NISTIR 7320

User Interface of Simulation of the Shipbuilding Operations

Guodong Shao
Charles McLean
Swee Leong

National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

NISTIR 7320

User Interface of Simulation of the Shipbuilding Operations

Guodong Shao
Charles McLean
Swee Leong
Manufacturing Systems Integration Division
Manufacturing Engineering Laboratory

May 2006



U.S. DEPARTMENT OF COMMERCE
Carlos M. Gutierrez, Secretary
TECHNOLOGY ADMINISTRATION
Robert Cresanti, Under Secretary of Commerce for Technology
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
William Jeffrey, Director

USER INTERFACE OF SIMULATION OF THE SHIPBUILDING OPERATIONS

Guodong Shao
Charles McLean
Swee Leong

Manufacturing Simulation and Modeling Group
National Institute of Standards and Technology (NIST)
100 Bureau Drive, MS 8260
Gaithersburg, Maryland 20899, U.S.A.

ABSTRACT

This paper discusses the development of the user interface for a simulation of the shipbuilding operations. It presents an overview of a generic simulation of shipbuilding operations, the simulation system's user profiles, and the user interface screen designs. The simulation model of the shipbuilding operations can be used as a tool to analyze the schedule impact of new workload, evaluate production scenarios, and identify resource problems. The animated user interface display allows the users to use interactively the simulation model to evaluate and analyze the production process model of the shipyard, to view and manipulate the production data. The simulation user interface shows expected results of inserting new technologies or equipment into the shipyard, particularly with respect to operating costs and schedule impact.

KEY WORDS

User interface design, Shipbuilding operations, Manufacturing simulation

DISCLAIMER

No approval or endorsement of any commercial product by the National Institute of Standards and Technology is intended or implied. The work described was funded by the United States Government and is not subject to copyright.

ACKNOWLEDGMENTS

Work described in this paper was sponsored by the National Shipbuilding Research Program (NSRP) Advanced Shipbuilding Enterprise (ASE) and the NIST Systems Integration for Manufacturing Applications (SIMA) Program.

1 INTRODUCTION

This paper discusses a user interface for the shipbuilding manufacturing simulation system. The simulation system was used to simulate shipbuilding operations and manipulate required data. The dynamically displayed user interface allows the user to use

the simulation model interactively to evaluate and analyze the production process model of the shipyard. Scientists and engineers working on the Manufacturing Interoperability Program at the National Institute of Standards and Technology (NIST) performed the research presented in this paper.

The National Shipbuilding Research Program (NSRP) is a program to achieve significant technology and process improvements in the U.S. shipbuilding industry. NRSP is being carried out as collaboration among U.S. shipyards, government, industry, and academia. The Shipyard Production Process Technologies is a major initiative that addresses all production processes used to transform raw material, components, and equipment into completed products (NSRP 2001). The simulation of shipbuilding operations presented in this paper was carried out as part of this initiative.

The objective of the simulation model for shipbuilding operations is to provide data to support shipyard management decisions including (McLean and Shao 2001):

- Analysis of schedule impact due to additional projects with differing production scenarios
- Identification of labor resource conflicts by craft and skill level
- Analysis of cost and effects of exercising overtime, new hire, and/or subcontractor to cover labor shortages
- Analysis of cost tradeoffs of holding over employees without work, versus lay-off, rehires, and new hires
- Prediction of optimum staffing levels-based sales and labor demand forecasts
- Demonstration of the expected results of inserting new technology or equipment into a shipyard, particularly with respect to operating costs and schedule impact
- Visual display of work location, resources, and identification resource constraints and conflicts between competing jobs (including cranes and forklifts)
- The archive of simulation runs as files to preserve historical data.

2 DEVELOPMENT SOFTWARE OVERVIEW

The development software used to implement the simulation system in this paper is ProModel 6.0. ProModel is a PC-based discrete event simulation software that can be used for evaluating, planning, or re-designing manufacturing, warehousing, and logistics systems. The simulation software allows users to build a graphical representation of an application system and test it in a variety of scenarios to provide better solution for the problems in the organization. The graphical animation and reports are useful tools for visualizing, understanding, and improving the real system. The development simulator includes functions for developing and executing the simulation models. A separate graphics library was used to generate graphical displays. A Microsoft COM interface is included to allow the simulator to execute external code (ProModel 1996).

3 USER PROFILE OF THE SHIPBUILDING SIMULAITON

The potential users of this simulation system are from several categories. Understanding the intended user, their tasks and needs, the organization where they work, and the environment where the system may be used helps the design of the user interface. The potential users are identified as follows:

