511 (0.174) : [2] inspect (passed; r=0.844; p=0.090)
> 174.93 : final bay departure scheduled
[t: 171.451] pid 137/2 departure event (bay 5)
[t: 171.451] handling pid 145/1 in bay 5
+ 1.885/1.885 (0.715) : [1] inspect (passed; r=0.832; p=0.260)
+ 3.712/1.826 (0.912) : [2] inspect (passed; r=0.825; p=0.260)
+ 5.639/1.927 (0.576) : [3] inspect (passed; r=0.652; p=0.260)
> 177.09 : final bay departure scheduled
[t: 171.641] pid 244/4 arrival event - queue len: 114
> 181.82 (0.298) : scheduled arrival for pid 262/4
* all bays busy; inserting pid 244/4 in queue
[t: 171.938] pid 261/2 arrival event - queue len: 115
> 173.50 (0.584) : scheduled arrival for pid 263/2
* all bays busy; inserting pid 261/2 in queue
[t: 171.951] pid 257/3 arrival event - queue len: 116
> 174.95 (0.435) : scheduled arrival for pid 264/3
* all bays busy; inserting pid 257/3 in queue
[t: 172.199] pid 119/5 departure event (bay 3)
[t: 172.199] handling pid 133/7 in bay 3
+ 1.476/1.476 (0.483) : [1] inspect (passed; r=0.990; p=0.360)
+ 2.910/1.434 (0.453) : [2] inspect (passed; r=0.564; p=0.360)
+ 3.932/1.022 (0.158) : [3] inspect (passed; r=0.420; p=0.360)
+ 6.075/2.143 (0.959) : [4] inspect (passed; r=0.418; p=0.360)
> 178.27 : final bay departure scheduled
[t: 173.500] pid 263/2 arrival event - queue len: 116
> 174.21 (0.784) : scheduled arrival for pid 265/2
* all bays busy; inserting pid 263/2 in queue
[t: 173.667] pid 139/1 departure event (bay 4)
[t: 173.667] handling pid 147/1 in bay 4
+ 1.848/1.848 (0.820) : [1] inspect (failed; r=0.063; p=0.300)
+ 4.462/2.614 (0.228) : [1] repair
+ 4.970/0.508 (0.226) : [1] inspect (passed; r=0.333; p=0.150)
+ 6.281/1.311 (0.436) : [2] inspect (passed; r=0.333; p=0.300)
+ 7.640/1.359 (0.471) : [3] inspect (passed; r=0.726; p=0.300)
+ 9.232/1.592 (0.637) : [4] inspect (passed; r=0.790; p=0.300)
> 182.90 : final bay departure scheduled
[t: 173.773] pid 243/5 arrival event - queue len: 116
> 188.88 (0.250) : scheduled arrival for pid 266/5
* all bays busy; inserting pid 243/5 in queue
[t: 173.801] pid 142/2 departure event (bay 2)
[t: 173.801] handling pid 136/3 in bay 2
+ 1.947/1.947 (0.511) : [1] inspect (passed; r=0.482; p=0.260)
+ 3.841/1.894 (0.687) : [2] inspect (failed; r=0.153; p=0.260)
+ 5.202/1.361 (0.740) : [2] repair
+ 6.172/0.970 (0.531) : [2] inspect (passed; r=0.263; p=0.130)
+ 8.101/1.929 (0.571) : [3] inspect (passed; r=0.970; p=0.260)
> 181.90 : final bay departure scheduled
[t: 174.207] pid 265/2 arrival event - queue len: 116
> 174.34 (0.954) : scheduled arrival for pid 267/2
* all bays busy; inserting pid 265/2 in queue
[t: 174.342] pid 267/2 arrival event - queue len: 117
> 175.18 (0.748) : scheduled arrival for pid 268/2
* all bays busy; inserting pid 267/2 in queue
[t: 174.931] pid 144/2 departure event (bay 6)
[t: 174.931] handling pid 146/2 in bay 6
+ 1.810/1.810 (0.968) : [1] inspect (passed; r=0.877; p=0.260)
+ 3.786/1.977 (0.410) : [2] inspect (passed; r=0.799; p=0.260)
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 5.606/1.820 (0.933) : [3] inspect (failed; r=0.119; p=0.260)
+ 6.091/0.484 (0.774) : [3] repair
+ 7.117/1.026 (0.161) : [3] inspect (passed; r=0.303; p=0.130)
> 182.05 : final bay departure scheduled
[t: 174.946] pid 264/3 arrival event - queue len: 117
> 178.68 (0.355) : scheduled arrival for pid 269/3
* all bays busy; inserting pid 264/3 in queue
[t: 175.183] pid 268/2 arrival event - queue len: 118
> 175.69 (0.841) : scheduled arrival for pid 270/2
* all bays busy; inserting pid 268/2 in queue
[t: 175.340] pid 256/6 arrival event - queue len: 119
> 183.17 (0.311) : scheduled arrival for pid 271/6
* all bays busy; inserting pid 256/6 in queue
[t: 175.687] pid 270/2 arrival event - queue len: 120
> 175.97 (0.907) : scheduled arrival for pid 272/2
* all bays busy; inserting pid 270/2 in queue
[t: 175.971] pid 272/2 arrival event - queue len: 121
> 175.98 (0.997) : scheduled arrival for pid 273/2
* all bays busy; inserting pid 272/2 in queue
[t: 175.979] pid 273/2 arrival event - queue len: 122
> 176.27 (0.904) : scheduled arrival for pid 274/2
* all bays busy; inserting pid 273/2 in queue
[t: 176.271] pid 274/2 arrival event - queue len: 123
> 176.73 (0.855) : scheduled arrival for pid 275/2
* all bays busy; inserting pid 274/2 in queue
[t: 176.725] pid 275/2 arrival event - queue len: 124
> 177.22 (0.844) : scheduled arrival for pid 276/2
* all bays busy; inserting pid 275/2 in queue
[t: 177.090] pid 145/1 departure event (bay 5)
[t: 177.090] handling pid 138/5 in bay 5
+ 0.899/0.899 (0.142) : [1] inspect (passed; r=0.870; p=0.300)
+ 1.801/0.902 (0.144) : [2] inspect (passed; r=0.605; p=0.300)
+ 2.760/0.959 (0.185) : [3] inspect (passed; r=0.738; p=0.300)
+ 3.714/0.955 (0.182) : [4] inspect (passed; r=0.818; p=0.300)
> 180.80 : final bay departure scheduled
[t: 177.219] pid 276/2 arrival event - queue len: 124
> 182.05 (0.189) : scheduled arrival for pid 277/2
* all bays busy; inserting pid 276/2 in queue
[t: 178.274] pid 133/7 departure event (bay 3)
[t: 178.274] handling pid 150/3 in bay 3
+ 1.968/1.968 (0.683) : [1] inspect (passed; r=0.827; p=0.210)
+ 3.956/1.987 (0.872) : [2] inspect (failed; r=0.174; p=0.210)
+ 6.495/2.539 (0.779) : [2] repair
+ 7.493/0.998 (0.963) : [2] inspect (failed; r=0.051; p=0.105)
+ 8.318/0.825 (0.756) : [2] repair
+ 9.306/0.988 (0.764) : [2] inspect (passed; r=0.485; p=0.105)
+11.232/1.926 (0.264) : [3] inspect (passed; r=0.883; p=0.210)
> 189.51 : final bay departure scheduled
[t: 178.550] pid 260/7 arrival event - queue len: 124
> 179.67 (0.689) : scheduled arrival for pid 278/7
* all bays busy; inserting pid 260/7 in queue
[t: 178.676] pid 269/3 arrival event - queue len: 125
> 180.47 (0.607) : scheduled arrival for pid 279/3
* all bays busy; inserting pid 269/3 in queue
[t: 179.098] pid 143/2 departure event (bay 1)
[t: 179.098] handling pid 152/5 in bay 1
+ 1.805/1.805 (0.983) : [1] inspect (passed; r=0.290; p=0.260)
+ 3.677/1.872 (0.761) : [2] inspect (passed; r=0.899; p=0.260)
+ 5.635/1.958 (0.472) : [3] inspect (passed; r=0.721; p=0.260)
> 184.73 : final bay departure scheduled
[t: 179.668] pid 278/7 arrival event - queue len: 125
> 180.87 (0.671) : scheduled arrival for pid 280/7
* all bays busy; inserting pid 278/7 in queue
[t: 180.473] pid 279/3 arrival event - queue len: 126
> 180.60 (0.965) : scheduled arrival for pid 281/3
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 279/3 in queue
[t: 180.603] pid 281/3 arrival event - queue len: 127
> 182.71 (0.557) : scheduled arrival for pid 282/3
* all bays busy; inserting pid 281/3 in queue
[t: 180.805] pid 138/5 departure event (bay 5)
[t: 180.805] handling pid 149/1 in bay 5
+ 1.380/1.380 (0.414) : [1] inspect (passed; r=0.518; p=0.360)
+ 2.960/1.579 (0.557) : [2] inspect (passed; r=0.840; p=0.360)
+ 4.511/1.552 (0.537) : [3] inspect (passed; r=0.733; p=0.360)
+ 6.670/2.159 (0.971) : [4] inspect (passed; r=0.755; p=0.360)
> 187.47 : final bay departure scheduled
[t: 180.867] pid 280/7 arrival event - queue len: 127
> 183.30 (0.444) : scheduled arrival for pid 283/7
* all bays busy; inserting pid 280/7 in queue
[t: 181.821] pid 262/4 arrival event - queue len: 128
> 195.92 (0.187) : scheduled arrival for pid 284/4
* all bays busy; inserting pid 262/4 in queue
[t: 181.902] pid 136/3 departure event (bay 2)
[t: 181.902] handling pid 153/3 in bay 2
+ 1.093/1.093 (0.276) : [1] inspect (passed; r=0.751; p=0.180)
+ 2.559/1.466 (0.808) : [2] inspect (failed; r=0.119; p=0.180)
+ 4.486/1.927 (0.163) : [2] repair
+ 4.984/0.498 (0.137) : [2] inspect (passed; r=0.396; p=0.090)
> 186.89 : final bay departure scheduled
[t: 182.048] pid 146/2 departure event (bay 6)
[t: 182.048] handling pid 156/3 in bay 6
+ 2.029/2.029 (0.238) : [1] inspect (passed; r=0.395; p=0.260)
+ 3.846/1.817 (0.944) : [2] inspect (failed; r=0.184; p=0.260)
+ 5.105/1.260 (0.599) : [2] repair
+ 6.023/0.918 (0.881) : [2] inspect (passed; r=0.175; p=0.130)
+ 7.861/1.838 (0.875) : [3] inspect (passed; r=0.467; p=0.260)
> 189.91 : final bay departure scheduled
[t: 182.052] pid 277/2 arrival event - queue len: 127
> 185.65 (0.290) : scheduled arrival for pid 285/2
* all bays busy; inserting pid 277/2 in queue
[t: 182.712] pid 282/3 arrival event - queue len: 128
> 193.16 (0.055) : scheduled arrival for pid 286/3
* all bays busy; inserting pid 282/3 in queue
[t: 182.899] pid 147/1 departure event (bay 4)
[t: 182.899] handling pid 155/1 in bay 4
+ 1.784/1.784 (0.774) : [1] inspect (passed; r=0.657; p=0.300)
+ 2.865/1.081 (0.272) : [2] inspect (failed; r=0.240; p=0.300)
+ 6.162/3.297 (0.127) : [2] repair
+ 7.178/1.016 (0.951) : [2] inspect (passed; r=0.367; p=0.150)
+ 8.176/0.998 (0.213) : [3] inspect (passed; r=0.584; p=0.300)
+ 9.004/0.829 (0.092) : [4] inspect (failed; r=0.295; p=0.300)
+10.290/1.286 (0.418) : [4] repair
+11.009/0.719 (0.527) : [4] inspect (passed; r=0.971; p=0.150)
> 193.91 : final bay departure scheduled
[t: 183.166] pid 271/6 arrival event - queue len: 128
> 193.60 (0.211) : scheduled arrival for pid 287/6
* all bays busy; inserting pid 271/6 in queue
[t: 183.239] pid 241/1 arrival event - queue len: 129
> 187.63 (0.582) : scheduled arrival for pid 288/1
* all bays busy; inserting pid 241/1 in queue
[t: 183.302] pid 283/7 arrival event - queue len: 130
> 189.10 (0.145) : scheduled arrival for pid 289/7
* all bays busy; inserting pid 283/7 in queue
[t: 184.733] pid 152/5 departure event (bay 1)
[t: 184.733] handling pid 151/2 in bay 1
+ 1.178/1.178 (0.270) : [1] inspect (passed; r=0.592; p=0.360)
+ 2.082/0.905 (0.075) : [2] inspect (passed; r=0.742; p=0.360)
+ 3.088/1.006 (0.147) : [3] inspect (passed; r=0.470; p=0.360)
+ 5.042/1.954 (0.825) : [4] inspect (failed; r=0.223; p=0.360)
+ 6.468/1.425 (0.644) : [4] repair
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 7.397/0.929 (0.756) : [4] inspect (passed; r=0.260; p=0.180)
> 192.13 : final bay departure scheduled
[t: 185.645] pid 285/2 arrival event - queue len: 130
> 186.16 (0.837) : scheduled arrival for pid 290/2
* all bays busy; inserting pid 285/2 in queue
[t: 186.162] pid 290/2 arrival event - queue len: 131
> 188.52 (0.444) : scheduled arrival for pid 291/2
* all bays busy; inserting pid 290/2 in queue
[t: 186.885] pid 153/3 departure event (bay 2)
[t: 186.885] handling pid 157/3 in bay 2
+ 1.100/1.100 (0.285) : [1] inspect (failed; r=0.131; p=0.180)
+ 3.884/2.784 (0.648) : [1] repair
+ 4.463/0.580 (0.371) : [1] inspect (passed; r=0.773; p=0.090)
+ 5.494/1.031 (0.187) : [2] inspect (failed; r=0.084; p=0.180)
+ 6.100/0.606 (0.731) : [2] repair
+ 6.627/0.527 (0.220) : [2] inspect (passed; r=0.786; p=0.090)
> 193.51 : final bay departure scheduled
[t: 187.475] pid 149/1 departure event (bay 5)
[t: 187.475] handling pid 159/2 in bay 5
+ 2.013/2.013 (0.938) : [1] inspect (passed; r=0.766; p=0.300)
+ 3.729/1.716 (0.726) : [2] inspect (passed; r=0.545; p=0.300)
+ 4.704/0.975 (0.196) : [3] inspect (passed; r=0.576; p=0.300)
+ 5.766/1.062 (0.259) : [4] inspect (failed; r=0.130; p=0.300)
+ 6.748/0.981 (0.536) : [4] repair
+ 7.120/0.373 (0.032) : [4] inspect (passed; r=0.767; p=0.150)
> 194.59 : final bay departure scheduled
[t: 187.629] pid 288/1 arrival event - queue len: 130
> 214.57 (0.036) : scheduled arrival for pid 292/1
* all bays busy; inserting pid 288/1 in queue
[t: 188.519] pid 291/2 arrival event - queue len: 131
> 189.18 (0.797) : scheduled arrival for pid 293/2
* all bays busy; inserting pid 291/2 in queue
[t: 188.880] pid 266/5 arrival event - queue len: 132
> 200.29 (0.351) : scheduled arrival for pid 294/5
* all bays busy; inserting pid 266/5 in queue
[t: 189.102] pid 289/7 arrival event - queue len: 133
> 190.24 (0.685) : scheduled arrival for pid 295/7
* all bays busy; inserting pid 289/7 in queue
[t: 189.177] pid 293/2 arrival event - queue len: 134
> 191.62 (0.431) : scheduled arrival for pid 296/2
* all bays busy; inserting pid 293/2 in queue
[t: 189.506] pid 150/3 departure event (bay 3)
[t: 189.506] handling pid 161/2 in bay 3
+ 0.998/0.998 (0.140) : [1] inspect (passed; r=0.720; p=0.180)
+ 2.312/1.314 (0.591) : [2] inspect (passed; r=0.803; p=0.180)
> 191.82 : final bay departure scheduled
[t: 189.909] pid 156/3 departure event (bay 6)
[t: 189.909] handling pid 122/4 in bay 6
+ 1.399/1.399 (0.713) : [1] inspect (failed; r=0.067; p=0.180)
+ 2.682/1.282 (0.203) : [1] repair
+ 3.380/0.698 (0.709) : [1] inspect (passed; r=0.285; p=0.090)
+ 4.335/0.955 (0.078) : [2] inspect (failed; r=0.083; p=0.180)
+ 4.807/0.472 (0.783) : [2] repair
+ 5.572/0.765 (0.901) : [2] inspect (passed; r=0.253; p=0.090)
> 195.48 : final bay departure scheduled
[t: 190.237] pid 295/7 arrival event - queue len: 133
> 191.94 (0.567) : scheduled arrival for pid 297/7
* all bays busy; inserting pid 295/7 in queue
[t: 191.619] pid 296/2 arrival event - queue len: 134
> 198.36 (0.098) : scheduled arrival for pid 298/2
* all bays busy; inserting pid 296/2 in queue
[t: 191.818] pid 161/2 departure event (bay 3)
[t: 191.818] handling pid 160/3 in bay 3
+ 2.022/2.022 (0.259) : [1] inspect (passed; r=0.453; p=0.260)
+ 3.879/1.857 (0.809) : [2] inspect (passed; r=0.586; p=0.260)
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 5.921/2.042 (0.194) : [3] inspect (failed; r=0.141; p=0.260)
+ 6.730/0.808 (0.560) : [3] repair
