PRELIMINARY DESIGN REVIEW

Aircraft Repair Model Simulation

October 11, 2000

TEAM 9: GOLDRUSH

Nick Capron

Dave Stone	Carina Lansing
Russell Swannack	Kyle Klicker
Miguel Vilchez	Jonathan McPherson
Kelly Yearout	Andy Reynolds

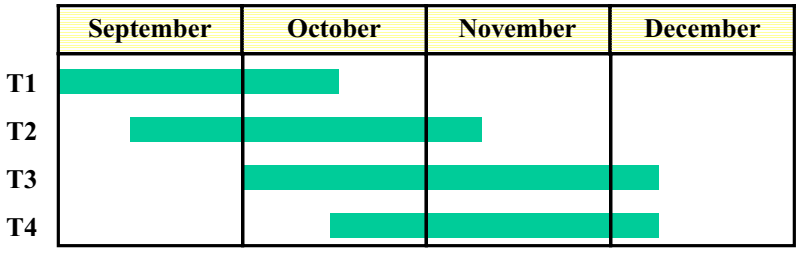
Agenda

1. Team Member Introduction and Responsibilities
2. Requirements
3. Schedule and Tasks
4. Assumptions
5. Context Diagram
6. High Level DFD
7. Extended Modules
8. Three Prototype Test Cases
9. Traceability Approach
10. Scope Limitations
11. Current Status
12. Action Items

Requirements

- Model aircraft repair process using stand-alone, event-based simulation
- Represent seven aircraft types
- Offer three queuing disciplines
- Provide Graphical User Interface
- Utilize Simlib library
- Offer insert and delete capabilities
- Develop in Java

Schedule and Tasks



T1: Understand Problem/Requirements
(Client discussions, Simlib study, Java review, SRS development)

T2: Design ARMS Application
(GUI design, PDR, CDR, Design Notebook)

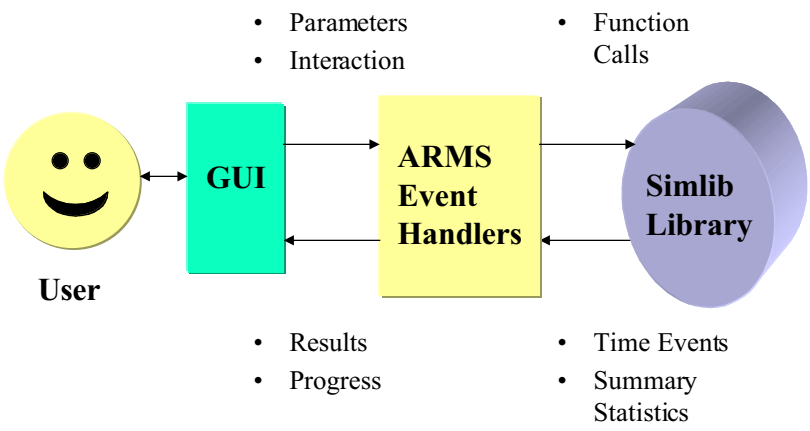
T3: Code ARMS Application
(GUI, Simlab extensions, event-handlers, User Manual)

T4: Test/Validate ARMS Application
(Test cases, requirements validation, code modifications)

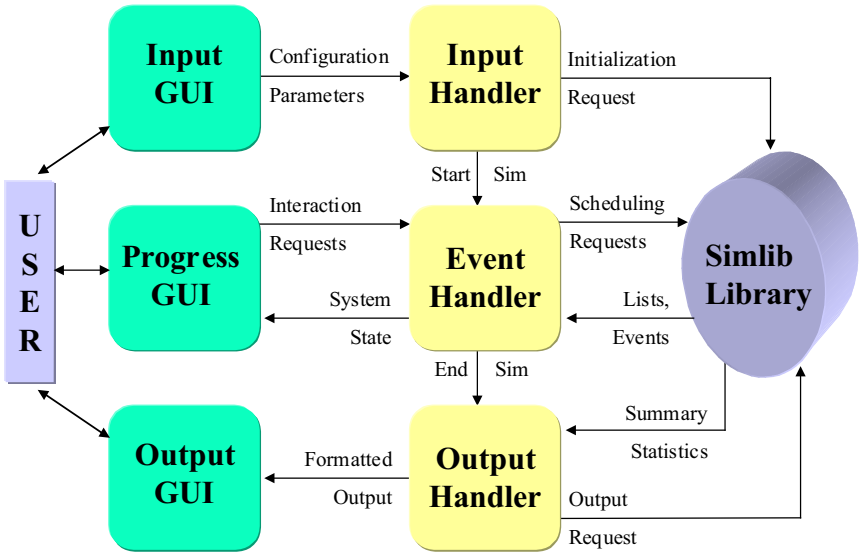
Assumptions

- Maximum of 10 service bays
- User responsible for output data storage
- Not safety or time critical
- Existing Simlib functions are valid
- Pre-defined aircraft parameters are valid

Context Diagram



High Level DFD



Extended Modules

Modifications to Simlib API:

(Necessary to interface with Java)

- Add modifier functions to insert/delete from a list
- Add functions to access global variables and constants
- Add function to retrieve list contents

Prototype Test Cases

1) **Queuing Scenario #1**
   (One FIFO queue)

2) **Queuing Scenario #2**
   (Two FIFO queues, non-preemptive widebody priority)

3) **Queuing Scenario #3**
   (Two FIFO queues, two sets of service bays)

- Valid and invalid input ranges
- GUI functionality and usability
- Output validation

Traceability Approach

- Categorize/number requirements
- Identify testing method for each requirement
- Develop Requirements Traceability Matrix (RTM)
- Link DFD modules to RTM requirement categories
- Test modules against assigned requirements

Scope Limitations

Limitations:

- GUI will be simple
  (All necessary information will be displayed, but an animated process flow diagram is unlikely)
- Limit bays to 10
  (To avoid overcrowding screen)

Impact on Schedule:

- No impact

Open Issues

- How to display changing system state without too much complexity
- How to seed the Simlib random number generator

Current Status

Accomplishments:

- Simlib/Java interface
- Project Plan, Draft SRS
- Top-level design
- RTM

Plans:

- GUI finalization
- Second-level design
- Design documents
- Coding

Problems:

- Network outages
- Confusing guidelines
- Changing Specs
- Limited experience

Action Items