- *Simulation Analyst*: They are the core users of the system. They are responsible for executing the model and analyzing the simulation results on a daily basis. They are perhaps involved in the system development process from the beginning. They understand the problem, are the source of some requirements assumptions for the system, and might help in the data collection. They can verify the model according to each scenario, help to make the decisions concerning the length of the simulation run, the number of runs needed, and the initial conditions. They are the knowledgeable intermittent users of the simulation system and might become expert frequent users after using the simulation for a period of time. The Simulation Analyst may have an IT and manufacturing background in education or in working experience.
- *Process Engineer*: They are responsible for producing the process plan for all operations of the shipbuilding. They are responsible for improving manufacturing capabilities of the shipbuilding process, providing manufacturing quality engineering services to ensure current new products are produced at the highest level of quality, as economically, efficiently, and safely as possible. They are responsible for the "cradle to grave" process of a job. They provide the time study and determine rules for the simulation. They are the knowledgeable intermittent users.
- *Scheduler*: They are responsible for producing and managing the overall shipyard schedules. In the simplest case, these persons ultimately make all changes to the schedule. They are the knowledgeable intermittent users.
- *Area Supervisor*: They are responsible for managing production operations in different areas of the shipyard. They might use the simulator to view simulation displays, track progress, and identify potential problems (e.g., expected job delays). In circumstances where problems are detected, the simulator's what-if capabilities can be used to analyze different options, and make suggestions to the scheduler. They may be categorized as knowledgeable intermittent users.
- *Project Manager*: They are responsible for managing a particular set of jobs. They might use the simulator in the same ways as area supervisors. They are experienced in managing personnel resources to keep a project moving to completion, are able to handle and manage multiple assignments concurrently, have hands-on problem solving, and a strong background in shipbuilding operations' processes. They are the knowledgeable intermittent users.
- *Shipyard/workshop Manager*: They are responsible for managing the whole workshop, and coordinate the processes of the shipyard. With the help of the simulation system, they can accomplish their job, which is to manage the employees and all day-to-day activities to ensure timely execution of all

requirements. They can also plan and execute the expansion goals to increase shipyard and work shop capability; Moreover, to leverage supply chain processes to reduce material cost and increase in sourcing activities, increase equipment utilization and labor efficiency, drive planning and scheduling activities to level load workflow and balance with current capacity, optimize staffing resources by providing clear direction, communication, performance management, and corrective action when necessary. They normally have degrees in Business or Engineering, and years of leadership experience. However, they do not have simulation experience or background, so they are categorized as novice users.

- *Sale Staff*: They are those persons who are responsible for negotiating new orders. They can use the simulator to determine due dates that are consistent with actual production status and load. In addition, they are responsible for developing the strategic sales and marketing objectives for the shipyard. They need to interact with internal and external customers, have the capability to adjust for the new culture. They may be categorized as novice users of the simulation system.
- *IT Staff*: They are responsible for installing and maintaining the computer hardware and software systems. They are not frequent users of the simulation system. They may be categorized as novice users of the system.

For those novice or first time users such as production staff, although they are professionals who understand the terminology of shipbuilding operations, they might have very shallow knowledge of the simulation system and the interface concept. These users might have anxiety about using computers that inhibit learning. Therefore, the design of the system interface is a challenge. It is better to use the terminologies that they are familiar with. The numbers of the actions should be small, so that these users can carry out simple tasks successfully and thus reduce anxiety, build confidence, and gain positive reinforcement. Informative feedback should be helpful and constructive. Specific error messages may need to be provided when they make mistakes. Clearly designed manuals and systematic tutorials may be effective. They need to be trained thoroughly before they use the system. After they use the system for a while, they may become knowledgeable intermittent users (Cooper 2002).

For those knowledgeable intermittent users such as simulation analyst and process engineer, they have stable task concepts and broad knowledge of the system concept and terminology. When they first start to use the system, retaining the structure and the location of features might still be hard, so some training may be necessary. The burden on their memories will be lightened by orderly structure in the menu, consistent terminology, and high interface appearance that emphasizes recognition rather than recall. Consistent sequences of actions, meaningful messages, and guides to frequent patterns of usage will help to rediscover how to perform the tasks properly. User manuals will be helpful. After they become expert power users, they are thoroughly familiar with the system and the interface, they might like to have quicker response times, brief and non-distracting feedback, and the capacity to carry out actions with just a few keystrokes or selections.

Because of the several communities of users for the system, the design effort of the user interface will be multiplied. The eight “Golden” Principles for design user interface according to Shneiderman are (Shneiderman 1998):

1. Strive for consistency
Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent commands should be employed throughout.

2. Enable frequent users to use shortcuts
As the frequency of use increases, so do the user's desires to reduce the number of interactions and to increase the pace of interaction. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.

3. Offer informative feedback
For every operator action, there should be some system feedback. For frequent and minor actions, the response can be modest, while for infrequent and major actions, the response should be more substantial.

4. Design dialog to yield closure
Sequences of actions should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans and options from their minds, and an indication that the way is clear to prepare for the next group of actions.

5. Offer simple error handling
As much as possible, design the system so the user cannot make a serious error. If an error is made, the system should be able to detect the error and offer simple, comprehensible mechanisms for handling the error.

6. Permit easy reversal of actions
This feature relieves anxiety, since the user knows that errors can be undone; it thus encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.

7. Support internal locus of control
Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.

8. Reduce short-term memory load
The limitation of human information processing in short-term memory requires that displays be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.

4 USER INTERFACE GOAL

Careful determination of the user community and of the benchmark set of tasks is the basis for establishing human-factors' goals. Based on the user profiles discussed above, the general user's goals are as simple as the following (Shneiderman 1998):

- Not looking stupid
- Knowing what's going on
- Knowing how to use the system
- Remembering how to use the system
- Making sure that displays are clear enough to understand
- Having information clearly, meaningfully, and consistently categorized on different screens
- Minimizing the user input
- Minimizing the chance of errors
- Involving no dangers even if mistakes are made
- Getting useful feedback from the system
- Having visually pleasing displays

The above user goals are translated to the Shneiderman's five measurable goals for evaluation. They are listed below:

- Time to learn: Easy and quick, about two minutes for a typical member of the user community.
- Speed of performance: five seconds to carry out the benchmark tasks.
- Rate of errors by users: the goal for the error rate by a user is low.
- Retention over time: depends on the user level and frequency of use of the system, since the time to learn is short, the retention over time is high.
- Subjective satisfaction: the goal of this aspect is moderate.