+ 7.779/1.049 (0.008) : [3] inspect (passed; r=0.436; p=0.130)
> 199.60 : final bay departure scheduled
[t: 191.939] pid 297/7 arrival event - queue len: 134
> 192.14 (0.936) : scheduled arrival for pid 299/7
* all bays busy; inserting pid 297/7 in queue
[t: 192.131] pid 151/2 departure event (bay 1)
[t: 192.131] handling pid 162/2 in bay 1
+ 2.019/2.019 (0.271) : [1] inspect (failed; r=0.123; p=0.260)
+ 3.616/1.598 (0.384) : [1] repair
+ 4.631/1.015 (0.232) : [1] inspect (passed; r=0.900; p=0.130)
+ 6.519/1.887 (0.709) : [2] inspect (failed; r=0.003; p=0.260)
+ 7.019/0.500 (0.691) : [2] repair
+ 7.956/0.938 (0.749) : [2] inspect (passed; r=0.981; p=0.130)
+ 9.817/1.861 (0.797) : [3] inspect (passed; r=0.763; p=0.260)
> 201.95 : final bay departure scheduled
[t: 192.138] pid 299/7 arrival event - queue len: 134
> 195.53 (0.323) : scheduled arrival for pid 300/7
* all bays busy; inserting pid 299/7 in queue
[t: 193.162] pid 286/3 arrival event - queue len: 135
> 193.18 (0.995) : scheduled arrival for pid 301/3
* all bays busy; inserting pid 286/3 in queue
[t: 193.181] pid 301/3 arrival event - queue len: 136
> 200.63 (0.126) : scheduled arrival for pid 302/3
* all bays busy; inserting pid 301/3 in queue
[t: 193.513] pid 157/3 departure event (bay 2)
[t: 193.513] handling pid 154/5 in bay 2
+ 1.302/1.302 (0.574) : [1] inspect (passed; r=0.263; p=0.180)
+ 2.316/1.014 (0.163) : [2] inspect (passed; r=0.376; p=0.180)
> 195.83 : final bay departure scheduled
[t: 193.600] pid 287/6 arrival event - queue len: 136
> 222.17 (0.014) : scheduled arrival for pid 303/6
* all bays busy; inserting pid 287/6 in queue
[t: 193.908] pid 155/1 departure event (bay 4)
[t: 193.908] handling pid 165/2 in bay 4
+ 0.933/0.933 (0.167) : [1] inspect (passed; r=0.671; p=0.300)
+ 2.041/1.108 (0.291) : [2] inspect (passed; r=0.824; p=0.300)
+ 2.850/0.809 (0.078) : [3] inspect (passed; r=0.817; p=0.300)
+ 4.237/1.387 (0.491) : [4] inspect (passed; r=0.314; p=0.300)
> 198.15 : final bay departure scheduled
[t: 194.595] pid 159/2 departure event (bay 5)
[t: 194.595] handling pid 148/7 in bay 5
+ 1.826/1.826 (0.914) : [1] inspect (failed; r=0.181; p=0.260)
+ 4.898/3.073 (0.395) : [1] repair
+ 5.914/1.015 (0.232) : [1] inspect (passed; r=0.891; p=0.130)
+ 7.994/2.081 (0.064) : [2] inspect (passed; r=0.767; p=0.260)
+ 9.859/1.865 (0.784) : [3] inspect (passed; r=0.707; p=0.260)
> 204.45 : final bay departure scheduled
[t: 195.481] pid 122/4 departure event (bay 6)
[t: 195.481] handling pid 166/5 in bay 6
+ 2.055/2.055 (0.255) : [1] inspect (passed; r=0.595; p=0.120)
+ 3.863/1.808 (0.008) : [2] inspect (passed; r=0.414; p=0.120)
+ 6.012/2.149 (0.349) : [3] inspect (passed; r=0.752; p=0.120)
+ 7.959/1.947 (0.147) : [4] inspect (passed; r=0.901; p=0.120)
> 203.44 : final bay departure scheduled
[t: 195.529] pid 300/7 arrival event - queue len: 134
> 209.62 (0.009) : scheduled arrival for pid 304/7
* all bays busy; inserting pid 300/7 in queue
[t: 195.829] pid 154/5 departure event (bay 2)
[t: 195.829] handling pid 168/7 in bay 2
+ 1.473/1.473 (0.480) : [1] inspect (passed; r=0.663; p=0.360)
+ 2.712/1.239 (0.314) : [2] inspect (failed; r=0.267; p=0.360)
+ 5.369/2.657 (0.099) : [2] repair
+ 6.135/0.766 (0.523) : [2] inspect (passed; r=0.209; p=0.180)
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 7.727/1.592 (0.566) : [3] inspect (passed; r=0.462; p=0.360)
+ 9.337/1.610 (0.579) : [4] inspect (passed; r=0.398; p=0.360)
> 205.17 : final bay departure scheduled
[t: 195.918] pid 284/4 arrival event - queue len: 134
> 202.60 (0.451) : scheduled arrival for pid 305/4
* all bays busy; inserting pid 284/4 in queue
[t: 198.146] pid 165/2 departure event (bay 4)
[t: 198.146] handling pid 170/7 in bay 4
+ 2.021/2.021 (0.265) : [1] inspect (passed; r=0.757; p=0.260)
+ 3.946/1.925 (0.582) : [2] inspect (passed; r=0.441; p=0.260)
+ 5.809/1.863 (0.789) : [3] inspect (passed; r=0.765; p=0.260)
> 203.95 : final bay departure scheduled
[t: 198.365] pid 298/2 arrival event - queue len: 134
> 201.75 (0.312) : scheduled arrival for pid 306/2
* all bays busy; inserting pid 298/2 in queue
[t: 199.596] pid 160/3 departure event (bay 3)
[t: 199.596] handling pid 141/6 in bay 3
+ 1.538/1.538 (0.911) : [1] inspect (passed; r=0.756; p=0.180)
+ 2.569/1.031 (0.188) : [2] inspect (passed; r=0.408; p=0.180)
> 202.17 : final bay departure scheduled
[t: 200.288] pid 294/5 arrival event - queue len: 134
> 201.55 (0.891) : scheduled arrival for pid 307/5
* all bays busy; inserting pid 294/5 in queue
[t: 200.628] pid 302/3 arrival event - queue len: 135
> 201.62 (0.760) : scheduled arrival for pid 308/3
* all bays busy; inserting pid 302/3 in queue
[t: 201.552] pid 307/5 arrival event - queue len: 136
> 205.08 (0.723) : scheduled arrival for pid 309/5
* all bays busy; inserting pid 307/5 in queue
[t: 201.617] pid 308/3 arrival event - queue len: 137
> 206.95 (0.228) : scheduled arrival for pid 310/3
* all bays busy; inserting pid 308/3 in queue
[t: 201.747] pid 306/2 arrival event - queue len: 138
> 202.14 (0.872) : scheduled arrival for pid 311/2
* all bays busy; inserting pid 306/2 in queue
[t: 201.948] pid 162/2 departure event (bay 1)
[t: 201.948] handling pid 164/3 in bay 1
+ 1.949/1.949 (0.502) : [1] inspect (passed; r=0.648; p=0.260)
+ 3.920/1.971 (0.432) : [2] inspect (passed; r=0.693; p=0.260)
+ 5.866/1.946 (0.512) : [3] inspect (passed; r=0.350; p=0.260)
> 207.81 : final bay departure scheduled
[t: 202.143] pid 311/2 arrival event - queue len: 138
> 203.56 (0.612) : scheduled arrival for pid 312/2
* all bays busy; inserting pid 311/2 in queue
[t: 202.165] pid 141/6 departure event (bay 3)
[t: 202.165] handling pid 167/2 in bay 3
+ 1.607/1.607 (0.069) : [1] inspect (passed; r=0.939; p=0.140)
+ 3.249/1.642 (0.417) : [2] inspect (failed; r=0.121; p=0.140)
+ 4.413/1.164 (0.658) : [2] repair
+ 5.244/0.831 (0.629) : [2] inspect (passed; r=0.743; p=0.070)
> 207.41 : final bay departure scheduled
[t: 202.602] pid 305/4 arrival event - queue len: 138
> 203.54 (0.895) : scheduled arrival for pid 313/4
* all bays busy; inserting pid 305/4 in queue
[t: 203.440] pid 166/5 departure event (bay 6)
[t: 203.440] handling pid 173/3 in bay 6
+ 1.326/1.326 (0.376) : [1] inspect (passed; r=0.761; p=0.360)
+ 2.465/1.139 (0.242) : [2] inspect (passed; r=0.709; p=0.360)
+ 3.927/1.462 (0.473) : [3] inspect (failed; r=0.267; p=0.360)
+ 6.106/2.180 (0.368) : [3] repair
+ 6.897/0.790 (0.558) : [3] inspect (passed; r=0.676; p=0.180)
+ 8.327/1.430 (0.450) : [4] inspect (passed; r=0.990; p=0.360)
> 211.77 : final bay departure scheduled
[t: 203.535] pid 313/4 arrival event - queue len: 138
> 212.25 (0.354) : scheduled arrival for pid 314/4
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 313/4 in queue
[t: 203.564] pid 312/2 arrival event - queue len: 139
> 205.04 (0.601) : scheduled arrival for pid 315/2
* all bays busy; inserting pid 312/2 in queue
[t: 203.955] pid 170/7 departure event (bay 4)
[t: 203.955] handling pid 172/6 in bay 4
+ 1.945/1.945 (0.446) : [1] inspect (passed; r=0.859; p=0.210)
+ 3.889/1.944 (0.441) : [2] inspect (failed; r=0.047; p=0.210)
+ 6.911/3.022 (0.159) : [2] repair
+ 7.879/0.969 (0.371) : [2] inspect (passed; r=0.178; p=0.105)
+ 9.840/1.961 (0.609) : [3] inspect (failed; r=0.153; p=0.210)
+12.076/2.236 (0.352) : [3] repair
+13.031/0.955 (0.104) : [3] inspect (passed; r=0.731; p=0.105)
> 216.99 : final bay departure scheduled
[t: 204.454] pid 148/7 departure event (bay 5)
[t: 204.454] handling pid 174/2 in bay 5
+ 1.961/1.961 (0.608) : [1] inspect (passed; r=0.763; p=0.210)
+ 3.871/1.910 (0.104) : [2] inspect (passed; r=0.634; p=0.210)
+ 5.853/1.982 (0.820) : [3] inspect (passed; r=0.774; p=0.210)
> 210.31 : final bay departure scheduled
[t: 205.040] pid 315/2 arrival event - queue len: 138
> 205.82 (0.765) : scheduled arrival for pid 316/2
* all bays busy; inserting pid 315/2 in queue
[t: 205.084] pid 309/5 arrival event - queue len: 139
> 208.93 (0.703) : scheduled arrival for pid 317/5
* all bays busy; inserting pid 309/5 in queue
[t: 205.166] pid 168/7 departure event (bay 2)
[t: 205.166] handling pid 171/7 in bay 2
+ 1.983/1.983 (0.827) : [1] inspect (passed; r=0.919; p=0.210)
+ 3.889/1.906 (0.059) : [2] inspect (failed; r=0.112; p=0.210)
+ 5.137/1.248 (0.918) : [2] repair
+ 6.089/0.952 (0.039) : [2] inspect (passed; r=0.947; p=0.105)
+ 7.997/1.908 (0.082) : [3] inspect (passed; r=0.287; p=0.210)
> 213.16 : final bay departure scheduled
[t: 205.817] pid 316/2 arrival event - queue len: 139
> 206.00 (0.940) : scheduled arrival for pid 318/2
* all bays busy; inserting pid 316/2 in queue
[t: 205.998] pid 318/2 arrival event - queue len: 140
> 206.62 (0.806) : scheduled arrival for pid 319/2
* all bays busy; inserting pid 318/2 in queue
[t: 206.623] pid 319/2 arrival event - queue len: 141
> 207.67 (0.697) : scheduled arrival for pid 320/2
* all bays busy; inserting pid 319/2 in queue
[t: 206.946] pid 310/3 arrival event - queue len: 142
> 212.99 (0.187) : scheduled arrival for pid 321/3
* all bays busy; inserting pid 310/3 in queue
[t: 207.410] pid 167/2 departure event (bay 3)
[t: 207.410] handling pid 177/2 in bay 3
+ 1.959/1.959 (0.470) : [1] inspect (passed; r=0.951; p=0.260)
+ 4.052/2.093 (0.022) : [2] inspect (passed; r=0.494; p=0.260)
+ 6.137/2.085 (0.051) : [3] inspect (failed; r=0.063; p=0.260)
+ 7.958/1.821 (0.499) : [3] repair
+ 8.928/0.970 (0.533) : [3] inspect (passed; r=0.568; p=0.130)
> 216.34 : final bay departure scheduled
[t: 207.669] pid 320/2 arrival event - queue len: 142
> 212.16 (0.213) : scheduled arrival for pid 322/2
* all bays busy; inserting pid 320/2 in queue
[t: 207.814] pid 164/3 departure event (bay 1)
[t: 207.814] handling pid 178/7 in bay 1
+ 1.176/1.176 (0.395) : [1] inspect (passed; r=0.183; p=0.180)
+ 2.688/1.511 (0.874) : [2] inspect (passed; r=0.522; p=0.180)
> 210.50 : final bay departure scheduled
[t: 208.932] pid 317/5 arrival event - queue len: 142
> 223.99 (0.251) : scheduled arrival for pid 323/5
* all bays busy; inserting pid 317/5 in queue
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 209.618] pid 304/7 arrival event - queue len: 143
> 211.14 (0.602) : scheduled arrival for pid 324/7
* all bays busy; inserting pid 304/7 in queue
[t: 210.307] pid 174/2 departure event (bay 5)
[t: 210.307] handling pid 158/1 in bay 5
+ 1.813/1.813 (0.958) : [1] inspect (failed; r=0.077; p=0.260)
+ 5.593/3.780 (0.047) : [1] repair
+ 6.533/0.940 (0.733) : [1] inspect (passed; r=0.521; p=0.130)
+ 8.488/1.956 (0.481) : [2] inspect (failed; r=0.187; p=0.260)
+10.150/1.661 (0.247) : [2] repair
+11.186/1.036 (0.094) : [2] inspect (passed; r=0.801; p=0.130)
+13.189/2.003 (0.323) : [3] inspect (passed; r=0.288; p=0.260)
> 223.50 : final bay departure scheduled
[t: 210.502] pid 178/7 departure event (bay 1)
[t: 210.502] handling pid 180/7 in bay 1
+ 1.983/1.983 (0.825) : [1] inspect (passed; r=0.384; p=0.210)
+ 3.964/1.982 (0.817) : [2] inspect (failed; r=0.022; p=0.210)
+ 8.791/4.827 (0.234) : [2] repair
+ 9.751/0.960 (0.198) : [2] inspect (passed; r=0.498; p=0.105)
+11.689/1.938 (0.378) : [3] inspect (passed; r=0.884; p=0.210)
> 222.19 : final bay departure scheduled
[t: 211.138] pid 324/7 arrival event - queue len: 142
> 214.89 (0.287) : scheduled arrival for pid 325/7
* all bays busy; inserting pid 324/7 in queue
[t: 211.767] pid 173/3 departure event (bay 6)
[t: 211.767] handling pid 179/2 in bay 6
+ 1.024/1.024 (0.177) : [1] inspect (passed; r=0.529; p=0.180)
+ 2.600/1.576 (0.966) : [2] inspect (passed; r=0.242; p=0.180)
> 214.37 : final bay departure scheduled
[t: 212.158] pid 322/2 arrival event - queue len: 142
> 214.27 (0.483) : scheduled arrival for pid 326/2
* all bays busy; inserting pid 322/2 in queue
[t: 212.250] pid 314/4 arrival event - queue len: 143
> 228.46 (0.145) : scheduled arrival for pid 327/4
* all bays busy; inserting pid 314/4 in queue
[t: 212.988] pid 321/3 arrival event - queue len: 144
> 215.65 (0.477) : scheduled arrival for pid 328/3
* all bays busy; inserting pid 321/3 in queue
[t: 213.163] pid 171/7 departure event (bay 2)
[t: 213.163] handling pid 169/5 in bay 2
+ 1.919/1.919 (0.189) : [1] inspect (passed; r=0.426; p=0.210)
+ 3.880/1.961 (0.614) : [2] inspect (passed; r=0.599; p=0.210)
+ 5.878/1.998 (0.976) : [3] inspect (passed; r=0.456; p=0.210)
> 219.04 : final bay departure scheduled
[t: 214.272] pid 326/2 arrival event - queue len: 144
> 216.85 (0.412) : scheduled arrival for pid 329/2
* all bays busy; inserting pid 326/2 in queue
[t: 214.367] pid 179/2 departure event (bay 6)
[t: 214.367] handling pid 183/2 in bay 6
+ 1.963/1.963 (0.458) : [1] inspect (passed; r=0.664; p=0.260)
+ 3.810/1.847 (0.842) : [2] inspect (passed; r=0.594; p=0.260)
+ 5.715/1.905 (0.649) : [3] inspect (failed; r=0.086; p=0.260)
+ 7.799/2.084 (0.142) : [3] repair
+ 8.778/0.979 (0.473) : [3] inspect (passed; r=0.478; p=0.130)
> 223.15 : final bay departure scheduled
[t: 214.575] pid 292/1 arrival event - queue len: 144
> 225.21 (0.269) : scheduled arrival for pid 330/1
* all bays busy; inserting pid 292/1 in queue
[t: 214.885] pid 325/7 arrival event - queue len: 145
> 217.69 (0.392) : scheduled arrival for pid 331/7
* all bays busy; inserting pid 325/7 in queue