The simulation system is used to simulate the shipbuilding operations and manipulate required data. This includes:

- Execution of orders, jobs, and tasks
- Emulation of the production calendar and shift schedule
- Enforcement of precedence constraints in process plan routing and operation sheets
- Sequencing of steps in the operation sheet associated with an individual task
- Availability of resources: Space, cranes, labors
- Allocation of resources to tasks

The user interface includes the following features:

- Managing different display screens based on different functionalities
- Observing simulation runs
- Monitoring location, job, task status dynamically
- Displaying simulation output results
- Generating custom reports of simulation results

5 SIMULATION SYSTEM DEVELOPMENT

The term "Simulation system" is used to refer to the simulation model, the simulation engine, a user interface system, and associated data files. The main simulation system components are described below:

Control Logic: The control logic is customized code and data developed to model the production process of the shipyard. Most manufacturing simulations concentrate on modeling material flow. This shipbuilding simulation project is different in that it concentrates on modeling the flow of work through various planning and processing stages. Users can analyze and manipulate the production data through a spreadsheet type of user interface. The control logic for the simulation model manages the execution of the following modules (McLean and Shao 2001):

- Graphical user interface
- Data input
- Job and task management
- Production area allocation and space management
- Labor allocation
- Resource allocation
- Calendar and clock
- Report generator

The major control logic is discussed briefly as follows: Jobs and tasks are defined as logical entities in the simulation flow to simulate the production process. A job entity is created for each subproject within the project, e.g., a steel fabrication job, a pipe fabrication. Each job is decomposed into many tasks. Figure 1 shows the relationships between and among project, jobs, and tasks. Job entities are relatively independent of each other. Tasks within a job are dependent upon each other by precedence relationships. A set of unique task entities is created for each job according to the critical path network. Jobs and tasks flow through a series of locations within the simulation model as they are processed. Figure 2 shows the entity – location relation among project, job, and task.