[t: 215.653] pid 328/3 arrival event - queue len: 146
> 216.63 (0.762) : scheduled arrival for pid 332/3
* all bays busy; inserting pid 328/3 in queue
[t: 216.338] pid 177/2 departure event (bay 3)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 216.338] handling pid 176/6 in bay 3
+ 1.808/1.808 (0.975) : [1] inspect (passed; r=0.334; p=0.260)
+ 3.852/2.044 (0.186) : [2] inspect (failed; r=0.256; p=0.260)
+ 8.222/4.370 (0.036) : [2] repair
+ 9.254/1.032 (0.119) : [2] inspect (failed; r=0.045; p=0.130)
+ 9.937/0.683 (0.406) : [2] repair
+10.854/0.917 (0.887) : [2] inspect (passed; r=0.638; p=0.130)
+12.683/1.829 (0.904) : [3] inspect (failed; r=0.096; p=0.260)
+12.893/0.210 (0.801) : [3] repair
+13.797/0.904 (0.971) : [3] inspect (passed; r=0.797; p=0.130)
> 230.13 : final bay departure scheduled
[t: 216.631] pid 332/3 arrival event - queue len: 146
> 217.50 (0.786) : scheduled arrival for pid 333/3
* all bays busy; inserting pid 332/3 in queue
[t: 216.847] pid 329/2 arrival event - queue len: 147
> 218.18 (0.631) : scheduled arrival for pid 334/2
* all bays busy; inserting pid 329/2 in queue
[t: 216.986] pid 172/6 departure event (bay 4)
[t: 216.986] handling pid 175/3 in bay 4
+ 1.640/1.640 (0.402) : [1] inspect (failed; r=0.083; p=0.140)
+ 4.150/2.510 (0.073) : [1] repair
+ 4.968/0.818 (0.359) : [1] inspect (passed; r=0.331; p=0.070)
+ 6.665/1.697 (0.973) : [2] inspect (passed; r=0.978; p=0.140)
> 223.65 : final bay departure scheduled
[t: 217.500] pid 333/3 arrival event - queue len: 147
> 218.27 (0.807) : scheduled arrival for pid 335/3
* all bays busy; inserting pid 333/3 in queue
[t: 217.695] pid 331/7 arrival event - queue len: 148
> 219.36 (0.573) : scheduled arrival for pid 336/7
* all bays busy; inserting pid 331/7 in queue
[t: 218.180] pid 334/2 arrival event - queue len: 149
> 218.68 (0.843) : scheduled arrival for pid 337/2
* all bays busy; inserting pid 334/2 in queue
[t: 218.273] pid 335/3 arrival event - queue len: 150
> 218.63 (0.906) : scheduled arrival for pid 338/3
* all bays busy; inserting pid 335/3 in queue
[t: 218.629] pid 338/3 arrival event - queue len: 151
> 228.11 (0.072) : scheduled arrival for pid 339/3
* all bays busy; inserting pid 338/3 in queue
[t: 218.675] pid 337/2 arrival event - queue len: 152
> 219.80 (0.678) : scheduled arrival for pid 340/2
* all bays busy; inserting pid 337/2 in queue
[t: 219.041] pid 169/5 departure event (bay 2)
[t: 219.041] handling pid 185/2 in bay 2
+ 1.777/1.777 (0.698) : [1] inspect (failed; r=0.081; p=0.360)
+ 5.330/3.553 (0.073) : [1] repair
+ 6.074/0.744 (0.492) : [1] inspect (failed; r=0.179; p=0.180)
+ 6.355/0.282 (0.687) : [1] repair
+ 7.027/0.672 (0.388) : [1] inspect (passed; r=0.363; p=0.180)
+ 7.922/0.895 (0.068) : [2] inspect (failed; r=0.297; p=0.360)
+10.022/2.100 (0.350) : [2] repair
+10.734/0.712 (0.446) : [2] inspect (passed; r=0.523; p=0.180)
+11.552/0.817 (0.012) : [3] inspect (passed; r=0.414; p=0.360)
+13.590/2.038 (0.884) : [4] inspect (failed; r=0.142; p=0.360)
+14.697/1.107 (0.916) : [4] repair
+15.652/0.955 (0.794) : [4] inspect (passed; r=0.486; p=0.180)
> 234.69 : final bay departure scheduled
[t: 219.364] pid 336/7 arrival event - queue len: 152
> 221.03 (0.575) : scheduled arrival for pid 341/7
* all bays busy; inserting pid 336/7 in queue
[t: 219.804] pid 340/2 arrival event - queue len: 153
> 220.08 (0.911) : scheduled arrival for pid 342/2
* all bays busy; inserting pid 340/2 in queue
[t: 220.076] pid 342/2 arrival event - queue len: 154
> 222.97 (0.368) : scheduled arrival for pid 343/2
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 342/2 in queue
[t: 221.026] pid 341/7 arrival event - queue len: 155
> 227.30 (0.124) : scheduled arrival for pid 344/7
* all bays busy; inserting pid 341/7 in queue
[t: 222.175] pid 303/6 arrival event - queue len: 156
> 223.25 (0.851) : scheduled arrival for pid 345/6
* all bays busy; inserting pid 303/6 in queue
[t: 222.191] pid 180/7 departure event (bay 1)
[t: 222.191] handling pid 186/6 in bay 1
+ 1.988/1.988 (0.880) : [1] inspect (failed; r=0.101; p=0.210)
+ 5.445/3.457 (0.100) : [1] repair
+ 6.435/0.990 (0.809) : [1] inspect (passed; r=0.757; p=0.105)
+ 8.393/1.958 (0.577) : [2] inspect (passed; r=0.932; p=0.210)
+10.313/1.920 (0.205) : [3] inspect (passed; r=0.542; p=0.210)
> 232.50 : final bay departure scheduled
[t: 222.972] pid 343/2 arrival event - queue len: 156
> 223.29 (0.896) : scheduled arrival for pid 346/2
* all bays busy; inserting pid 343/2 in queue
[t: 223.145] pid 183/2 departure event (bay 6)
[t: 223.145] handling pid 182/7 in bay 6
+ 2.027/2.027 (0.243) : [1] inspect (failed; r=0.130; p=0.260)
+ 5.568/3.541 (0.430) : [1] repair
+ 6.538/0.970 (0.532) : [1] inspect (passed; r=0.179; p=0.130)
+ 8.583/2.045 (0.183) : [2] inspect (failed; r=0.144; p=0.260)
+ 9.805/1.221 (0.673) : [2] repair
+10.756/0.951 (0.662) : [2] inspect (passed; r=0.515; p=0.130)
+12.770/2.015 (0.284) : [3] inspect (passed; r=0.732; p=0.260)
> 235.92 : final bay departure scheduled
[t: 223.255] pid 345/6 arrival event - queue len: 156
> 237.79 (0.114) : scheduled arrival for pid 347/6
* all bays busy; inserting pid 345/6 in queue
[t: 223.290] pid 346/2 arrival event - queue len: 157
> 224.63 (0.630) : scheduled arrival for pid 348/2
* all bays busy; inserting pid 346/2 in queue
[t: 223.496] pid 158/1 departure event (bay 5)
[t: 223.496] handling pid 188/2 in bay 5
+ 1.404/1.404 (0.503) : [1] inspect (passed; r=0.555; p=0.300)
+ 2.222/0.819 (0.085) : [2] inspect (passed; r=0.734; p=0.300)
+ 3.361/1.139 (0.313) : [3] inspect (passed; r=0.723; p=0.300)
+ 4.422/1.060 (0.257) : [4] inspect (passed; r=0.862; p=0.300)
> 227.92 : final bay departure scheduled
[t: 223.651] pid 175/3 departure event (bay 4)
[t: 223.651] handling pid 191/2 in bay 4
+ 1.160/1.160 (0.372) : [1] inspect (passed; r=0.257; p=0.180)
+ 2.480/1.320 (0.600) : [2] inspect (failed; r=0.094; p=0.180)
+ 4.572/2.092 (0.430) : [2] repair
+ 5.032/0.460 (0.028) : [2] inspect (passed; r=0.128; p=0.090)
> 228.68 : final bay departure scheduled
[t: 223.994] pid 323/5 arrival event - queue len: 156
> 232.34 (0.465) : scheduled arrival for pid 349/5
* all bays busy; inserting pid 323/5 in queue
[t: 224.631] pid 348/2 arrival event - queue len: 157
> 225.61 (0.713) : scheduled arrival for pid 350/2
* all bays busy; inserting pid 348/2 in queue
[t: 225.207] pid 330/1 arrival event - queue len: 158
> 240.24 (0.156) : scheduled arrival for pid 351/1
* all bays busy; inserting pid 330/1 in queue
[t: 225.614] pid 350/2 arrival event - queue len: 159
> 225.71 (0.968) : scheduled arrival for pid 352/2
* all bays busy; inserting pid 350/2 in queue
[t: 225.708] pid 352/2 arrival event - queue len: 160
> 228.56 (0.374) : scheduled arrival for pid 353/2
* all bays busy; inserting pid 352/2 in queue
[t: 227.300] pid 344/7 arrival event - queue len: 161
> 232.27 (0.191) : scheduled arrival for pid 354/7
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 344/7 in queue
[t: 227.918] pid 188/2 departure event (bay 5)
[t: 227.918] handling pid 163/4 in bay 5
+ 1.960/1.960 (0.466) : [1] inspect (passed; r=0.378; p=0.260)
+ 3.883/1.923 (0.590) : [2] inspect (passed; r=0.432; p=0.260)
+ 5.719/1.836 (0.879) : [3] inspect (passed; r=0.510; p=0.260)
> 233.64 : final bay departure scheduled
[t: 228.106] pid 339/3 arrival event - queue len: 161
> 233.83 (0.204) : scheduled arrival for pid 355/3
* all bays busy; inserting pid 339/3 in queue
[t: 228.460] pid 327/4 arrival event - queue len: 162
> 228.56 (0.988) : scheduled arrival for pid 356/4
* all bays busy; inserting pid 327/4 in queue
[t: 228.558] pid 353/2 arrival event - queue len: 163
> 231.90 (0.316) : scheduled arrival for pid 357/2
* all bays busy; inserting pid 353/2 in queue
[t: 228.559] pid 356/4 arrival event - queue len: 164
> 229.99 (0.843) : scheduled arrival for pid 358/4
* all bays busy; inserting pid 356/4 in queue
[t: 228.683] pid 191/2 departure event (bay 4)
[t: 228.683] handling pid 189/6 in bay 4
+ 1.840/1.840 (0.865) : [1] inspect (passed; r=0.302; p=0.260)
+ 3.700/1.860 (0.801) : [2] inspect (passed; r=0.467; p=0.260)
+ 5.732/2.032 (0.227) : [3] inspect (failed; r=0.074; p=0.260)
+ 6.571/0.839 (0.556) : [3] repair
+ 7.492/0.921 (0.860) : [3] inspect (passed; r=0.151; p=0.130)
> 236.18 : final bay departure scheduled
[t: 229.995] pid 358/4 arrival event - queue len: 164
> 238.43 (0.366) : scheduled arrival for pid 359/4
* all bays busy; inserting pid 358/4 in queue
[t: 230.135] pid 176/6 departure event (bay 3)
[t: 230.135] handling pid 190/7 in bay 3
+ 1.684/1.684 (0.842) : [1] inspect (passed; r=0.951; p=0.140)
+ 3.288/1.604 (0.040) : [2] inspect (passed; r=0.304; p=0.140)
> 233.42 : final bay departure scheduled
[t: 231.898] pid 357/2 arrival event - queue len: 164
> 231.91 (0.996) : scheduled arrival for pid 360/2
* all bays busy; inserting pid 357/2 in queue
[t: 231.910] pid 360/2 arrival event - queue len: 165
> 232.74 (0.752) : scheduled arrival for pid 361/2
* all bays busy; inserting pid 360/2 in queue
[t: 232.267] pid 354/7 arrival event - queue len: 166
> 233.99 (0.563) : scheduled arrival for pid 362/7
* all bays busy; inserting pid 354/7 in queue
[t: 232.336] pid 349/5 arrival event - queue len: 167
> 232.70 (0.967) : scheduled arrival for pid 363/5
* all bays busy; inserting pid 349/5 in queue
[t: 232.504] pid 186/6 departure event (bay 1)
[t: 232.504] handling pid 195/7 in bay 1
+ 1.611/1.611 (0.111) : [1] inspect (passed; r=0.818; p=0.140)
+ 3.239/1.628 (0.284) : [2] inspect (passed; r=0.960; p=0.140)
> 235.74 : final bay departure scheduled
[t: 232.697] pid 363/5 arrival event - queue len: 167
> 237.71 (0.631) : scheduled arrival for pid 364/5
* all bays busy; inserting pid 363/5 in queue
[t: 232.736] pid 361/2 arrival event - queue len: 168
> 239.71 (0.090) : scheduled arrival for pid 365/2
* all bays busy; inserting pid 361/2 in queue
[t: 233.423] pid 190/7 departure event (bay 3)
[t: 233.423] handling pid 192/2 in bay 3
+ 1.956/1.956 (0.555) : [1] inspect (passed; r=0.978; p=0.210)
+ 3.915/1.959 (0.595) : [2] inspect (passed; r=0.509; p=0.210)
+ 5.851/1.936 (0.361) : [3] inspect (failed; r=0.004; p=0.210)
+ 9.536/3.685 (0.196) : [3] repair
+10.530/0.994 (0.870) : [3] inspect (passed; r=0.285; p=0.105)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 243.95 : final bay departure scheduled
[t: 233.637] pid 163/4 departure event (bay 5)
[t: 233.637] handling pid 194/6 in bay 5
+ 2.201/2.201 (0.401) : [1] inspect (passed; r=0.999; p=0.120)
+ 4.445/2.244 (0.444) : [2] inspect (passed; r=0.972; p=0.120)
+ 6.516/2.071 (0.271) : [3] inspect (passed; r=0.279; p=0.120)
+ 8.332/1.816 (0.016) : [4] inspect (passed; r=0.496; p=0.120)
> 241.97 : final bay departure scheduled
[t: 233.825] pid 355/3 arrival event - queue len: 167
> 235.32 (0.661) : scheduled arrival for pid 366/3
* all bays busy; inserting pid 355/3 in queue
[t: 233.988] pid 362/7 arrival event - queue len: 168
> 234.02 (0.991) : scheduled arrival for pid 367/7
* all bays busy; inserting pid 362/7 in queue
[t: 234.016] pid 367/7 arrival event - queue len: 169
> 234.07 (0.982) : scheduled arrival for pid 368/7
* all bays busy; inserting pid 367/7 in queue
[t: 234.070] pid 368/7 arrival event - queue len: 170
> 238.22 (0.250) : scheduled arrival for pid 369/7
* all bays busy; inserting pid 368/7 in queue
[t: 234.693] pid 185/2 departure event (bay 2)
[t: 234.693] handling pid 187/3 in bay 2
+ 1.957/1.957 (0.477) : [1] inspect (passed; r=0.706; p=0.260)
+ 3.918/1.961 (0.463) : [2] inspect (passed; r=0.322; p=0.260)
+ 5.788/1.871 (0.765) : [3] inspect (passed; r=0.485; p=0.260)
> 240.48 : final bay departure scheduled
[t: 235.316] pid 366/3 arrival event - queue len: 170
> 242.55 (0.134) : scheduled arrival for pid 370/3
* all bays busy; inserting pid 366/3 in queue
[t: 235.744] pid 195/7 departure event (bay 1)
[t: 235.744] handling pid 198/6 in bay 1
+ 1.959/1.959 (0.594) : [1] inspect (failed; r=0.189; p=0.210)
+ 8.111/6.152 (0.033) : [1] repair
+ 9.074/0.963 (0.251) : [1] inspect (failed; r=0.023; p=0.105)
+11.663/2.589 (0.047) : [1] repair
+12.621/0.959 (0.173) : [1] inspect (passed; r=0.848; p=0.105)
+14.585/1.963 (0.632) : [2] inspect (passed; r=0.490; p=0.210)
+16.548/1.964 (0.639) : [3] inspect (passed; r=0.843; p=0.210)
> 252.29 : final bay departure scheduled
[t: 235.916] pid 182/7 departure event (bay 6)
[t: 235.916] handling pid 196/7 in bay 6
+ 1.936/1.936 (0.356) : [1] inspect (passed; r=0.606; p=0.210)
+ 3.851/1.915 (0.151) : [2] inspect (failed; r=0.134; p=0.210)
+ 8.585/4.735 (0.515) : [2] repair
+ 9.583/0.998 (0.957) : [2] inspect (passed; r=0.719; p=0.105)
+11.526/1.943 (0.428) : [3] inspect (passed; r=0.920; p=0.210)
> 247.44 : final bay departure scheduled
[t: 236.175] pid 189/6 departure event (bay 4)
[t: 236.175] handling pid 197/2 in bay 4
+ 1.614/1.614 (0.142) : [1] inspect (passed; r=0.965; p=0.140)
+ 3.247/1.633 (0.331) : [2] inspect (passed; r=0.213; p=0.140)
> 239.42 : final bay departure scheduled
[t: 237.709] pid 364/5 arrival event - queue len: 168
> 323.11 (0.000) : scheduled arrival for pid 371/5
* all bays busy; inserting pid 364/5 in queue
[t: 237.791] pid 347/6 arrival event - queue len: 169
> 248.05 (0.216) : scheduled arrival for pid 372/6