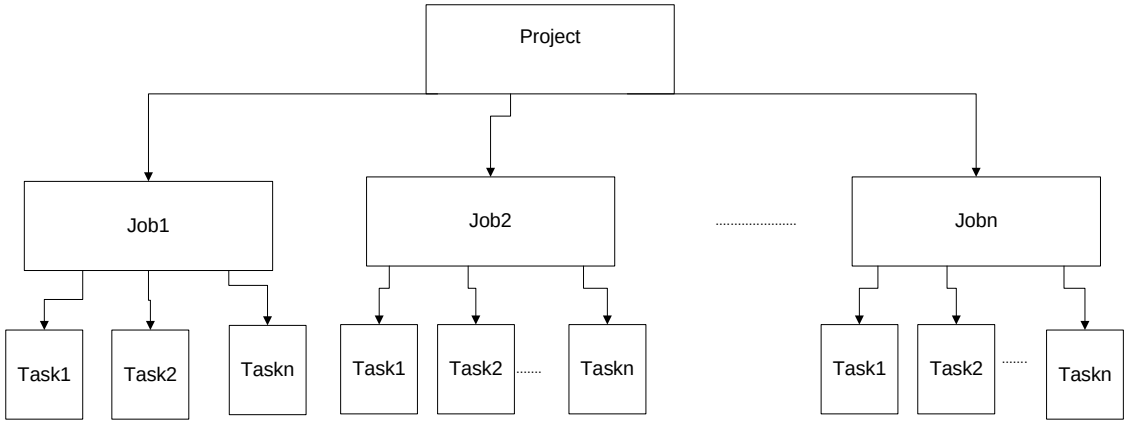


Figure 1: Relation among Project, Job, and Task

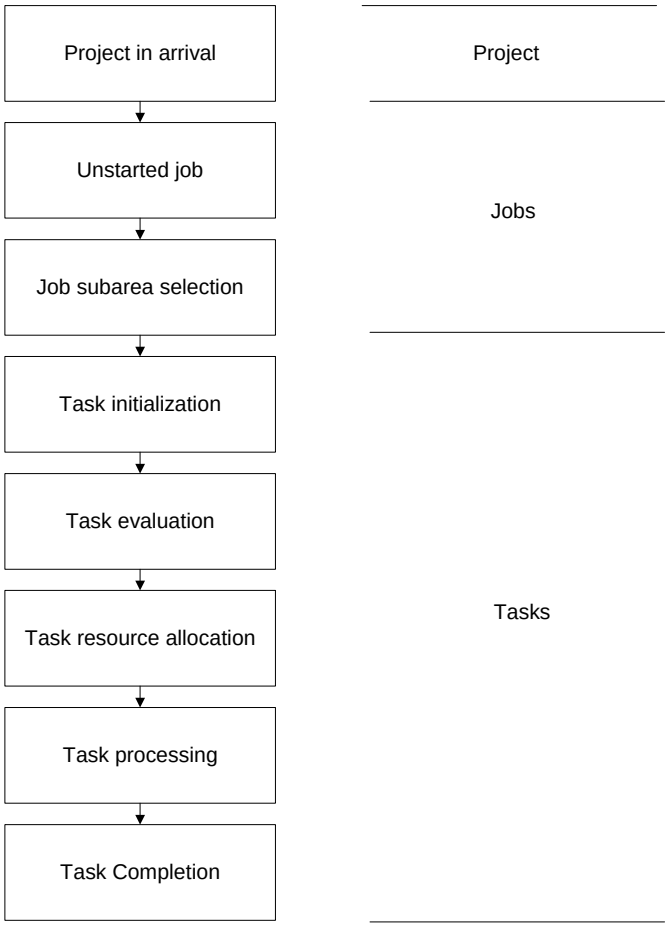


Figure 2: Entity – Location relation among Project, Job, and Task

Job entities are created at an *Arrival* location within the simulation model. Once the job data is initialized from data files, it is moved to the *Unstarted_jobs* location. At the *Unstarted_jobs* location, if current date is later than or equal to the job start date, the job is moved to the *Job_area_selection* location. At the *Job_area_selection* location, the production area where the job will be processed is determined. If the allocation is successful, the job moves to the *Task_initialization* location for the selected production area. A unique set of "task" locations is associated with each production area. At the *Task_initialization* location, preparation, decomposition, and calculation for the job and its tasks are being done. A unique task entity is created for each task in the job, and then moved to the *Task_evaluation* location for the selected working area. At the *Task_evaluation* location, a determination is made as to whether the task is ready to start based on the precedent conditions. Ready tasks are moved to the *Task_labor_allocation* location. At the *Task_labor_allocation* location, workers are allocated from the labor pool to each task. At the *Task_processing* location, the task waits until either the work content is completed or the shift ends. If the work completes before the end of the shift,

the task is moved to the *Task_completion* location. If the shift ends before the work is completed, the task is moved back to the *Task_labor_allocation* location.

User Interface: The dynamically updated user interface allows the users to use interactively the simulation model to evaluate and analyze the production process model of the shipyard. The multi-screen displays are used to show the progress of different aspects of the simulation. The screens show the overall status of the simulation in terms of several logical views of the data. Each screen displays the shift calendar and clock data. Screen information is updated when selecting to view this screen or the overall screen. See Appendix A for the logical diagram for the screen display. The nine user interface screens showed in Figure 3 are briefly described below:

- ***Simulation Status*** – shows the status of jobs and tasks from the project plan flows through simulation processing locations. Each position shows a count of the jobs or tasks currently in that state.
- ***Subarea Allocation*** – lists the individual job numbers for a particular subarea location. Subareas are divided into slots. The slots number and location number for the job are also shown.
- ***Location Status*** **-** lists the individual job numbers for a particular simulation processing location.
- ***Job Status*** – shows the data for an individual job that is being processed. There is also a job drawing or picture for the selected job being displayed on this screen.
- ***Task Status*** – shows the data for an individual task as well as the task precedent status for the selected task in the selected job.
- ***Resource Allocation Status*** –shows the status of resources, e.g., three types of cranes and five types of labors that are actively working and the number of resources currently available.
- ***Production Schedule*** – shows a list of the scheduled jobs according to the mode of selection, which is sorted by either job identifier starting with selected job identifier or by lateness rank for the selected area.
- ***Part and Inventory Status*** – shows the run time status of parts and inventory and in process.
- ***Statistical Data Summary*** - shows the summary statistics data for production and resources' utilization.

- ***Production Calendar, Clock, and Shift Management Panels*** – shows a production clock that gives elapsed time in days, shifts, hours and minutes from the simulation start. It provides production calendar information. The time unit is minutes. The current calendar assumes a seven-day work schedule.

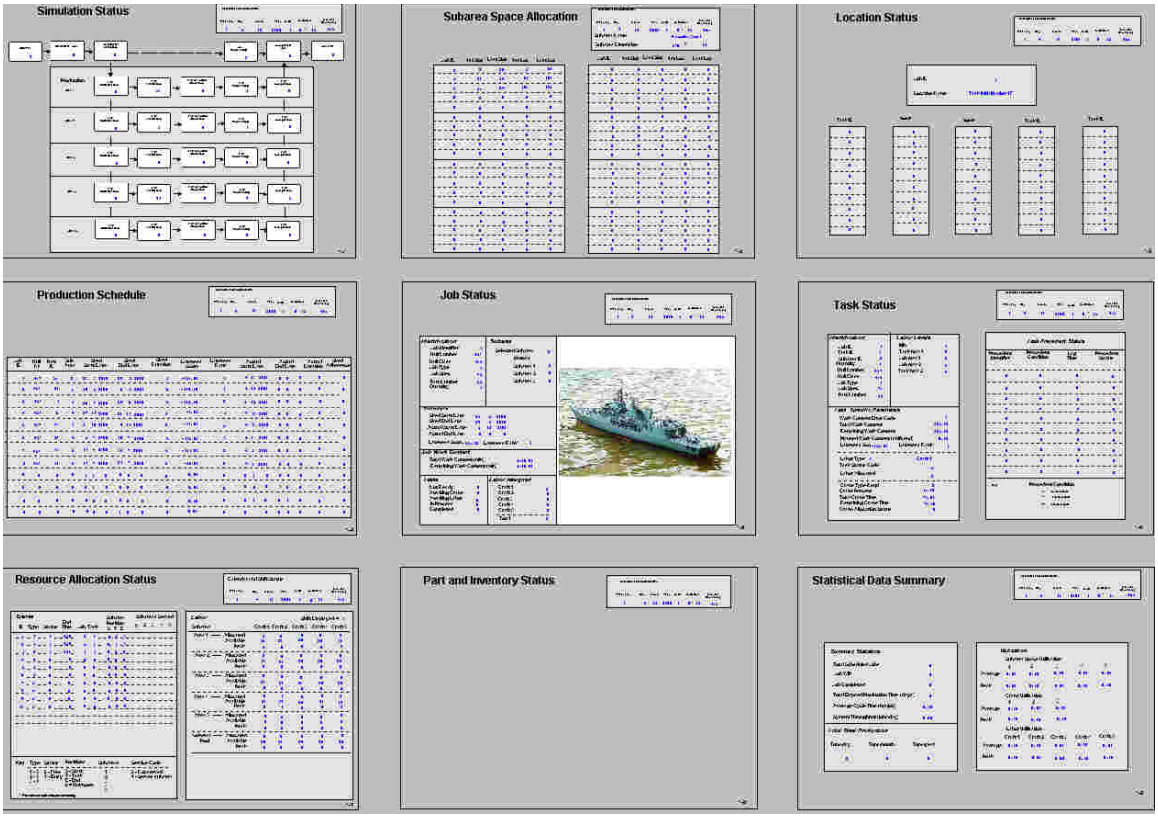


Figure 3: Overall Screen

Simulation Data Management: The shipyard input data for the simulation may originally reside in Microsoft Excel spreadsheets, Microsoft Project files, or people's heads. An XML structure of the shipyard data interface is used as the input data format. ProModel reads the XML data and translates it to a format of the internal representation of ProModel. In this version of ProModel, all data must essentially be represented as an integer or real numbers. The following major sets of data are required to configure and run the shipyard simulation models:

- Schedule file
- Shipbuilding project plans
- Labor configuration data
- Calendar and shift information

Figure 4 shows that all shipyard data files must be processed into appropriate input file formats for ProModel.

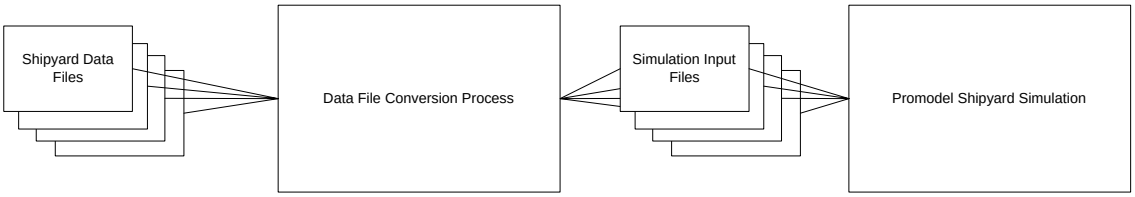


Figure 4: Input data conversion process

The preprocessor is a NIST-developed module, written in the C programming language that is used to read the shipyard data into the simulation model. The data files are read into the simulator during the initialization phase of the simulation run. Three user-required formatted output reports are generated after the simulation run:

Jobs summary report - Contains a log of information for jobs and those tasks that make up the jobs from the project plan. Not only the identification information but also the information calculated and allocated is listed.
*Labor log report - C*ontains a log of labor information for the task, including the start day/month/year, shift identifier, labor status, labor start/end min, job/task identifier, labor type, local pool, global pool count, labor source.
Space log report - Contains a log of sub area information for the task, including the simulation time, start day/month/year, shift identifier, shift time, pier identifier, percentage utilization and portion in use.

6 USER INTERFACE SCREEN DESIGNS

Figure 5 shows the default screen after the execution of the simulation model – *Simulation status* screen; it shows the status of the jobs and the tasks from the project plan flows through the following simulation-processing locations: Arrival, Unstarted Jobs, Subarea selection, Task initialization, Task Evaluation, Task Resource Allocation, Task Processing, Task Completion, Job Processing, Completed Jobs, and Jobs Exit. Each position shows a count of the jobs or tasks currently in that state. Figure 6 is the *Subarea space allocation* screen; a table is referred to generically as a subarea. This screen is responsible for displaying the dynamically allocating and releasing chunks of workspace within the subarea for jobs. The subarea is divided up into slots. Each slot consumes 3 feet of the table's length dimension. Figure 7 is the *Production schedule* screen; it shows scheduled, actual start, and end dates as well as the lateness for jobs listed by different display modes. Figure 8 is the *Job status* screen; it shows information including job identifier, schedules, working area, work contents and labor information as well as the drawing or picture of the job. Figure 9 is the *Task status* screen; it shows information including the job information, which the task belongs to, task parameters and resources information, and includes the precedent conditions with other tasks. Figure 10 is the *Resource allocation status* screen, which includes labor and crane allocation. Number of workers assigned refers to the number of workers allocated for the current shift. If a crane

is required, the crane type is indicated. A real value between zero and one indicates the percentage time of the total work content of the task that requires a crane. Figure 11 is the *Statistical data summary* screen. It shows statistical data for production and resource utilizations both for real time and at the end of the simulation execution.