* all bays busy; inserting pid 347/6 in queue
[t: 238.224] pid 369/7 arrival event - queue len: 170
> 240.43 (0.479) : scheduled arrival for pid 373/7
* all bays busy; inserting pid 369/7 in queue
[t: 238.432] pid 359/4 arrival event - queue len: 171
> 241.10 (0.728) : scheduled arrival for pid 374/4
* all bays busy; inserting pid 359/4 in queue
[t: 239.422] pid 197/2 departure event (bay 4)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 239.422] handling pid 201/7 in bay 4
+ 1.910/1.910 (0.635) : [1] inspect (passed; r=0.551; p=0.260)
+ 3.882/1.973 (0.425) : [2] inspect (failed; r=0.005; p=0.260)
+ 4.356/0.474 (0.991) : [2] repair
+ 5.303/0.947 (0.684) : [2] inspect (passed; r=0.963; p=0.130)
+ 7.335/2.032 (0.228) : [3] inspect (failed; r=0.082; p=0.260)
+ 7.896/0.561 (0.671) : [3] repair
+ 8.933/1.037 (0.086) : [3] inspect (passed; r=0.354; p=0.130)
> 248.36 : final bay departure scheduled
[t: 239.714] pid 365/2 arrival event - queue len: 171
> 242.10 (0.439) : scheduled arrival for pid 375/2
* all bays busy; inserting pid 365/2 in queue
[t: 240.236] pid 351/1 arrival event - queue len: 172
> 263.77 (0.055) : scheduled arrival for pid 376/1
* all bays busy; inserting pid 351/1 in queue
[t: 240.434] pid 373/7 arrival event - queue len: 173
> 240.46 (0.991) : scheduled arrival for pid 377/7
* all bays busy; inserting pid 373/7 in queue
[t: 240.462] pid 377/7 arrival event - queue len: 174
> 242.52 (0.503) : scheduled arrival for pid 378/7
* all bays busy; inserting pid 377/7 in queue
[t: 240.481] pid 187/3 departure event (bay 2)
[t: 240.481] handling pid 199/3 in bay 2
+ 1.124/1.124 (0.320) : [1] inspect (failed; r=0.131; p=0.180)
+ 1.981/0.857 (0.671) : [1] repair
+ 2.575/0.594 (0.410) : [1] inspect (failed; r=0.038; p=0.090)
+ 3.480/0.905 (0.328) : [1] repair
+ 4.133/0.653 (0.580) : [1] inspect (passed; r=0.318; p=0.090)
+ 5.513/1.380 (0.686) : [2] inspect (passed; r=0.277; p=0.180)
> 245.99 : final bay departure scheduled
[t: 241.104] pid 374/4 arrival event - queue len: 174
> 259.66 (0.110) : scheduled arrival for pid 379/4
* all bays busy; inserting pid 374/4 in queue
[t: 241.969] pid 194/6 departure event (bay 5)
[t: 241.969] handling pid 181/1 in bay 5
+ 1.672/1.672 (0.723) : [1] inspect (passed; r=0.161; p=0.140)
+ 3.294/1.622 (0.218) : [2] inspect (passed; r=0.517; p=0.140)
> 245.26 : final bay departure scheduled
[t: 242.104] pid 375/2 arrival event - queue len: 174
> 243.75 (0.567) : scheduled arrival for pid 380/2
* all bays busy; inserting pid 375/2 in queue
[t: 242.523] pid 378/7 arrival event - queue len: 175
> 248.70 (0.127) : scheduled arrival for pid 381/7
* all bays busy; inserting pid 378/7 in queue
[t: 242.554] pid 370/3 arrival event - queue len: 176
> 243.85 (0.699) : scheduled arrival for pid 382/3
* all bays busy; inserting pid 370/3 in queue
[t: 243.749] pid 380/2 arrival event - queue len: 177
> 244.54 (0.761) : scheduled arrival for pid 383/2
* all bays busy; inserting pid 380/2 in queue
[t: 243.845] pid 382/3 arrival event - queue len: 178
> 244.31 (0.879) : scheduled arrival for pid 384/3
* all bays busy; inserting pid 382/3 in queue
[t: 243.953] pid 192/2 departure event (bay 3)
[t: 243.953] handling pid 184/5 in bay 3
+ 1.849/1.849 (0.837) : [1] inspect (passed; r=0.777; p=0.260)
+ 3.764/1.915 (0.616) : [2] inspect (passed; r=0.949; p=0.260)
+ 5.627/1.863 (0.788) : [3] inspect (passed; r=0.732; p=0.260)
> 249.58 : final bay departure scheduled
[t: 244.308] pid 384/3 arrival event - queue len: 178
> 244.93 (0.842) : scheduled arrival for pid 385/3
* all bays busy; inserting pid 384/3 in queue
[t: 244.541] pid 383/2 arrival event - queue len: 179
> 246.78 (0.462) : scheduled arrival for pid 386/2
* all bays busy; inserting pid 383/2 in queue
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 244.927] pid 385/3 arrival event - queue len: 180
> 251.17 (0.176) : scheduled arrival for pid 387/3
* all bays busy; inserting pid 385/3 in queue
[t: 245.264] pid 181/1 departure event (bay 5)
[t: 245.264] handling pid 205/1 in bay 5
+ 1.166/1.166 (0.333) : [1] inspect (passed; r=0.664; p=0.300)
+ 2.554/1.388 (0.491) : [2] inspect (failed; r=0.098; p=0.300)
+ 4.231/1.678 (0.566) : [2] repair
+ 4.957/0.726 (0.537) : [2] inspect (passed; r=0.568; p=0.150)
+ 6.972/2.015 (0.939) : [3] inspect (failed; r=0.155; p=0.300)
+ 9.935/2.963 (0.233) : [3] repair
+10.924/0.989 (0.913) : [3] inspect (passed; r=0.668; p=0.150)
+12.169/1.245 (0.389) : [4] inspect (passed; r=0.715; p=0.300)
> 257.43 : final bay departure scheduled
[t: 245.994] pid 199/3 departure event (bay 2)
[t: 245.994] handling pid 200/6 in bay 2
+ 0.999/0.999 (0.141) : [1] inspect (passed; r=0.236; p=0.180)
+ 2.377/1.378 (0.683) : [2] inspect (passed; r=0.811; p=0.180)
> 248.37 : final bay departure scheduled
[t: 246.781] pid 386/2 arrival event - queue len: 179
> 247.04 (0.916) : scheduled arrival for pid 388/2
* all bays busy; inserting pid 386/2 in queue
[t: 247.036] pid 388/2 arrival event - queue len: 180
> 253.58 (0.105) : scheduled arrival for pid 389/2
* all bays busy; inserting pid 388/2 in queue
[t: 247.442] pid 196/7 departure event (bay 6)
[t: 247.442] handling pid 208/6 in bay 6
+ 1.937/1.937 (0.369) : [1] inspect (passed; r=0.225; p=0.210)
+ 3.933/1.996 (0.959) : [2] inspect (failed; r=0.153; p=0.210)
+ 6.158/2.225 (0.622) : [2] repair
+ 7.127/0.969 (0.371) : [2] inspect (passed; r=0.924; p=0.105)
+ 9.039/1.912 (0.117) : [3] inspect (passed; r=0.595; p=0.210)
> 256.48 : final bay departure scheduled
[t: 248.052] pid 372/6 arrival event - queue len: 180
> 250.30 (0.715) : scheduled arrival for pid 390/6
* all bays busy; inserting pid 372/6 in queue
[t: 248.356] pid 201/7 departure event (bay 4)
[t: 248.356] handling pid 202/2 in bay 4
+ 1.910/1.910 (0.105) : [1] inspect (passed; r=0.681; p=0.210)
+ 3.840/1.929 (0.294) : [2] inspect (passed; r=0.892; p=0.210)
+ 5.801/1.962 (0.616) : [3] inspect (failed; r=0.081; p=0.210)
+ 8.344/2.543 (0.443) : [3] repair
+ 9.326/0.982 (0.641) : [3] inspect (passed; r=0.618; p=0.105)
> 257.68 : final bay departure scheduled
[t: 248.371] pid 200/6 departure event (bay 2)
[t: 248.371] handling pid 204/3 in bay 2
+ 1.693/1.693 (0.931) : [1] inspect (passed; r=0.528; p=0.140)
+ 3.372/1.679 (0.793) : [2] inspect (passed; r=0.731; p=0.140)
> 251.74 : final bay departure scheduled
[t: 248.703] pid 381/7 arrival event - queue len: 179
> 250.27 (0.594) : scheduled arrival for pid 391/7
* all bays busy; inserting pid 381/7 in queue
[t: 249.580] pid 184/5 departure event (bay 3)
[t: 249.580] handling pid 203/7 in bay 3
+ 1.399/1.399 (0.428) : [1] inspect (failed; r=0.028; p=0.360)
+ 3.816/2.417 (0.144) : [1] repair
+ 4.805/0.989 (0.841) : [1] inspect (passed; r=0.757; p=0.180)
+ 6.455/1.650 (0.607) : [2] inspect (passed; r=0.763; p=0.360)
+ 8.280/1.825 (0.732) : [3] inspect (passed; r=0.938; p=0.360)
+ 9.302/1.022 (0.159) : [4] inspect (passed; r=0.579; p=0.360)
> 258.88 : final bay departure scheduled
[t: 250.267] pid 391/7 arrival event - queue len: 179
> 257.70 (0.084) : scheduled arrival for pid 392/7
* all bays busy; inserting pid 391/7 in queue
[t: 250.299] pid 390/6 arrival event - queue len: 180
```

## APPENDIX C: The Application Trace File (ATF)

```
> 263.40 (0.142) : scheduled arrival for pid 393/6
* all bays busy; inserting pid 390/6 in queue
[t: 251.172] pid 387/3 arrival event - queue len: 181
> 261.31 (0.060) : scheduled arrival for pid 394/3
* all bays busy; inserting pid 387/3 in queue
[t: 251.744] pid 204/3 departure event (bay 2)
[t: 251.744] handling pid 209/6 in bay 2
+ 1.241/1.241 (0.487) : [1] inspect (passed; r=0.964; p=0.180)
+ 2.348/1.107 (0.296) : [2] inspect (passed; r=0.397; p=0.180)
> 254.09 : final bay departure scheduled
[t: 252.292] pid 198/6 departure event (bay 1)
[t: 252.292] handling pid 213/6 in bay 1
+ 1.698/1.698 (0.983) : [1] inspect (passed; r=0.931; p=0.140)
+ 3.320/1.622 (0.215) : [2] inspect (failed; r=0.088; p=0.140)
+ 4.800/1.480 (0.660) : [2] repair
+ 5.600/0.800 (0.000) : [2] inspect (passed; r=0.760; p=0.070)
> 257.89 : final bay departure scheduled
[t: 253.575] pid 389/2 arrival event - queue len: 180
> 255.60 (0.497) : scheduled arrival for pid 395/2
* all bays busy; inserting pid 389/2 in queue
[t: 254.092] pid 209/6 departure event (bay 2)
[t: 254.092] handling pid 212/7 in bay 2
+ 1.689/1.689 (0.891) : [1] inspect (passed; r=0.778; p=0.140)
+ 3.354/1.664 (0.644) : [2] inspect (passed; r=0.190; p=0.140)
> 257.45 : final bay departure scheduled
[t: 255.604] pid 395/2 arrival event - queue len: 180
> 261.83 (0.117) : scheduled arrival for pid 396/2
* all bays busy; inserting pid 395/2 in queue
[t: 256.480] pid 208/6 departure event (bay 6)
[t: 256.480] handling pid 215/7 in bay 6
+ 1.622/1.622 (0.225) : [1] inspect (passed; r=0.750; p=0.140)
+ 3.250/1.628 (0.276) : [2] inspect (passed; r=0.506; p=0.140)
> 259.73 : final bay departure scheduled
[t: 257.433] pid 205/1 departure event (bay 5)
[t: 257.433] handling pid 193/4 in bay 5
+ 0.869/0.869 (0.121) : [1] inspect (failed; r=0.166; p=0.300)
+ 1.231/0.361 (0.951) : [1] repair
+ 2.151/0.921 (0.815) : [1] inspect (passed; r=0.893; p=0.150)
+ 3.592/1.441 (0.529) : [2] inspect (passed; r=0.469; p=0.300)
+ 4.829/1.237 (0.384) : [3] inspect (passed; r=0.750; p=0.300)
+ 6.480/1.651 (0.679) : [4] inspect (passed; r=0.945; p=0.300)
> 263.91 : final bay departure scheduled
[t: 257.446] pid 212/7 departure event (bay 2)
[t: 257.446] handling pid 206/5 in bay 2
+ 1.935/1.935 (0.345) : [1] inspect (passed; r=0.832; p=0.210)
+ 3.928/1.994 (0.938) : [2] inspect (passed; r=0.496; p=0.210)
+ 5.850/1.921 (0.212) : [3] inspect (passed; r=0.551; p=0.210)
> 263.30 : final bay departure scheduled
[t: 257.682] pid 202/2 departure event (bay 4)
[t: 257.682] handling pid 210/2 in bay 4
+ 1.959/1.959 (0.471) : [1] inspect (passed; r=0.743; p=0.260)
+ 3.969/2.010 (0.299) : [2] inspect (failed; r=0.039; p=0.260)
+ 4.550/0.581 (0.874) : [2] repair
+ 5.455/0.905 (0.967) : [2] inspect (passed; r=0.243; p=0.130)
+ 7.552/2.097 (0.008) : [3] inspect (passed; r=0.667; p=0.260)
> 265.23 : final bay departure scheduled
[t: 257.705] pid 392/7 arrival event - queue len: 177
> 261.40 (0.292) : scheduled arrival for pid 397/7
* all bays busy; inserting pid 392/7 in queue
[t: 257.892] pid 213/6 departure event (bay 1)
[t: 257.892] handling pid 216/7 in bay 1
+ 1.635/1.635 (0.347) : [1] inspect (failed; r=0.113; p=0.140)
+ 4.369/2.734 (0.099) : [1] repair
+ 5.215/0.846 (0.924) : [1] inspect (passed; r=0.100; p=0.070)
+ 6.901/1.686 (0.859) : [2] inspect (passed; r=0.505; p=0.140)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 264.79 : final bay departure scheduled
[t: 258.883] pid 203/7 departure event (bay 3)
[t: 258.883] handling pid 211/3 in bay 3
+ 1.923/1.923 (0.230) : [1] inspect (passed; r=0.479; p=0.210)
+ 3.908/1.985 (0.850) : [2] inspect (failed; r=0.053; p=0.210)
+ 9.158/5.250 (0.442) : [2] repair
+10.135/0.977 (0.539) : [2] inspect (passed; r=0.412; p=0.105)
+12.125/1.990 (0.895) : [3] inspect (passed; r=0.739; p=0.210)
> 271.01 : final bay departure scheduled
[t: 259.663] pid 379/4 arrival event - queue len: 176
> 267.51 (0.393) : scheduled arrival for pid 398/4
* all bays busy; inserting pid 379/4 in queue
[t: 259.730] pid 215/7 departure event (bay 6)
[t: 259.730] handling pid 221/3 in bay 6
+ 1.978/1.978 (0.785) : [1] inspect (failed; r=0.171; p=0.210)
+ 4.320/2.342 (0.630) : [1] repair
+ 5.287/0.966 (0.329) : [1] inspect (failed; r=0.051; p=0.105)
+ 7.532/2.246 (0.724) : [1] repair
+ 8.504/0.971 (0.426) : [1] inspect (passed; r=0.654; p=0.105)
+10.425/1.921 (0.214) : [2] inspect (passed; r=0.448; p=0.210)
+12.402/1.977 (0.769) : [3] inspect (passed; r=0.558; p=0.210)
> 272.13 : final bay departure scheduled
[t: 261.309] pid 394/3 arrival event - queue len: 176
> 261.98 (0.831) : scheduled arrival for pid 399/3
* all bays busy; inserting pid 394/3 in queue
[t: 261.397] pid 397/7 arrival event - queue len: 177
> 261.67 (0.913) : scheduled arrival for pid 400/7
* all bays busy; inserting pid 397/7 in queue
[t: 261.671] pid 400/7 arrival event - queue len: 178
> 265.90 (0.244) : scheduled arrival for pid 401/7
* all bays busy; inserting pid 400/7 in queue
[t: 261.829] pid 396/2 arrival event - queue len: 179
> 263.37 (0.589) : scheduled arrival for pid 402/2
* all bays busy; inserting pid 396/2 in queue
[t: 261.976] pid 399/3 arrival event - queue len: 180
> 266.90 (0.255) : scheduled arrival for pid 403/3
* all bays busy; inserting pid 399/3 in queue
[t: 263.295] pid 206/5 departure event (bay 2)
[t: 263.295] handling pid 222/3 in bay 2
+ 1.425/1.425 (0.447) : [1] inspect (failed; r=0.098; p=0.360)
+ 1.812/0.387 (0.736) : [1] repair
+ 2.281/0.469 (0.099) : [1] inspect (passed; r=0.303; p=0.180)
+ 4.046/1.764 (0.689) : [2] inspect (failed; r=0.136; p=0.360)
+ 7.853/3.807 (0.989) : [2] repair
+ 8.682/0.829 (0.613) : [2] inspect (passed; r=0.463; p=0.180)
+10.486/1.804 (0.717) : [3] inspect (passed; r=0.887; p=0.360)
+12.455/1.969 (0.835) : [4] inspect (failed; r=0.322; p=0.360)
+18.607/6.152 (0.137) : [4] repair
+19.421/0.814 (0.592) : [4] inspect (passed; r=0.838; p=0.180)
> 282.72 : final bay departure scheduled
[t: 263.366] pid 402/2 arrival event - queue len: 180