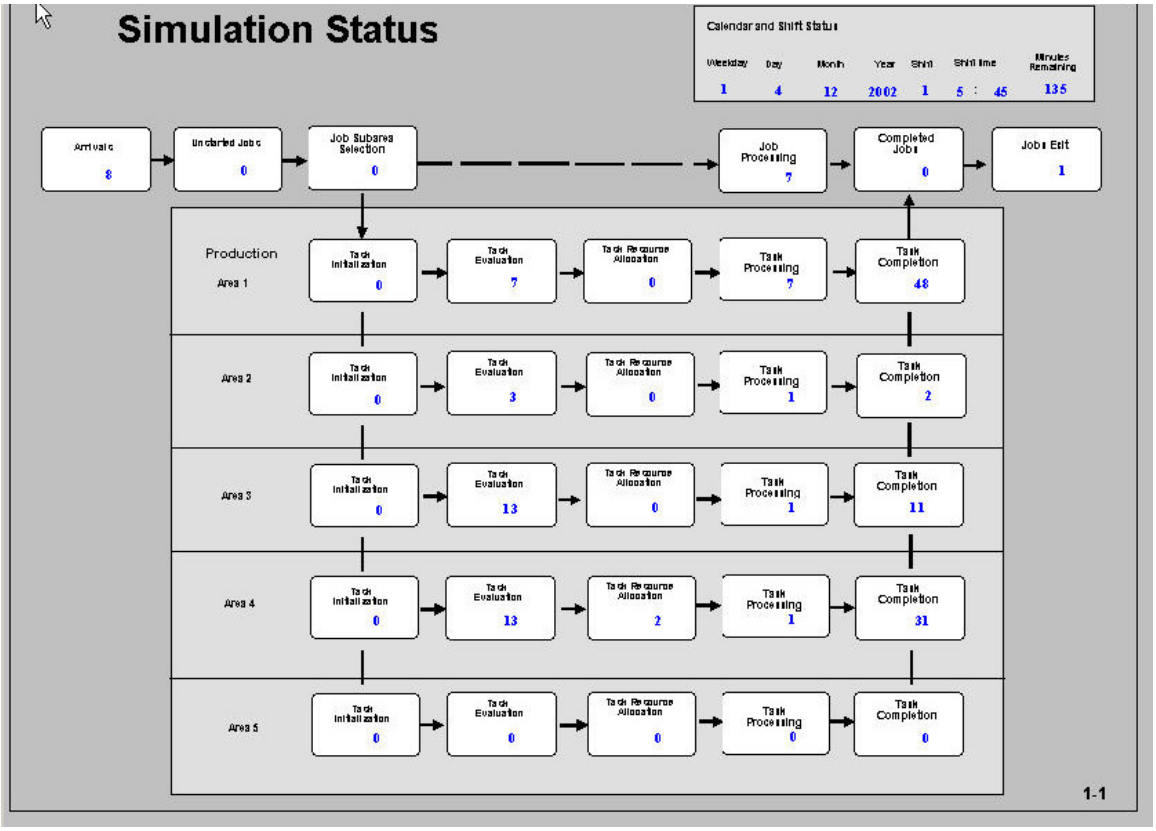


Figure 5: Simulation Status Screen

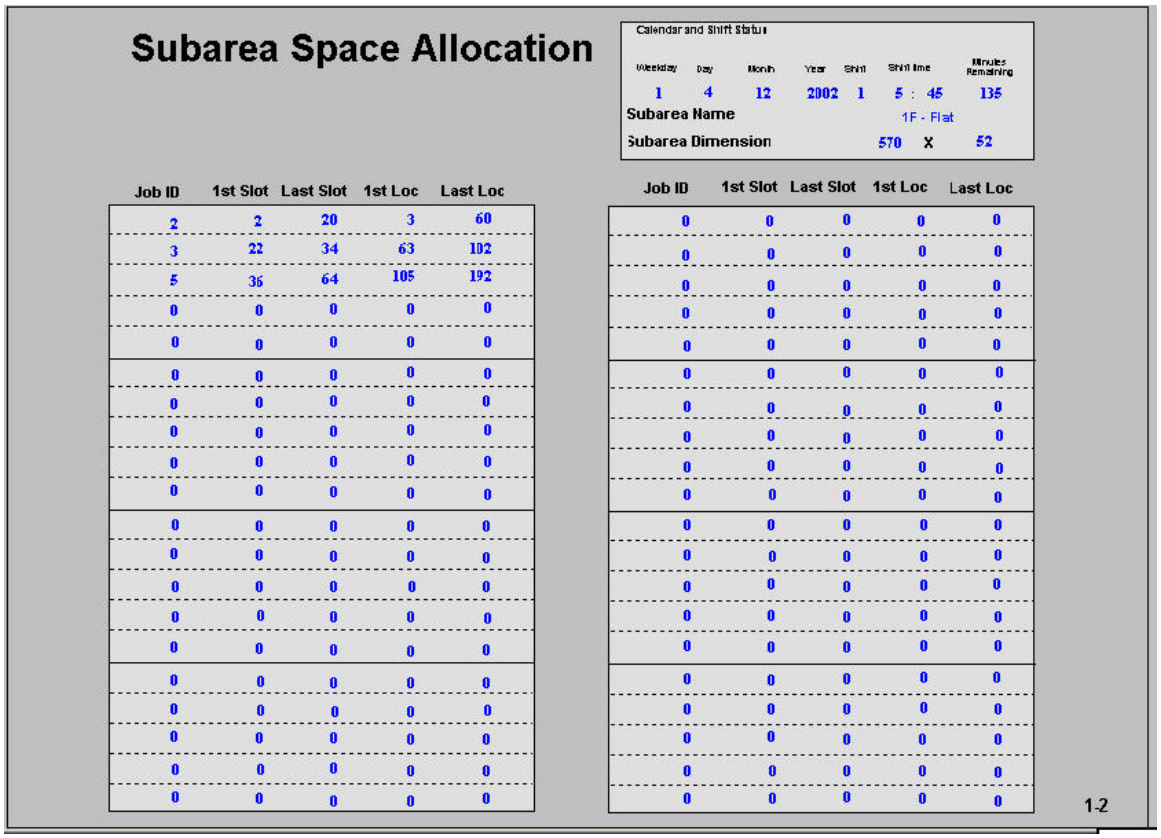


Figure 6: Production Subarea Space Allocation Screen

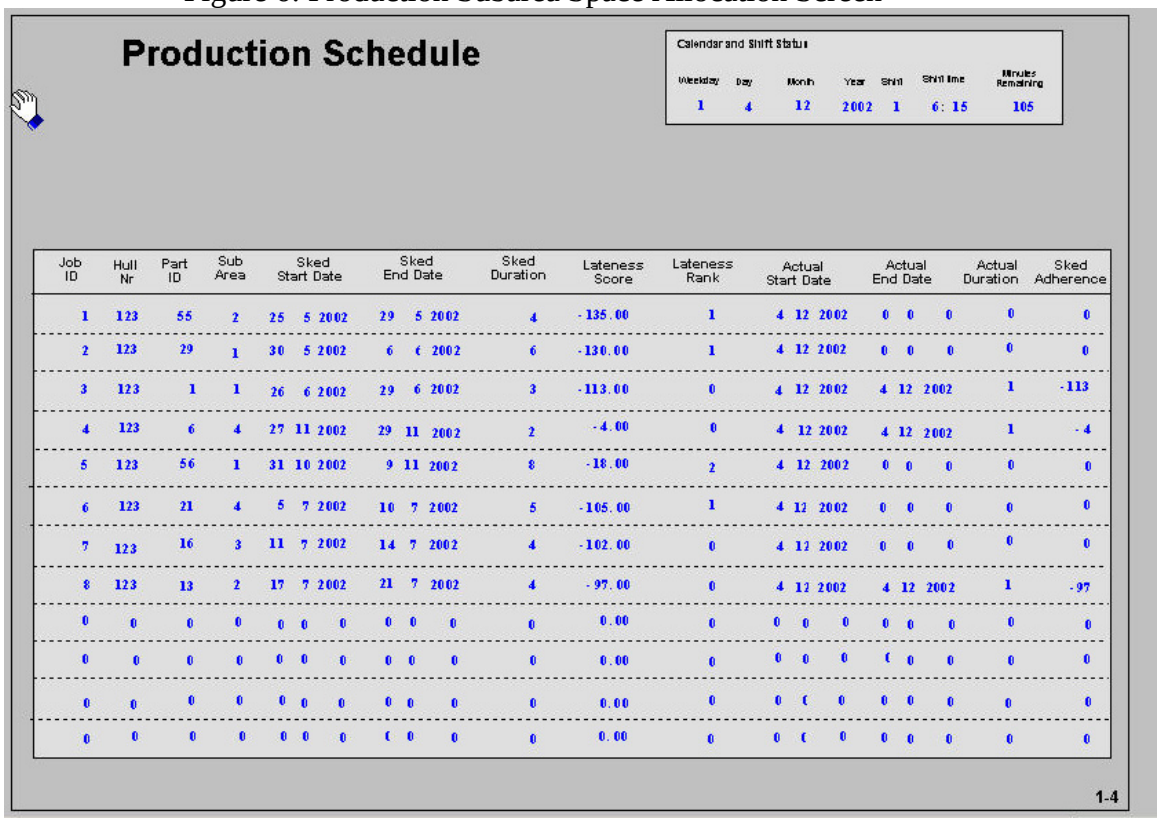


Figure 7: Production Schedule Status Screen