> 266.34 (0.359) : scheduled arrival for pid 404/2
* all bays busy; inserting pid 402/2 in queue
[t: 263.396] pid 393/6 arrival event - queue len: 181
> 265.73 (0.705) : scheduled arrival for pid 405/6
* all bays busy; inserting pid 393/6 in queue
[t: 263.766] pid 376/1 arrival event - queue len: 182
> 266.62 (0.703) : scheduled arrival for pid 406/1
* all bays busy; inserting pid 376/1 in queue
[t: 263.913] pid 193/4 departure event (bay 5)
[t: 263.913] handling pid 220/7 in bay 5
+ 2.136/2.136 (0.336) : [1] inspect (passed; r=0.818; p=0.120)
+ 4.369/2.232 (0.432) : [2] inspect (passed; r=0.960; p=0.120)
+ 6.882/2.513 (0.713) : [3] inspect (passed; r=0.440; p=0.120)
+ 9.130/2.248 (0.448) : [4] inspect (passed; r=0.278; p=0.120)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 273.04 : final bay departure scheduled
[t: 264.793] pid 216/7 departure event (bay 1)
[t: 264.793] handling pid 214/6 in bay 1
+ 1.969/1.969 (0.694) : [1] inspect (passed; r=0.263; p=0.210)
+ 3.947/1.978 (0.778) : [2] inspect (passed; r=0.958; p=0.210)
+ 5.888/1.941 (0.405) : [3] inspect (passed; r=0.995; p=0.210)
> 270.68 : final bay departure scheduled
[t: 265.234] pid 210/2 departure event (bay 4)
[t: 265.234] handling pid 225/6 in bay 4
+ 2.015/2.015 (0.285) : [1] inspect (passed; r=0.416; p=0.260)
+ 3.863/1.849 (0.837) : [2] inspect (failed; r=0.130; p=0.260)
+ 4.842/0.978 (0.519) : [2] repair
+ 5.852/1.010 (0.268) : [2] inspect (failed; r=0.102; p=0.130)
+ 6.895/1.044 (0.589) : [2] repair
+ 7.937/1.042 (0.055) : [2] inspect (passed; r=0.351; p=0.130)
+ 9.823/1.886 (0.713) : [3] inspect (failed; r=0.019; p=0.260)
+10.335/0.511 (0.609) : [3] repair
+11.298/0.963 (0.577) : [3] inspect (passed; r=0.496; p=0.130)
> 276.53 : final bay departure scheduled
[t: 265.734] pid 405/6 arrival event - queue len: 180
> 267.27 (0.795) : scheduled arrival for pid 407/6
* all bays busy; inserting pid 405/6 in queue
[t: 265.899] pid 401/7 arrival event - queue len: 181
> 276.33 (0.031) : scheduled arrival for pid 408/7
* all bays busy; inserting pid 401/7 in queue
[t: 266.338] pid 404/2 arrival event - queue len: 182
> 272.10 (0.137) : scheduled arrival for pid 409/2
* all bays busy; inserting pid 404/2 in queue
[t: 266.621] pid 406/1 arrival event - queue len: 183
> 276.14 (0.309) : scheduled arrival for pid 410/1
* all bays busy; inserting pid 406/1 in queue
[t: 266.896] pid 403/3 arrival event - queue len: 184
> 267.81 (0.775) : scheduled arrival for pid 411/3
* all bays busy; inserting pid 403/3 in queue
[t: 267.275] pid 407/6 arrival event - queue len: 185
> 269.20 (0.750) : scheduled arrival for pid 412/6
* all bays busy; inserting pid 407/6 in queue
[t: 267.511] pid 398/4 arrival event - queue len: 186
> 271.42 (0.628) : scheduled arrival for pid 413/4
* all bays busy; inserting pid 398/4 in queue
[t: 267.812] pid 411/3 arrival event - queue len: 187
> 275.25 (0.127) : scheduled arrival for pid 414/3
* all bays busy; inserting pid 411/3 in queue
[t: 269.203] pid 412/6 arrival event - queue len: 188
> 292.19 (0.032) : scheduled arrival for pid 415/6
* all bays busy; inserting pid 412/6 in queue
[t: 270.681] pid 214/6 departure event (bay 1)
[t: 270.681] handling pid 207/1 in bay 1
+ 1.639/1.639 (0.390) : [1] inspect (passed; r=0.431; p=0.140)
+ 3.306/1.667 (0.673) : [2] inspect (passed; r=0.856; p=0.140)
> 273.99 : final bay departure scheduled
[t: 271.008] pid 211/3 departure event (bay 3)
[t: 271.008] handling pid 224/7 in bay 3
+ 1.165/1.165 (0.378) : [1] inspect (passed; r=0.208; p=0.180)
+ 2.213/1.048 (0.212) : [2] inspect (passed; r=0.768; p=0.180)
> 273.22 : final bay departure scheduled
[t: 271.417] pid 413/4 arrival event - queue len: 187
> 274.68 (0.678) : scheduled arrival for pid 416/4
* all bays busy; inserting pid 413/4 in queue
[t: 272.097] pid 409/2 arrival event - queue len: 188
> 276.59 (0.212) : scheduled arrival for pid 417/2
* all bays busy; inserting pid 409/2 in queue
[t: 272.133] pid 221/3 departure event (bay 6)
[t: 272.133] handling pid 223/3 in bay 6
+ 1.108/1.108 (0.297) : [1] inspect (passed; r=0.356; p=0.180)
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 2.152/1.045 (0.207) : [2] inspect (passed; r=0.880; p=0.180)
> 274.28 : final bay departure scheduled
[t: 273.042] pid 220/7 departure event (bay 5)
[t: 273.042] handling pid 229/3 in bay 5
+ 1.937/1.937 (0.368) : [1] inspect (passed; r=0.426; p=0.210)
+ 3.891/1.955 (0.546) : [2] inspect (passed; r=0.649; p=0.210)
+ 5.856/1.965 (0.648) : [3] inspect (passed; r=0.515; p=0.210)
> 278.90 : final bay departure scheduled
[t: 273.221] pid 224/7 departure event (bay 3)
[t: 273.221] handling pid 226/6 in bay 3
+ 1.934/1.934 (0.336) : [1] inspect (passed; r=0.861; p=0.210)
+ 3.862/1.929 (0.288) : [2] inspect (passed; r=0.372; p=0.210)
+ 5.822/1.960 (0.597) : [3] inspect (passed; r=0.258; p=0.210)
> 279.04 : final bay departure scheduled
[t: 273.987] pid 207/1 departure event (bay 1)
[t: 273.987] handling pid 217/4 in bay 1
+ 1.406/1.406 (0.504) : [1] inspect (passed; r=0.556; p=0.300)
+ 2.659/1.253 (0.395) : [2] inspect (passed; r=0.334; p=0.300)
+ 4.684/2.025 (0.946) : [3] inspect (passed; r=0.716; p=0.300)
+ 5.469/0.786 (0.061) : [4] inspect (failed; r=0.299; p=0.300)
+ 6.021/0.552 (0.833) : [4] repair
+ 6.493/0.472 (0.174) : [4] inspect (passed; r=0.665; p=0.150)
> 280.48 : final bay departure scheduled
[t: 274.285] pid 223/3 departure event (bay 6)
[t: 274.285] handling pid 219/2 in bay 6
+ 1.039/1.039 (0.198) : [1] inspect (failed; r=0.120; p=0.180)
+ 3.960/2.922 (0.143) : [1] repair
+ 4.527/0.566 (0.332) : [1] inspect (passed; r=0.997; p=0.090)
+ 5.665/1.138 (0.340) : [2] inspect (passed; r=0.246; p=0.180)
> 279.95 : final bay departure scheduled
[t: 274.677] pid 416/4 arrival event - queue len: 184
> 295.87 (0.080) : scheduled arrival for pid 418/4
* all bays busy; inserting pid 416/4 in queue
[t: 275.251] pid 414/3 arrival event - queue len: 185
> 280.16 (0.256) : scheduled arrival for pid 419/3
* all bays busy; inserting pid 414/3 in queue
[t: 276.137] pid 410/1 arrival event - queue len: 186
> 284.61 (0.351) : scheduled arrival for pid 420/1
* all bays busy; inserting pid 410/1 in queue
[t: 276.334] pid 408/7 arrival event - queue len: 187
> 279.84 (0.311) : scheduled arrival for pid 421/7
* all bays busy; inserting pid 408/7 in queue
[t: 276.532] pid 225/6 departure event (bay 4)
[t: 276.532] handling pid 231/6 in bay 4
+ 1.608/1.608 (0.083) : [1] inspect (failed; r=0.100; p=0.140)
+ 7.622/6.014 (0.944) : [1] repair
+ 8.443/0.820 (0.402) : [1] inspect (passed; r=0.249; p=0.070)
+10.129/1.686 (0.860) : [2] inspect (passed; r=0.455; p=0.140)
> 286.66 : final bay departure scheduled
[t: 276.591] pid 417/2 arrival event - queue len: 187
> 279.06 (0.427) : scheduled arrival for pid 422/2
* all bays busy; inserting pid 417/2 in queue
[t: 278.899] pid 229/3 departure event (bay 5)
[t: 278.899] handling pid 228/7 in bay 5
+ 1.397/1.397 (0.710) : [1] inspect (passed; r=0.497; p=0.180)
+ 2.456/1.058 (0.226) : [2] inspect (failed; r=0.049; p=0.180)
+ 4.752/2.297 (0.153) : [2] repair
+ 5.372/0.619 (0.483) : [2] inspect (passed; r=0.876; p=0.090)
> 284.27 : final bay departure scheduled
[t: 279.043] pid 226/6 departure event (bay 3)
[t: 279.043] handling pid 233/2 in bay 3
+ 1.624/1.624 (0.242) : [1] inspect (passed; r=0.662; p=0.140)
+ 3.312/1.688 (0.879) : [2] inspect (passed; r=0.363; p=0.140)
> 282.35 : final bay departure scheduled
[t: 279.061] pid 422/2 arrival event - queue len: 186
```

## APPENDIX C: The Application Trace File (ATF)

```
> 286.62 (0.074) : scheduled arrival for pid 423/2
* all bays busy; inserting pid 422/2 in queue
[t: 279.838] pid 421/7 arrival event - queue len: 187
> 280.39 (0.832) : scheduled arrival for pid 424/7
* all bays busy; inserting pid 421/7 in queue
[t: 279.950] pid 219/2 departure event (bay 6)
[t: 279.950] handling pid 235/7 in bay 6
+ 1.976/1.976 (0.412) : [1] inspect (passed; r=0.540; p=0.260)
+ 3.971/1.995 (0.351) : [2] inspect (passed; r=0.354; p=0.260)
+ 5.839/1.868 (0.775) : [3] inspect (passed; r=0.451; p=0.260)
> 285.79 : final bay departure scheduled
[t: 280.158] pid 419/3 arrival event - queue len: 187
> 286.69 (0.163) : scheduled arrival for pid 425/3
* all bays busy; inserting pid 419/3 in queue
[t: 280.390] pid 424/7 arrival event - queue len: 188
> 287.91 (0.082) : scheduled arrival for pid 426/7
* all bays busy; inserting pid 424/7 in queue
[t: 280.480] pid 217/4 departure event (bay 1)
[t: 280.480] handling pid 230/3 in bay 1
+ 2.275/2.275 (0.475) : [1] inspect (failed; r=0.099; p=0.120)
+ 2.962/0.687 (0.801) : [1] repair
+ 4.093/1.131 (0.462) : [1] inspect (passed; r=0.171; p=0.060)
+ 6.018/1.925 (0.125) : [2] inspect (passed; r=0.181; p=0.120)
+ 8.536/2.518 (0.718) : [3] inspect (passed; r=0.399; p=0.120)
+10.763/2.227 (0.427) : [4] inspect (passed; r=0.326; p=0.120)
> 291.24 : final bay departure scheduled
[t: 282.355] pid 233/2 departure event (bay 3)
[t: 282.355] handling pid 238/3 in bay 3
+ 2.098/2.098 (0.006) : [1] inspect (passed; r=0.674; p=0.260)
+ 4.099/2.001 (0.330) : [2] inspect (passed; r=0.428; p=0.260)
+ 5.920/1.821 (0.930) : [3] inspect (passed; r=0.791; p=0.260)
> 288.27 : final bay departure scheduled
[t: 282.716] pid 222/3 departure event (bay 2)
[t: 282.716] handling pid 239/3 in bay 2
+ 1.441/1.441 (0.772) : [1] inspect (passed; r=0.600; p=0.180)
+ 2.820/1.379 (0.684) : [2] inspect (passed; r=0.303; p=0.180)
> 285.54 : final bay departure scheduled
[t: 284.270] pid 228/7 departure event (bay 5)
[t: 284.270] handling pid 227/1 in bay 5
+ 1.939/1.939 (0.385) : [1] inspect (failed; r=0.164; p=0.210)
+ 4.806/2.868 (0.741) : [1] repair
+ 5.782/0.976 (0.517) : [1] inspect (passed; r=0.315; p=0.105)
+ 7.699/1.917 (0.170) : [2] inspect (passed; r=0.938; p=0.210)
+ 9.640/1.941 (0.405) : [3] inspect (passed; r=0.331; p=0.210)
> 293.91 : final bay departure scheduled
[t: 284.609] pid 420/1 arrival event - queue len: 185
> 297.69 (0.199) : scheduled arrival for pid 427/1
* all bays busy; inserting pid 420/1 in queue
[t: 285.535] pid 239/3 departure event (bay 2)
[t: 285.535] handling pid 232/4 in bay 2
+ 0.930/0.930 (0.043) : [1] inspect (passed; r=0.185; p=0.180)
+ 2.435/1.505 (0.864) : [2] inspect (passed; r=0.576; p=0.180)
> 287.97 : final bay departure scheduled
[t: 285.788] pid 235/7 departure event (bay 6)
[t: 285.788] handling pid 218/5 in bay 6
+ 1.964/1.964 (0.642) : [1] inspect (failed; r=0.163; p=0.210)
+ 5.002/3.038 (0.908) : [1] repair
+ 5.994/0.992 (0.836) : [1] inspect (failed; r=0.020; p=0.105)
+ 6.471/0.477 (0.725) : [1] repair
+ 7.470/0.999 (0.978) : [1] inspect (passed; r=0.736; p=0.105)
+ 9.441/1.970 (0.705) : [2] inspect (passed; r=0.346; p=0.210)
+11.402/1.961 (0.615) : [3] inspect (passed; r=0.733; p=0.210)
> 297.19 : final bay departure scheduled
[t: 286.616] pid 423/2 arrival event - queue len: 184
> 287.37 (0.770) : scheduled arrival for pid 428/2
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 423/2 in queue
[t: 286.661] pid 231/6 departure event (bay 4)
[t: 286.661] handling pid 242/4 in bay 4
+ 1.639/1.639 (0.390) : [1] inspect (passed; r=0.957; p=0.140)
+ 3.246/1.607 (0.068) : [2] inspect (passed; r=0.709; p=0.140)
> 289.91 : final bay departure scheduled
[t: 286.691] pid 425/3 arrival event - queue len: 184
> 290.71 (0.327) : scheduled arrival for pid 429/3
* all bays busy; inserting pid 425/3 in queue
[t: 287.373] pid 428/2 arrival event - queue len: 185
> 287.69 (0.895) : scheduled arrival for pid 430/2
* all bays busy; inserting pid 428/2 in queue
[t: 287.694] pid 430/2 arrival event - queue len: 186
> 289.61 (0.517) : scheduled arrival for pid 431/2
* all bays busy; inserting pid 430/2 in queue
[t: 287.906] pid 426/7 arrival event - queue len: 187
> 294.59 (0.108) : scheduled arrival for pid 432/7
* all bays busy; inserting pid 426/7 in queue
[t: 287.970] pid 232/4 departure event (bay 2)
[t: 287.970] handling pid 236/2 in bay 2
+ 2.449/2.449 (0.649) : [1] inspect (passed; r=0.601; p=0.120)
+ 4.865/2.416 (0.616) : [2] inspect (failed; r=0.056; p=0.120)
+ 6.305/1.440 (0.876) : [2] repair
+ 7.428/1.123 (0.445) : [2] inspect (passed; r=0.795; p=0.060)
+ 9.567/2.139 (0.339) : [3] inspect (passed; r=0.846; p=0.120)
+11.591/2.024 (0.224) : [4] inspect (passed; r=0.653; p=0.120)
> 299.56 : final bay departure scheduled
[t: 288.275] pid 238/3 departure event (bay 3)
[t: 288.275] handling pid 237/7 in bay 3
+ 1.265/1.265 (0.521) : [1] inspect (passed; r=0.741; p=0.180)
+ 2.579/1.314 (0.592) : [2] inspect (passed; r=0.220; p=0.180)
> 290.85 : final bay departure scheduled
[t: 289.606] pid 431/2 arrival event - queue len: 186
> 293.48 (0.263) : scheduled arrival for pid 433/2
* all bays busy; inserting pid 431/2 in queue
[t: 289.907] pid 242/4 departure event (bay 4)
[t: 289.907] handling pid 245/2 in bay 4
+ 2.617/2.617 (0.817) : [1] inspect (passed; r=0.121; p=0.120)
+ 4.532/1.914 (0.114) : [2] inspect (failed; r=0.102; p=0.120)
+ 6.196/1.664 (0.391) : [2] repair
+ 7.396/1.200 (0.600) : [2] inspect (passed; r=0.356; p=0.060)
+10.138/2.741 (0.941) : [3] inspect (passed; r=0.598; p=0.120)
+12.651/2.513 (0.713) : [4] inspect (failed; r=0.092; p=0.120)
+18.343/5.693 (0.053) : [4] repair
+19.397/1.053 (0.307) : [4] inspect (passed; r=0.797; p=0.060)
> 309.30 : final bay departure scheduled
[t: 290.710] pid 429/3 arrival event - queue len: 186
> 291.59 (0.783) : scheduled arrival for pid 434/3
* all bays busy; inserting pid 429/3 in queue
[t: 290.854] pid 237/7 departure event (bay 3)
[t: 290.854] handling pid 247/2 in bay 3
+ 1.936/1.936 (0.362) : [1] inspect (passed; r=0.576; p=0.210)
+ 3.901/1.964 (0.643) : [2] inspect (passed; r=0.822; p=0.210)
+ 5.850/1.950 (0.495) : [3] inspect (passed; r=0.453; p=0.210)
> 296.70 : final bay departure scheduled
[t: 291.243] pid 230/3 departure event (bay 1)
[t: 291.243] handling pid 246/7 in bay 1
+ 1.589/1.589 (0.984) : [1] inspect (passed; r=0.306; p=0.180)
+ 2.792/1.203 (0.433) : [2] inspect (passed; r=0.452; p=0.180)
> 294.04 : final bay departure scheduled
[t: 291.591] pid 434/3 arrival event - queue len: 185
> 293.61 (0.571) : scheduled arrival for pid 435/3
* all bays busy; inserting pid 434/3 in queue
[t: 292.189] pid 415/6 arrival event - queue len: 186
> 294.39 (0.720) : scheduled arrival for pid 436/6
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 415/6 in queue
[t: 293.478] pid 433/2 arrival event - queue len: 187
> 299.99 (0.106) : scheduled arrival for pid 437/2
* all bays busy; inserting pid 433/2 in queue
[t: 293.608] pid 435/3 arrival event - queue len: 188
> 294.56 (0.767) : scheduled arrival for pid 438/3
* all bays busy; inserting pid 435/3 in queue
[t: 293.910] pid 227/1 departure event (bay 5)
[t: 293.910] handling pid 248/2 in bay 5
+ 1.668/1.668 (0.691) : [1] inspect (failed; r=0.010; p=0.300)
+ 5.100/3.433 (0.431) : [1] repair
+ 5.773/0.673 (0.462) : [1] inspect (passed; r=0.686; p=0.150)
+ 7.002/1.229 (0.378) : [2] inspect (passed; r=0.604; p=0.300)
+ 8.357/1.355 (0.468) : [3] inspect (passed; r=0.875; p=0.300)
+10.227/1.870 (0.836) : [4] inspect (failed; r=0.242; p=0.300)
+15.798/5.570 (0.059) : [4] repair
+16.560/0.763 (0.589) : [4] inspect (passed; r=0.767; p=0.150)
> 310.47 : final bay departure scheduled
[t: 294.035] pid 246/7 departure event (bay 1)
[t: 294.035] handling pid 250/2 in bay 1
+ 1.942/1.942 (0.418) : [1] inspect (passed; r=0.561; p=0.210)
+ 3.942/2.000 (0.999) : [2] inspect (passed; r=0.999; p=0.210)
+ 5.849/1.907 (0.071) : [3] inspect (passed; r=0.994; p=0.210)
> 299.88 : final bay departure scheduled
[t: 294.386] pid 436/6 arrival event - queue len: 187
> 304.44 (0.223) : scheduled arrival for pid 439/6
* all bays busy; inserting pid 436/6 in queue
[t: 294.565] pid 438/3 arrival event - queue len: 188
> 298.44 (0.341) : scheduled arrival for pid 440/3
* all bays busy; inserting pid 438/3 in queue
[t: 294.588] pid 432/7 arrival event - queue len: 189
> 296.88 (0.466) : scheduled arrival for pid 441/7
* all bays busy; inserting pid 432/7 in queue
[t: 295.874] pid 418/4 arrival event - queue len: 190
> 309.34 (0.201) : scheduled arrival for pid 442/4
* all bays busy; inserting pid 418/4 in queue
[t: 296.704] pid 247/2 departure event (bay 3)
[t: 296.704] handling pid 240/3 in bay 3
+ 2.021/2.021 (0.263) : [1] inspect (passed; r=0.907; p=0.260)
+ 4.076/2.055 (0.149) : [2] inspect (passed; r=0.287; p=0.260)
+ 6.011/1.934 (0.552) : [3] inspect (passed; r=0.852; p=0.260)
> 302.71 : final bay departure scheduled
[t: 296.877] pid 441/7 arrival event - queue len: 190
> 297.10 (0.928) : scheduled arrival for pid 443/7
* all bays busy; inserting pid 441/7 in queue
[t: 297.102] pid 443/7 arrival event - queue len: 191
> 297.38 (0.911) : scheduled arrival for pid 444/7
* all bays busy; inserting pid 443/7 in queue
[t: 297.191] pid 218/5 departure event (bay 6)
[t: 297.191] handling pid 252/3 in bay 6
+ 0.857/0.857 (0.041) : [1] inspect (passed; r=0.528; p=0.360)
+ 2.534/1.677 (0.627) : [2] inspect (passed; r=0.376; p=0.360)
+ 3.656/1.121 (0.230) : [3] inspect (passed; r=0.856; p=0.360)
+ 4.730/1.074 (0.196) : [4] inspect (passed; r=0.667; p=0.360)
> 301.92 : final bay departure scheduled
[t: 297.382] pid 444/7 arrival event - queue len: 191
> 301.15 (0.285) : scheduled arrival for pid 445/7
* all bays busy; inserting pid 444/7 in queue
[t: 297.688] pid 427/1 arrival event - queue len: 192
> 303.39 (0.494) : scheduled arrival for pid 446/1
* all bays busy; inserting pid 427/1 in queue
[t: 298.438] pid 440/3 arrival event - queue len: 193
> 302.86 (0.293) : scheduled arrival for pid 447/3
* all bays busy; inserting pid 440/3 in queue
[t: 299.562] pid 236/2 departure event (bay 2)
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 299.562] handling pid 249/7 in bay 2
+ 2.038/2.038 (0.207) : [1] inspect (failed; r=0.210; p=0.260)
+ 3.057/1.019 (0.341) : [1] repair
+ 3.969/0.911 (0.924) : [1] inspect (passed; r=0.576; p=0.130)
+ 5.942/1.973 (0.423) : [2] inspect (passed; r=0.747; p=0.260)
+ 7.794/1.853 (0.825) : [3] inspect (failed; r=0.205; p=0.260)
+ 9.492/1.698 (0.438) : [3] repair
+10.500/1.008 (0.280) : [3] inspect (passed; r=0.872; p=0.130)
> 310.06 : final bay departure scheduled
[t: 299.884] pid 250/2 departure event (bay 1)
[t: 299.884] handling pid 251/2 in bay 1
+ 1.867/1.867 (0.777) : [1] inspect (failed; r=0.230; p=0.260)
+ 3.488/1.621 (0.529) : [1] repair
+ 4.527/1.039 (0.071) : [1] inspect (passed; r=0.464; p=0.130)
+ 6.442/1.915 (0.617) : [2] inspect (passed; r=0.480; p=0.260)
+ 8.281/1.839 (0.869) : [3] inspect (failed; r=0.091; p=0.260)
+12.881/4.599 (0.028) : [3] repair
+13.878/0.997 (0.353) : [3] inspect (passed; r=0.581; p=0.130)
> 313.76 : final bay departure scheduled
[t: 299.987] pid 437/2 arrival event - queue len: 192
> 300.73 (0.773) : scheduled arrival for pid 448/2
* all bays busy; inserting pid 437/2 in queue
[t: 300.732] pid 448/2 arrival event - queue len: 193
> 302.04 (0.636) : scheduled arrival for pid 449/2
* all bays busy; inserting pid 448/2 in queue
[t: 301.149] pid 445/7 arrival event - queue len: 194
> 305.25 (0.255) : scheduled arrival for pid 450/7
* all bays busy; inserting pid 445/7 in queue
[t: 301.920] pid 252/3 departure event (bay 6)
[t: 301.920] handling pid 234/6 in bay 6
+ 1.463/1.463 (0.804) : [1] inspect (passed; r=0.220; p=0.180)
+ 2.660/1.197 (0.424) : [2] inspect (passed; r=0.410; p=0.180)
> 304.58 : final bay departure scheduled
[t: 302.044] pid 449/2 arrival event - queue len: 194
> 305.87 (0.268) : scheduled arrival for pid 451/2
* all bays busy; inserting pid 449/2 in queue
[t: 302.715] pid 240/3 departure event (bay 3)
[t: 302.715] handling pid 253/3 in bay 3
+ 1.136/1.136 (0.337) : [1] inspect (passed; r=0.938; p=0.180)
+ 2.512/1.377 (0.681) : [2] inspect (failed; r=0.092; p=0.180)
+ 2.869/0.357 (0.750) : [2] repair
+ 3.580/0.711 (0.746) : [2] inspect (passed; r=0.361; p=0.090)
> 306.30 : final bay departure scheduled
[t: 302.858] pid 447/3 arrival event - queue len: 194
> 307.72 (0.259) : scheduled arrival for pid 452/3
* all bays busy; inserting pid 447/3 in queue
[t: 303.392] pid 446/1 arrival event - queue len: 195
> 310.38 (0.422) : scheduled arrival for pid 453/1
* all bays busy; inserting pid 446/1 in queue
[t: 304.440] pid 439/6 arrival event - queue len: 196
> 310.47 (0.407) : scheduled arrival for pid 454/6
* all bays busy; inserting pid 439/6 in queue
[t: 304.581] pid 234/6 departure event (bay 6)
[t: 304.581] handling pid 255/2 in bay 6
+ 1.614/1.614 (0.139) : [1] inspect (passed; r=0.811; p=0.140)
+ 3.278/1.664 (0.636) : [2] inspect (passed; r=0.340; p=0.140)
> 307.86 : final bay departure scheduled
[t: 305.253] pid 450/7 arrival event - queue len: 196
> 305.89 (0.809) : scheduled arrival for pid 455/7
* all bays busy; inserting pid 450/7 in queue
[t: 305.867] pid 451/2 arrival event - queue len: 197
> 305.95 (0.973) : scheduled arrival for pid 456/2
* all bays busy; inserting pid 451/2 in queue
[t: 305.887] pid 455/7 arrival event - queue len: 198
> 306.92 (0.710) : scheduled arrival for pid 457/7
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 455/7 in queue
[t: 305.946] pid 456/2 arrival event - queue len: 199
> 308.98 (0.351) : scheduled arrival for pid 458/2
* all bays busy; inserting pid 456/2 in queue
[t: 306.295] pid 253/3 departure event (bay 3)
[t: 306.295] handling pid 254/7 in bay 3
+ 1.073/1.073 (0.247) : [1] inspect (passed; r=0.871; p=0.180)
+ 2.375/1.302 (0.574) : [2] inspect (passed; r=0.530; p=0.180)
> 308.67 : final bay departure scheduled
[t: 306.915] pid 457/7 arrival event - queue len: 199
> 309.61 (0.407) : scheduled arrival for pid 459/7
* all bays busy; inserting pid 457/7 in queue
[t: 307.723] pid 452/3 arrival event - queue len: 200
> 311.85 (0.317) : scheduled arrival for pid 460/3
* all bays busy; inserting pid 452/3 in queue
[t: 307.858] pid 255/2 departure event (bay 6)
[t: 307.858] handling pid 259/7 in bay 6
+ 1.875/1.875 (0.749) : [1] inspect (passed; r=0.655; p=0.260)
+ 3.848/1.973 (0.424) : [2] inspect (passed; r=0.758; p=0.260)
+ 5.658/1.810 (0.966) : [3] inspect (passed; r=0.850; p=0.260)
> 313.52 : final bay departure scheduled
[t: 308.670] pid 254/7 departure event (bay 3)
[t: 308.670] handling pid 258/2 in bay 3
+ 1.927/1.927 (0.268) : [1] inspect (passed; r=0.773; p=0.210)
+ 3.878/1.951 (0.509) : [2] inspect (failed; r=0.020; p=0.210)
+ 7.140/3.262 (0.201) : [2] repair
+ 8.130/0.990 (0.802) : [2] inspect (failed; r=0.067; p=0.105)
+ 8.738/0.608 (0.541) : [2] repair
+ 9.712/0.974 (0.485) : [2] inspect (passed; r=0.647; p=0.105)
+11.705/1.993 (0.929) : [3] inspect (passed; r=0.887; p=0.210)
> 320.37 : final bay departure scheduled
[t: 308.985] pid 458/2 arrival event - queue len: 199
> 309.50 (0.838) : scheduled arrival for pid 461/2
* all bays busy; inserting pid 458/2 in queue
[t: 309.303] pid 245/2 departure event (bay 4)
[t: 309.303] handling pid 244/4 in bay 4
+ 2.056/2.056 (0.146) : [1] inspect (passed; r=0.635; p=0.260)
+ 3.947/1.891 (0.698) : [2] inspect (failed; r=0.005; p=0.260)
+ 7.479/3.532 (0.857) : [2] repair
+ 8.486/1.008 (0.282) : [2] inspect (passed; r=0.573; p=0.130)
+10.506/2.019 (0.269) : [3] inspect (failed; r=0.089; p=0.260)
+11.523/1.017 (0.515) : [3] repair
+12.440/0.918 (0.883) : [3] inspect (failed; r=0.045; p=0.130)
+14.148/1.707 (0.169) : [3] repair
+15.048/0.900 (0.999) : [3] inspect (passed; r=0.178; p=0.130)
> 324.35 : final bay departure scheduled
[t: 309.342] pid 442/4 arrival event - queue len: 199
> 310.34 (0.888) : scheduled arrival for pid 462/4
* all bays busy; inserting pid 442/4 in queue
[t: 309.497] pid 461/2 arrival event - queue len: 200
> 309.95 (0.857) : scheduled arrival for pid 463/2
* all bays busy; inserting pid 461/2 in queue
[t: 309.609] pid 459/7 arrival event - queue len: 201
> 318.31 (0.055) : scheduled arrival for pid 464/7
* all bays busy; inserting pid 459/7 in queue
[t: 309.945] pid 463/2 arrival event - queue len: 202
> 313.55 (0.288) : scheduled arrival for pid 465/2
* all bays busy; inserting pid 463/2 in queue
[t: 310.062] pid 249/7 departure event (bay 2)
[t: 310.062] handling pid 261/2 in bay 2
+ 1.909/1.909 (0.094) : [1] inspect (failed; r=0.186; p=0.210)
+ 3.096/1.186 (0.809) : [1] repair
+ 4.070/0.974 (0.488) : [1] inspect (passed; r=0.909; p=0.105)
+ 5.979/1.909 (0.086) : [2] inspect (passed; r=0.600; p=0.210)
+ 7.941/1.963 (0.628) : [3] inspect (passed; r=0.513; p=0.210)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 318.00 : final bay departure scheduled
[t: 310.343] pid 462/4 arrival event - queue len: 202
> 324.18 (0.193) : scheduled arrival for pid 466/4
* all bays busy; inserting pid 462/4 in queue
[t: 310.381] pid 453/1 arrival event - queue len: 203
> 317.51 (0.415) : scheduled arrival for pid 467/1
* all bays busy; inserting pid 453/1 in queue
[t: 310.466] pid 454/6 arrival event - queue len: 204
> 311.00 (0.923) : scheduled arrival for pid 468/6
* all bays busy; inserting pid 454/6 in queue
[t: 310.470] pid 248/2 departure event (bay 5)
[t: 310.470] handling pid 257/3 in bay 5
+ 1.895/1.895 (0.684) : [1] inspect (passed; r=0.640; p=0.260)
+ 3.865/1.970 (0.434) : [2] inspect (failed; r=0.143; p=0.260)
+ 4.441/0.576 (0.587) : [2] repair
+ 5.481/1.040 (0.067) : [2] inspect (passed; r=0.881; p=0.130)
+ 7.331/1.850 (0.833) : [3] inspect (failed; r=0.142; p=0.260)
+ 9.755/2.424 (0.215) : [3] repair
+10.700/0.945 (0.701) : [3] inspect (failed; r=0.056; p=0.130)
+11.613/0.913 (0.246) : [3] repair
+12.643/1.030 (0.135) : [3] inspect (passed; r=0.160; p=0.130)
> 323.11 : final bay departure scheduled
[t: 311.004] pid 468/6 arrival event - queue len: 204
> 311.76 (0.893) : scheduled arrival for pid 469/6
* all bays busy; inserting pid 468/6 in queue
[t: 311.764] pid 469/6 arrival event - queue len: 205
> 322.15 (0.212) : scheduled arrival for pid 470/6
* all bays busy; inserting pid 469/6 in queue
[t: 311.854] pid 460/3 arrival event - queue len: 206
> 317.45 (0.211) : scheduled arrival for pid 471/3
* all bays busy; inserting pid 460/3 in queue
[t: 313.516] pid 259/7 departure event (bay 6)
[t: 313.516] handling pid 263/2 in bay 6
+ 1.942/1.942 (0.421) : [1] inspect (failed; r=0.083; p=0.210)
+ 3.945/2.003 (0.989) : [1] repair
+ 4.924/0.979 (0.582) : [1] inspect (passed; r=0.867; p=0.105)
+ 6.844/1.920 (0.195) : [2] inspect (failed; r=0.021; p=0.210)
+ 9.329/2.485 (0.352) : [2] repair
+10.303/0.975 (0.496) : [2] inspect (passed; r=0.566; p=0.105)
+12.266/1.963 (0.629) : [3] inspect (passed; r=0.345; p=0.210)
> 325.78 : final bay departure scheduled
[t: 313.555] pid 465/2 arrival event - queue len: 206
> 314.79 (0.653) : scheduled arrival for pid 472/2
* all bays busy; inserting pid 465/2 in queue
[t: 313.762] pid 251/2 departure event (bay 1)
[t: 313.762] handling pid 243/5 in bay 1
+ 1.952/1.952 (0.493) : [1] inspect (failed; r=0.097; p=0.260)
+ 3.315/1.363 (0.448) : [1] repair
+ 4.285/0.969 (0.538) : [1] inspect (passed; r=0.543; p=0.130)
+ 6.300/2.016 (0.281) : [2] inspect (failed; r=0.122; p=0.260)
+ 8.044/1.744 (0.209) : [2] repair
+ 9.018/0.974 (0.507) : [2] inspect (passed; r=0.372; p=0.130)
+10.868/1.850 (0.834) : [3] inspect (passed; r=0.394; p=0.260)
> 324.63 : final bay departure scheduled
[t: 314.792] pid 472/2 arrival event - queue len: 206
> 316.10 (0.636) : scheduled arrival for pid 473/2
* all bays busy; inserting pid 472/2 in queue
[t: 316.103] pid 473/2 arrival event - queue len: 207
> 330.71 (0.006) : scheduled arrival for pid 474/2
* all bays busy; inserting pid 473/2 in queue
[t: 317.450] pid 471/3 arrival event - queue len: 208
> 319.41 (0.579) : scheduled arrival for pid 475/3
* all bays busy; inserting pid 471/3 in queue
[t: 317.511] pid 467/1 arrival event - queue len: 209
> 349.42 (0.019) : scheduled arrival for pid 476/1
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 467/1 in queue
[t: 318.003] pid 261/2 departure event (bay 2)
[t: 318.003] handling pid 265/2 in bay 2
+ 1.953/1.953 (0.491) : [1] inspect (passed; r=0.533; p=0.260)
+ 4.000/2.048 (0.175) : [2] inspect (passed; r=0.560; p=0.260)
+ 6.086/2.086 (0.047) : [3] inspect (failed; r=0.192; p=0.260)
+ 7.313/1.227 (0.287) : [3] repair
+ 8.235/0.922 (0.852) : [3] inspect (passed; r=0.659; p=0.130)
> 326.24 : final bay departure scheduled
[t: 318.306] pid 464/7 arrival event - queue len: 209
> 325.07 (0.105) : scheduled arrival for pid 477/7
* all bays busy; inserting pid 464/7 in queue
[t: 319.415] pid 475/3 arrival event - queue len: 210
> 325.67 (0.176) : scheduled arrival for pid 478/3
* all bays busy; inserting pid 475/3 in queue
[t: 320.375] pid 258/2 departure event (bay 3)
[t: 320.375] handling pid 267/2 in bay 3
+ 1.871/1.871 (0.763) : [1] inspect (passed; r=0.703; p=0.260)
+ 3.814/1.943 (0.525) : [2] inspect (failed; r=0.203; p=0.260)
+ 4.657/0.843 (0.431) : [2] repair
+ 5.590/0.933 (0.780) : [2] inspect (passed; r=0.724; p=0.130)
+ 7.683/2.094 (0.021) : [3] inspect (passed; r=0.829; p=0.260)
> 328.06 : final bay departure scheduled
[t: 322.147] pid 470/6 arrival event - queue len: 210
> 325.56 (0.601) : scheduled arrival for pid 479/6
* all bays busy; inserting pid 470/6 in queue
[t: 323.108] pid 371/5 arrival event - queue len: 211
> 342.91 (0.163) : scheduled arrival for pid 480/5
* all bays busy; inserting pid 371/5 in queue
[t: 323.113] pid 257/3 departure event (bay 5)
[t: 323.113] handling pid 264/3 in bay 5
+ 1.489/1.489 (0.842) : [1] inspect (passed; r=0.651; p=0.180)
+ 2.522/1.033 (0.190) : [2] inspect (passed; r=0.549; p=0.180)
> 325.63 : final bay departure scheduled
[t: 324.181] pid 466/4 arrival event - queue len: 211
> 326.51 (0.758) : scheduled arrival for pid 481/4
* all bays busy; inserting pid 466/4 in queue
[t: 324.351] pid 244/4 departure event (bay 4)
[t: 324.351] handling pid 268/2 in bay 4
+ 2.439/2.439 (0.639) : [1] inspect (passed; r=0.434; p=0.120)
+ 4.791/2.352 (0.552) : [2] inspect (passed; r=0.357; p=0.120)
+ 7.353/2.561 (0.761) : [3] inspect (passed; r=0.628; p=0.120)
+ 9.282/1.929 (0.129) : [4] inspect (passed; r=0.426; p=0.120)
> 333.63 : final bay departure scheduled
[t: 324.630] pid 243/5 departure event (bay 1)
[t: 324.630] handling pid 256/6 in bay 1
+ 1.312/1.312 (0.366) : [1] inspect (passed; r=0.742; p=0.360)
+ 2.278/0.965 (0.118) : [2] inspect (failed; r=0.009; p=0.360)
+ 2.937/0.659 (0.624) : [2] repair
+ 3.902/0.964 (0.806) : [2] inspect (passed; r=0.609; p=0.180)
+ 6.029/2.128 (0.948) : [3] inspect (passed; r=0.590; p=0.360)
+ 8.065/2.036 (0.883) : [4] inspect (failed; r=0.261; p=0.360)
+ 8.763/0.698 (0.988) : [4] repair
+ 9.357/0.594 (0.277) : [4] inspect (passed; r=0.291; p=0.180)
> 333.99 : final bay departure scheduled
[t: 325.071] pid 477/7 arrival event - queue len: 210
> 329.63 (0.219) : scheduled arrival for pid 482/7
* all bays busy; inserting pid 477/7 in queue
[t: 325.559] pid 479/6 arrival event - queue len: 211
> 328.38 (0.656) : scheduled arrival for pid 483/6
* all bays busy; inserting pid 479/6 in queue
[t: 325.635] pid 264/3 departure event (bay 5)
[t: 325.635] handling pid 270/2 in bay 5
+ 1.479/1.479 (0.827) : [1] inspect (passed; r=0.718; p=0.180)
+ 2.418/0.939 (0.056) : [2] inspect (passed; r=0.355; p=0.180)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 328.05 : final bay departure scheduled
[t: 325.671] pid 478/3 arrival event - queue len: 211
> 339.50 (0.021) : scheduled arrival for pid 484/3
* all bays busy; inserting pid 478/3 in queue
[t: 325.782] pid 263/2 departure event (bay 6)
[t: 325.782] handling pid 272/2 in bay 6
+ 2.056/2.056 (0.145) : [1] inspect (passed; r=0.887; p=0.260)
+ 4.056/1.999 (0.336) : [2] inspect (passed; r=0.846; p=0.260)
+ 6.117/2.061 (0.129) : [3] inspect (passed; r=0.565; p=0.260)
> 331.90 : final bay departure scheduled
[t: 326.238] pid 265/2 departure event (bay 2)
[t: 326.238] handling pid 273/2 in bay 2
+ 2.093/2.093 (0.022) : [1] inspect (passed; r=0.339; p=0.260)
+ 4.058/1.965 (0.450) : [2] inspect (passed; r=0.428; p=0.260)
+ 5.947/1.888 (0.706) : [3] inspect (passed; r=0.401; p=0.260)
> 332.18 : final bay departure scheduled
[t: 326.513] pid 481/4 arrival event - queue len: 210
> 328.05 (0.833) : scheduled arrival for pid 485/4
* all bays busy; inserting pid 481/4 in queue
[t: 328.048] pid 485/4 arrival event - queue len: 211
> 330.18 (0.776) : scheduled arrival for pid 486/4
* all bays busy; inserting pid 485/4 in queue
[t: 328.053] pid 270/2 departure event (bay 5)
[t: 328.053] handling pid 274/2 in bay 5
+ 1.835/1.835 (0.884) : [1] inspect (failed; r=0.057; p=0.260)
+ 2.946/1.111 (0.542) : [1] repair
+ 3.928/0.982 (0.452) : [1] inspect (passed; r=0.939; p=0.130)
+ 6.018/2.090 (0.034) : [2] inspect (passed; r=0.605; p=0.260)
+ 7.886/1.868 (0.773) : [3] inspect (failed; r=0.122; p=0.260)
+ 9.434/1.548 (0.463) : [3] repair
+10.440/1.006 (0.296) : [3] inspect (passed; r=0.500; p=0.130)
> 338.49 : final bay departure scheduled
[t: 328.058] pid 267/2 departure event (bay 3)
[t: 328.058] handling pid 275/2 in bay 3
+ 1.881/1.881 (0.731) : [1] inspect (passed; r=0.674; p=0.260)
+ 3.970/2.089 (0.035) : [2] inspect (passed; r=0.530; p=0.260)
+ 6.013/2.043 (0.191) : [3] inspect (failed; r=0.194; p=0.260)
+ 7.872/1.859 (0.167) : [3] repair
+ 8.899/1.026 (0.159) : [3] inspect (passed; r=0.436; p=0.130)
> 336.96 : final bay departure scheduled
[t: 328.382] pid 483/6 arrival event - queue len: 210
> 331.49 (0.629) : scheduled arrival for pid 487/6
* all bays busy; inserting pid 483/6 in queue
[t: 329.631] pid 482/7 arrival event - queue len: 211
> 332.03 (0.450) : scheduled arrival for pid 488/7
* all bays busy; inserting pid 482/7 in queue
[t: 330.183] pid 486/4 arrival event - queue len: 212
> 341.41 (0.263) : scheduled arrival for pid 489/4
* all bays busy; inserting pid 486/4 in queue
[t: 330.713] pid 474/2 arrival event - queue len: 213
> 334.96 (0.231) : scheduled arrival for pid 490/2
* all bays busy; inserting pid 474/2 in queue
[t: 331.490] pid 487/6 arrival event - queue len: 214
> 335.57 (0.544) : scheduled arrival for pid 491/6
* all bays busy; inserting pid 487/6 in queue
[t: 331.899] pid 272/2 departure event (bay 6)
[t: 331.899] handling pid 276/2 in bay 6
+ 1.812/1.812 (0.958) : [1] inspect (passed; r=0.374; p=0.260)
+ 3.838/2.026 (0.248) : [2] inspect (passed; r=0.985; p=0.260)
+ 5.739/1.901 (0.663) : [3] inspect (failed; r=0.050; p=0.260)
+ 7.963/2.224 (0.863) : [3] repair
+ 8.972/1.009 (0.270) : [3] inspect (passed; r=0.349; p=0.130)
> 340.87 : final bay departure scheduled
[t: 332.025] pid 488/7 arrival event - queue len: 214
> 335.79 (0.286) : scheduled arrival for pid 492/7
```

## APPENDIX C: The Application Trace File (ATF)

```
* all bays busy; inserting pid 488/7 in queue
[t: 332.185] pid 273/2 departure event (bay 2)
[t: 332.185] handling pid 260/7 in bay 2
+ 2.074/2.074 (0.085) : [1] inspect (passed; r=0.339; p=0.260)
+ 3.932/1.858 (0.808) : [2] inspect (passed; r=0.488; p=0.260)
+ 5.996/2.063 (0.122) : [3] inspect (passed; r=0.700; p=0.260)
> 338.18 : final bay departure scheduled
[t: 333.633] pid 268/2 departure event (bay 4)
[t: 333.633] handling pid 269/3 in bay 4
+ 2.085/2.085 (0.050) : [1] inspect (passed; r=0.821; p=0.260)
+ 4.125/2.040 (0.200) : [2] inspect (passed; r=0.384; p=0.260)
+ 5.968/1.843 (0.856) : [3] inspect (passed; r=0.471; p=0.260)
> 339.60 : final bay departure scheduled
[t: 333.987] pid 256/6 departure event (bay 1)
[t: 333.987] handling pid 278/7 in bay 1
+ 1.697/1.697 (0.967) : [1] inspect (passed; r=0.349; p=0.140)
+ 3.307/1.610 (0.101) : [2] inspect (passed; r=0.866; p=0.140)
> 337.29 : final bay departure scheduled
[t: 334.959] pid 490/2 arrival event - queue len: 212
> 335.85 (0.735) : scheduled arrival for pid 493/2
* all bays busy; inserting pid 490/2 in queue
[t: 335.571] pid 491/6 arrival event - queue len: 213
> 336.32 (0.894) : scheduled arrival for pid 494/6
* all bays busy; inserting pid 491/6 in queue
[t: 335.785] pid 492/7 arrival event - queue len: 214
> 340.78 (0.189) : scheduled arrival for pid 495/7
* all bays busy; inserting pid 492/7 in queue
[t: 335.850] pid 493/2 arrival event - queue len: 215
> 341.27 (0.154) : scheduled arrival for pid 496/2
* all bays busy; inserting pid 493/2 in queue
[t: 336.324] pid 494/6 arrival event - queue len: 216
> 346.39 (0.223) : scheduled arrival for pid 497/6
* all bays busy; inserting pid 494/6 in queue
[t: 336.957] pid 275/2 departure event (bay 3)
[t: 336.957] handling pid 279/3 in bay 3
+ 1.914/1.914 (0.620) : [1] inspect (passed; r=0.487; p=0.260)
+ 3.717/1.802 (0.992) : [2] inspect (passed; r=0.732; p=0.260)
+ 5.707/1.991 (0.364) : [3] inspect (passed; r=0.569; p=0.260)
> 342.66 : final bay departure scheduled
[t: 337.294] pid 278/7 departure event (bay 1)
[t: 337.294] handling pid 281/3 in bay 1
+ 1.956/1.956 (0.556) : [1] inspect (passed; r=0.759; p=0.210)
+ 3.906/1.951 (0.506) : [2] inspect (failed; r=0.121; p=0.210)
+ 5.704/1.797 (0.432) : [2] repair
+ 6.689/0.986 (0.718) : [2] inspect (passed; r=0.850; p=0.105)
+ 8.608/1.919 (0.190) : [3] inspect (failed; r=0.004; p=0.210)
+ 8.887/0.278 (0.899) : [3] repair
+ 9.880/0.993 (0.865) : [3] inspect (passed; r=0.236; p=0.105)
> 347.17 : final bay departure scheduled
[t: 338.181] pid 260/7 departure event (bay 2)
[t: 338.181] handling pid 280/7 in bay 2
+ 1.927/1.927 (0.272) : [1] inspect (failed; r=0.141; p=0.210)
+ 3.809/1.882 (0.389) : [1] repair
+ 4.778/0.969 (0.376) : [1] inspect (passed; r=0.118; p=0.105)
+ 6.768/1.990 (0.903) : [2] inspect (failed; r=0.110; p=0.210)
+ 8.972/2.203 (0.552) : [2] repair
+ 9.958/0.987 (0.736) : [2] inspect (passed; r=0.272; p=0.105)
+11.891/1.932 (0.321) : [3] inspect (passed; r=0.392; p=0.210)
> 350.07 : final bay departure scheduled
[t: 338.493] pid 274/2 departure event (bay 5)
[t: 338.493] handling pid 262/4 in bay 5
+ 1.942/1.942 (0.526) : [1] inspect (passed; r=0.326; p=0.260)
+ 3.843/1.901 (0.663) : [2] inspect (passed; r=0.998; p=0.260)
+ 5.739/1.895 (0.682) : [3] inspect (passed; r=0.272; p=0.260)
> 344.23 : final bay departure scheduled
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 339.498] pid 484/3 arrival event - queue len: 213
> 340.88 (0.681) : scheduled arrival for pid 498/3
* all bays busy; inserting pid 484/3 in queue
[t: 339.601] pid 269/3 departure event (bay 4)
[t: 339.601] handling pid 277/2 in bay 4
+ 1.452/1.452 (0.788) : [1] inspect (passed; r=0.709; p=0.180)
+ 2.788/1.337 (0.624) : [2] inspect (passed; r=0.320; p=0.180)
> 342.39 : final bay departure scheduled
[t: 340.776] pid 495/7 arrival event - queue len: 213
> 343.72 (0.375) : scheduled arrival for pid 499/7
* all bays busy; inserting pid 495/7 in queue
[t: 340.872] pid 276/2 departure event (bay 6)
[t: 340.872] handling pid 282/3 in bay 6
+ 2.026/2.026 (0.246) : [1] inspect (passed; r=0.296; p=0.260)
+ 4.074/2.048 (0.172) : [2] inspect (passed; r=0.382; p=0.260)
+ 6.033/1.958 (0.472) : [3] inspect (failed; r=0.242; p=0.260)
+ 7.067/1.034 (0.675) : [3] repair
+ 8.027/0.960 (0.600) : [3] inspect (passed; r=0.525; p=0.130)
> 348.90 : final bay departure scheduled
[t: 340.879] pid 498/3 arrival event - queue len: 213
> 355.65 (0.017) : scheduled arrival for pid 500/3
* all bays busy; inserting pid 498/3 in queue
[t: 341.269] pid 496/2 arrival event - queue len: 214
> 342.05 (0.764) : scheduled arrival for pid 501/2
* all bays busy; inserting pid 496/2 in queue
[t: 341.414] pid 489/4 arrival event - queue len: 215
> 349.78 (0.370) : scheduled arrival for pid 502/4
* all bays busy; inserting pid 489/4 in queue
[t: 342.050] pid 501/2 arrival event - queue len: 216
> 342.13 (0.972) : scheduled arrival for pid 503/2
* all bays busy; inserting pid 501/2 in queue
[t: 342.133] pid 503/2 arrival event - queue len: 217
> 343.43 (0.640) : scheduled arrival for pid 504/2
* all bays busy; inserting pid 503/2 in queue
[t: 342.390] pid 277/2 departure event (bay 4)
[t: 342.390] handling pid 271/6 in bay 4
+ 2.094/2.094 (0.022) : [1] inspect (passed; r=0.427; p=0.260)
+ 4.040/1.946 (0.513) : [2] inspect (passed; r=0.519; p=0.260)
+ 6.129/2.089 (0.036) : [3] inspect (failed; r=0.250; p=0.260)
+ 6.358/0.229 (0.973) : [3] repair
+ 7.365/1.007 (0.284) : [3] inspect (passed; r=0.816; p=0.130)
> 349.75 : final bay departure scheduled
[t: 342.664] pid 279/3 departure event (bay 3)
[t: 342.664] handling pid 241/1 in bay 3
+ 0.919/0.919 (0.027) : [1] inspect (passed; r=0.599; p=0.180)
+ 2.178/1.259 (0.513) : [2] inspect (passed; r=0.338; p=0.180)
> 344.84 : final bay departure scheduled
[t: 342.908] pid 480/5 arrival event - queue len: 216
> 358.30 (0.244) : scheduled arrival for pid 505/5
* all bays busy; inserting pid 480/5 in queue
[t: 343.426] pid 504/2 arrival event - queue len: 217
> 349.96 (0.105) : scheduled arrival for pid 506/2
* all bays busy; inserting pid 504/2 in queue
[t: 343.715] pid 499/7 arrival event - queue len: 218
> 345.58 (0.538) : scheduled arrival for pid 507/7
* all bays busy; inserting pid 499/7 in queue
[t: 344.231] pid 262/4 departure event (bay 5)
[t: 344.231] handling pid 283/7 in bay 5
+ 2.065/2.065 (0.265) : [1] inspect (passed; r=0.526; p=0.120)
+ 4.749/2.685 (0.885) : [2] inspect (passed; r=0.310; p=0.120)
+ 7.433/2.684 (0.884) : [3] inspect (passed; r=0.863; p=0.120)
+10.153/2.720 (0.920) : [4] inspect (passed; r=0.912; p=0.120)
> 354.38 : final bay departure scheduled
[t: 344.842] pid 241/1 departure event (bay 3)
[t: 344.842] handling pid 285/2 in bay 3
```

## APPENDIX C: The Application Trace File (ATF)

```
+ 2.065/2.065 (0.975) : [1] inspect (passed; r=0.352; p=0.300)
+ 2.976/0.910 (0.150) : [2] inspect (failed; r=0.215; p=0.300)
+ 4.438/1.463 (0.273) : [2] repair
+ 5.069/0.631 (0.401) : [2] inspect (passed; r=0.332; p=0.150)
+ 5.807/0.738 (0.027) : [3] inspect (passed; r=0.983; p=0.300)
+ 7.688/1.881 (0.844) : [4] inspect (passed; r=0.920; p=0.300)
> 352.53 : final bay departure scheduled
[t: 345.575] pid 507/7 arrival event - queue len: 217
> 346.10 (0.839) : scheduled arrival for pid 508/7
* all bays busy; inserting pid 507/7 in queue
[t: 346.101] pid 508/7 arrival event - queue len: 218
> 349.94 (0.278) : scheduled arrival for pid 509/7
* all bays busy; inserting pid 508/7 in queue
[t: 346.390] pid 497/6 arrival event - queue len: 219
> 348.29 (0.753) : scheduled arrival for pid 510/6
* all bays busy; inserting pid 497/6 in queue
[t: 347.173] pid 281/3 departure event (bay 1)
[t: 347.173] handling pid 290/2 in bay 1
+ 1.093/1.093 (0.276) : [1] inspect (passed; r=0.805; p=0.180)
+ 2.676/1.583 (0.975) : [2] inspect (passed; r=0.232; p=0.180)
> 349.85 : final bay departure scheduled
[t: 348.294] pid 510/6 arrival event - queue len: 219
> 361.83 (0.133) : scheduled arrival for pid 511/6
* all bays busy; inserting pid 510/6 in queue
[t: 348.898] pid 282/3 departure event (bay 6)
[t: 348.898] handling pid 288/1 in bay 6
+ 1.287/1.287 (0.553) : [1] inspect (passed; r=0.275; p=0.180)
+ 2.325/1.038 (0.197) : [2] inspect (passed; r=0.638; p=0.180)
> 351.22 : final bay departure scheduled
[t: 349.423] pid 476/1 arrival event - queue len: 219
> 352.84 (0.656) : scheduled arrival for pid 512/1
* all bays busy; inserting pid 476/1 in queue
[t: 349.755] pid 271/6 departure event (bay 4)
[t: 349.755] handling pid 291/2 in bay 4
+ 1.677/1.677 (0.767) : [1] inspect (passed; r=0.832; p=0.140)
+ 3.331/1.655 (0.547) : [2] inspect (passed; r=0.644; p=0.140)
> 353.09 : final bay departure scheduled
[t: 349.777] pid 502/4 arrival event - queue len: 219
> 364.76 (0.168) : scheduled arrival for pid 513/4
* all bays busy; inserting pid 502/4 in queue
[t: 349.849] pid 290/2 departure event (bay 1)
[t: 349.849] handling pid 266/5 in bay 1
+ 1.898/1.898 (0.674) : [1] inspect (passed; r=0.906; p=0.260)
+ 3.870/1.972 (0.427) : [2] inspect (passed; r=0.856; p=0.260)
+ 5.967/2.097 (0.009) : [3] inspect (passed; r=0.899; p=0.260)
> 355.82 : final bay departure scheduled
[t: 349.944] pid 509/7 arrival event - queue len: 219
> 352.88 (0.376) : scheduled arrival for pid 514/7
* all bays busy; inserting pid 509/7 in queue
[t: 349.957] pid 506/2 arrival event - queue len: 220
> 363.54 (0.009) : scheduled arrival for pid 515/2
* all bays busy; inserting pid 506/2 in queue
[t: 350.071] pid 280/7 departure event (bay 2)
[t: 350.071] handling pid 289/7 in bay 2
+ 1.914/1.914 (0.136) : [1] inspect (passed; r=0.930; p=0.210)
+ 3.901/1.987 (0.869) : [2] inspect (passed; r=0.685; p=0.210)
+ 5.880/1.979 (0.791) : [3] inspect (passed; r=0.965; p=0.210)
> 355.95 : final bay departure scheduled
[t: 351.224] pid 288/1 departure event (bay 6)
[t: 351.224] handling pid 293/2 in bay 6
+ 1.335/1.335 (0.454) : [1] inspect (passed; r=0.316; p=0.300)
+ 2.651/1.315 (0.440) : [2] inspect (passed; r=0.735; p=0.300)
+ 4.082/1.432 (0.523) : [3] inspect (passed; r=0.735; p=0.300)
+ 5.344/1.262 (0.401) : [4] inspect (passed; r=0.558; p=0.300)
> 356.57 : final bay departure scheduled
```

## APPENDIX C: The Application Trace File (ATF)

```
[t: 352.530] pid 285/2 departure event (bay 3)
[t: 352.530] handling pid 295/7 in bay 3
+ 1.806/1.806 (0.981) : [1] inspect (passed; r=0.947; p=0.260)
+ 3.614/1.808 (0.972) : [2] inspect (failed; r=0.164; p=0.260)
+ 4.054/0.440 (0.740) : [2] repair
+ 5.090/1.036 (0.090) : [2] inspect (passed; r=0.978; p=0.130)
+ 7.086/1.995 (0.349) : [3] inspect (failed; r=0.152; p=0.260)
+10.229/3.144 (0.088) : [3] repair
+11.265/1.035 (0.097) : [3] inspect (passed; r=0.797; p=0.130)
> 363.79 : final bay departure scheduled
[t: 352.840] pid 512/1 arrival event - queue len: 218
> 352.90 (0.993) : scheduled arrival for pid 516/1
* all bays busy; inserting pid 512/1 in queue
[t: 352.880] pid 514/7 arrival event - queue len: 219
> 353.63 (0.780) : scheduled arrival for pid 517/7
* all bays busy; inserting pid 514/7 in queue
[t: 352.895] pid 516/1 arrival event - queue len: 220
> 354.22 (0.849) : scheduled arrival for pid 518/1
* all bays busy; inserting pid 516/1 in queue
[t: 353.086] pid 291/2 departure event (bay 4)
[t: 353.086] handling pid 296/2 in bay 4
+ 2.003/2.003 (0.323) : [1] inspect (passed; r=0.810; p=0.260)
+ 3.933/1.930 (0.567) : [2] inspect (passed; r=0.956; p=0.260)
+ 5.929/1.996 (0.346) : [3] inspect (failed; r=0.010; p=0.260)
+ 8.164/2.235 (0.132) : [3] repair
+ 9.174/1.010 (0.269) : [3] inspect (passed; r=0.455; p=0.130)
> 362.26 : final bay departure scheduled
[t: 353.627] pid 517/7 arrival event - queue len: 220
> 356.47 (0.387) : scheduled arrival for pid 519/7
* all bays busy; inserting pid 517/7 in queue
[t: 354.223] pid 518/1 arrival event - queue len: 221
> 355.10 (0.897) : scheduled arrival for pid 520/1
* all bays busy; inserting pid 518/1 in queue
[t: 354.385] pid 283/7 departure event (bay 5)
[t: 354.385] handling pid 297/7 in bay 5
+ 1.968/1.968 (0.675) : [1] inspect (failed; r=0.104; p=0.210)
+ 3.392/1.424 (0.415) : [1] repair
+ 4.344/0.953 (0.050) : [1] inspect (passed; r=0.432; p=0.105)
+ 6.277/1.932 (0.325) : [2] inspect (failed; r=0.124; p=0.210)
+11.660/5.383 (0.026) : [2] repair
+12.653/0.993 (0.856) : [2] inspect (passed; r=0.911; p=0.105)
+14.581/1.928 (0.281) : [3] inspect (passed; r=0.311; p=0.210)
> 368.97 : final bay departure scheduled
[t: 355.101] pid 520/1 arrival event - queue len: 221
> 356.26 (0.866) : scheduled arrival for pid 521/1
* all bays busy; inserting pid 520/1 in queue
[t: 355.646] pid 500/3 arrival event - queue len: 222
> 358.47 (0.456) : scheduled arrival for pid 522/3
* all bays busy; inserting pid 500/3 in queue
[t: 355.816] pid 266/5 departure event (bay 1)
[t: 355.816] handling pid 299/7 in bay 1
+ 1.455/1.455 (0.468) : [1] inspect (passed; r=0.824; p=0.360)
+ 2.805/1.350 (0.393) : [2] inspect (passed; r=0.385; p=0.360)
+ 4.281/1.476 (0.483) : [3] inspect (passed; r=0.769; p=0.360)
+ 5.243/0.962 (0.116) : [4] inspect (failed; r=0.200; p=0.360)
+ 7.803/2.559 (0.809) : [4] repair
+ 8.606/0.803 (0.576) : [4] inspect (passed; r=0.362; p=0.180)
> 364.42 : final bay departure scheduled
[t: 355.951] pid 289/7 departure event (bay 2)
[t: 355.951] handling pid 286/3 in bay 2
+ 1.902/1.902 (0.024) : [1] inspect (passed; r=0.500; p=0.210)
+ 3.884/1.982 (0.818) : [2] inspect (failed; r=0.082; p=0.210)
+ 5.272/1.388 (0.383) : [2] repair
+ 6.233/0.960 (0.209) : [2] inspect (passed; r=0.425; p=0.105)
+ 8.195/1.962 (0.622) : [3] inspect (passed; r=0.543; p=0.210)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 364.15 : final bay departure scheduled
[t: 356.263] pid 521/1 arrival event - queue len: 221
> 361.85 (0.502) : scheduled arrival for pid 523/1
* all bays busy; inserting pid 521/1 in queue
[t: 356.473] pid 519/7 arrival event - queue len: 222
> 357.17 (0.793) : scheduled arrival for pid 524/7
* all bays busy; inserting pid 519/7 in queue
[t: 356.568] pid 293/2 departure event (bay 6)
[t: 356.568] handling pid 301/3 in bay 6
+ 1.974/1.974 (0.420) : [1] inspect (passed; r=0.588; p=0.260)
+ 3.835/1.861 (0.796) : [2] inspect (passed; r=0.522; p=0.260)
+ 5.884/2.048 (0.172) : [3] inspect (passed; r=0.326; p=0.260)
> 362.45 : final bay departure scheduled
[t: 357.170] pid 524/7 arrival event - queue len: 222
> 363.64 (0.116) : scheduled arrival for pid 525/7
* all bays busy; inserting pid 524/7 in queue
[t: 358.302] pid 505/5 arrival event - queue len: 223
> 369.01 (0.375) : scheduled arrival for pid 526/5
* all bays busy; inserting pid 505/5 in queue
[t: 358.471] pid 522/3 arrival event - queue len: 224
> 358.97 (0.871) : scheduled arrival for pid 527/3
* all bays busy; inserting pid 522/3 in queue
[t: 358.969] pid 527/3 arrival event - queue len: 225
> 360.57 (0.642) : scheduled arrival for pid 528/3
* all bays busy; inserting pid 527/3 in queue
[t: 360.566] pid 528/3 arrival event - queue len: 226
> 364.06 (0.379) : scheduled arrival for pid 529/3
* all bays busy; inserting pid 528/3 in queue
[t: 361.831] pid 511/6 arrival event - queue len: 227
> 364.42 (0.680) : scheduled arrival for pid 530/6
* all bays busy; inserting pid 511/6 in queue
[t: 361.851] pid 523/1 arrival event - queue len: 228
> 363.15 (0.852) : scheduled arrival for pid 531/1
* all bays busy; inserting pid 523/1 in queue
[t: 362.260] pid 296/2 departure event (bay 4)
[t: 362.260] handling pid 287/6 in bay 4
+ 1.995/1.995 (0.349) : [1] inspect (failed; r=0.086; p=0.260)
+ 2.155/0.160 (0.911) : [1] repair
+ 3.125/0.970 (0.532) : [1] inspect (passed; r=0.486; p=0.130)
+ 5.037/1.911 (0.630) : [2] inspect (passed; r=0.333; p=0.260)
+ 7.113/2.076 (0.080) : [3] inspect (passed; r=0.589; p=0.260)
> 369.37 : final bay departure scheduled
[t: 362.451] pid 301/3 departure event (bay 6)
[t: 362.451] handling pid 300/7 in bay 6
+ 1.167/1.167 (0.382) : [1] inspect (passed; r=0.799; p=0.180)
+ 2.536/1.369 (0.670) : [2] inspect (failed; r=0.036; p=0.180)
+ 7.394/4.858 (0.011) : [2] repair
+ 8.164/0.770 (0.915) : [2] inspect (failed; r=0.014; p=0.090)
+ 8.345/0.181 (0.783) : [2] repair
+ 8.866/0.521 (0.202) : [2] inspect (passed; r=0.339; p=0.090)
> 371.32 : final bay departure scheduled
[t: 363.151] pid 531/1 arrival event - queue len: 227
> 366.57 (0.656) : scheduled arrival for pid 532/1
* all bays busy; inserting pid 531/1 in queue
[t: 363.544] pid 515/2 arrival event - queue len: 228
> 369.20 (0.142) : scheduled arrival for pid 533/2
* all bays busy; inserting pid 515/2 in queue
[t: 363.640] pid 525/7 arrival event - queue len: 229
> 368.09 (0.227) : scheduled arrival for pid 534/7
* all bays busy; inserting pid 525/7 in queue
[t: 363.795] pid 295/7 departure event (bay 3)
[t: 363.795] handling pid 284/4 in bay 3
+ 1.933/1.933 (0.329) : [1] inspect (passed; r=0.248; p=0.210)
+ 3.931/1.998 (0.979) : [2] inspect (passed; r=0.649; p=0.210)
+ 5.923/1.993 (0.925) : [3] inspect (passed; r=0.792; p=0.210)
```

## APPENDIX C: The Application Trace File (ATF)

```
> 369.72 : final bay departure scheduled
[t: 364.060] pid 529/3 arrival event - queue len: 229
> 364.14 (0.978) : scheduled arrival for pid 535/3
* all bays busy; inserting pid 529/3 in queue
[t: 364.139] pid 535/3 arrival event - queue len: 230
> 365.43 (0.699) : scheduled arrival for pid 536/3
* all bays busy; inserting pid 535/3 in queue
[t: 364.145] pid 286/3 departure event (bay 2)
[t: 364.145] handling pid 298/2 in bay 2
+ 1.123/1.123 (0.318) : [1] inspect (passed; r=0.375; p=0.180)
+ 2.489/1.366 (0.666) : [2] inspect (failed; r=0.179; p=0.180)
+ 4.259/1.770 (0.327) : [2] repair
+ 4.744/0.485 (0.100) : [2] inspect (passed; r=0.705; p=0.090)
> 368.89 : final bay departure scheduled
[t: 364.416] pid 530/6 arrival event - queue len: 230
> 368.95 (0.508) : scheduled arrival for pid 537/6
* all bays busy; inserting pid 530/6 in queue
[t: 364.422] pid 299/7 departure event (bay 1)
[t: 364.422] handling pid 294/5 in bay 1
+ 1.911/1.911 (0.110) : [1] inspect (passed; r=0.354; p=0.210)
+ 3.868/1.957 (0.575) : [2] inspect (passed; r=0.678; p=0.210)
+ 5.783/1.914 (0.141) : [3] inspect (passed; r=0.499; p=0.210)
> 370.20 : final bay departure scheduled
[t: 364.762] pid 513/4 arrival event - queue len: 230
> 369.88 (0.544) : scheduled arrival for pid 538/4
* all bays busy; inserting pid 513/4 in queue
[t: 365.000] close doors event
```