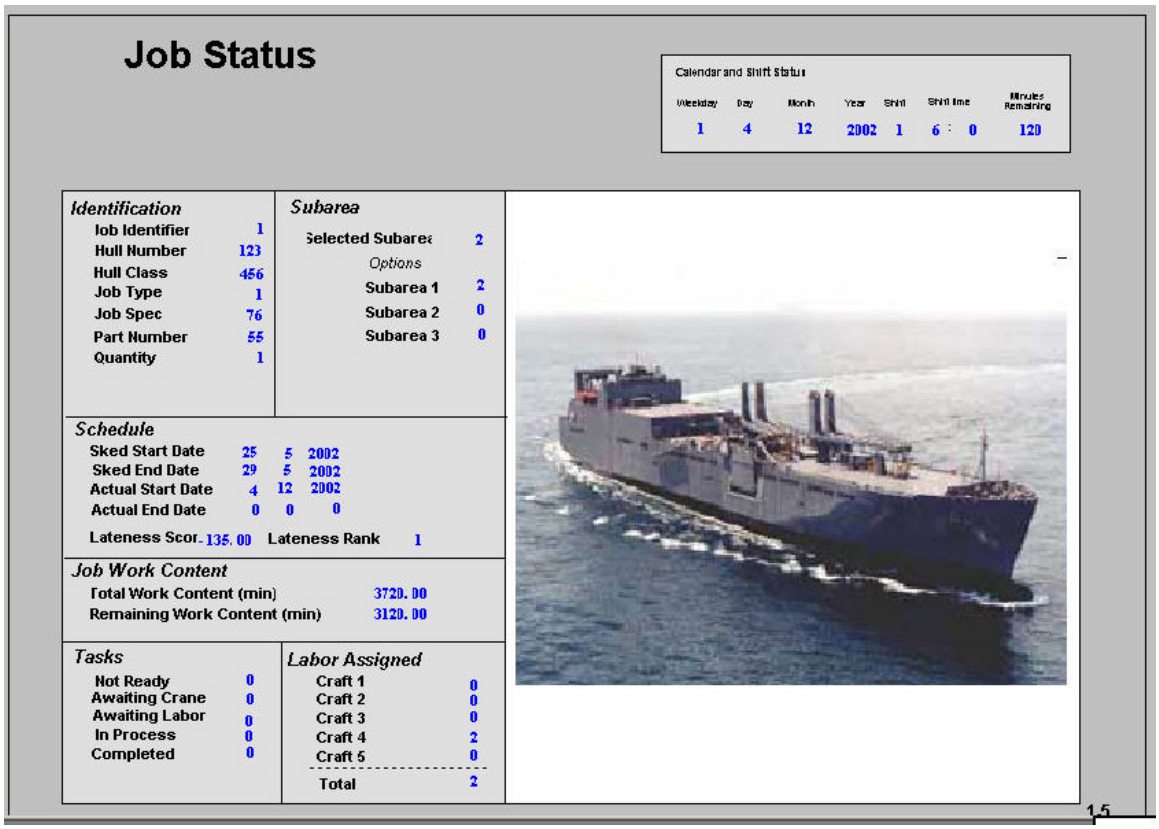


Figure 8: Job Status Screen

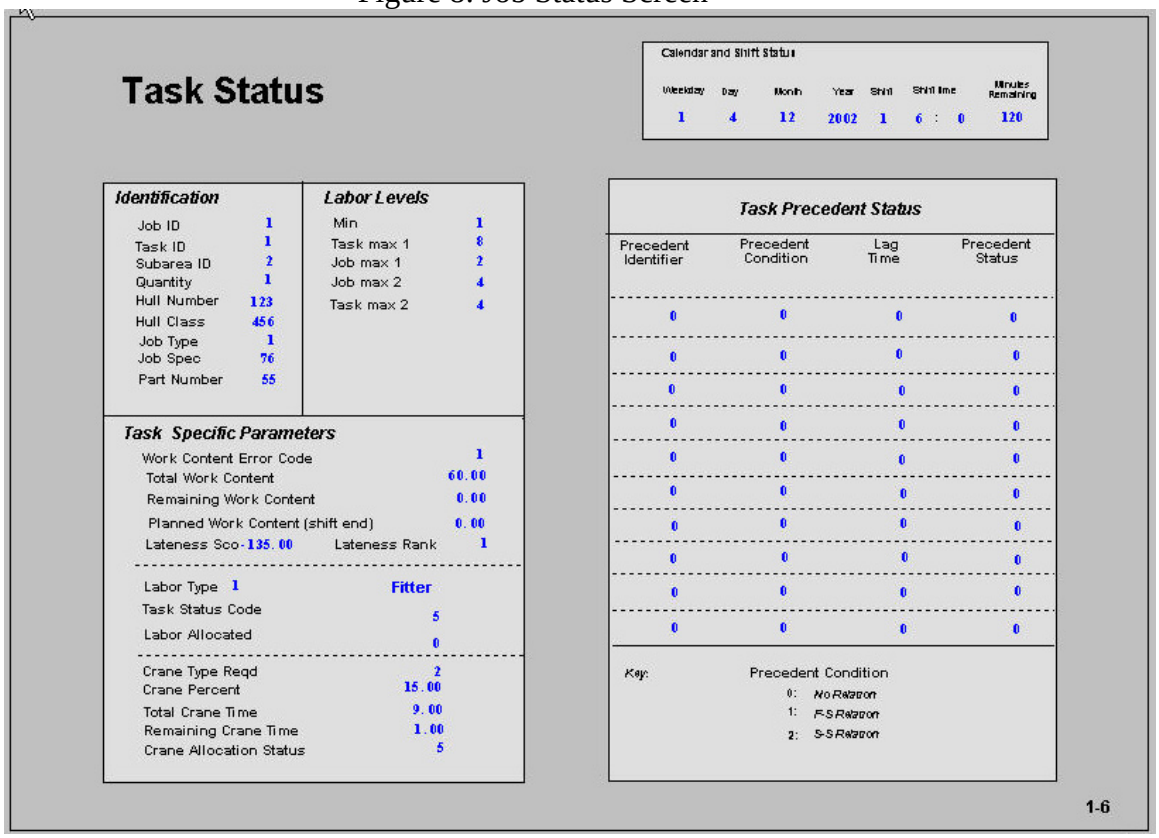


Figure 9: Task Status Screen

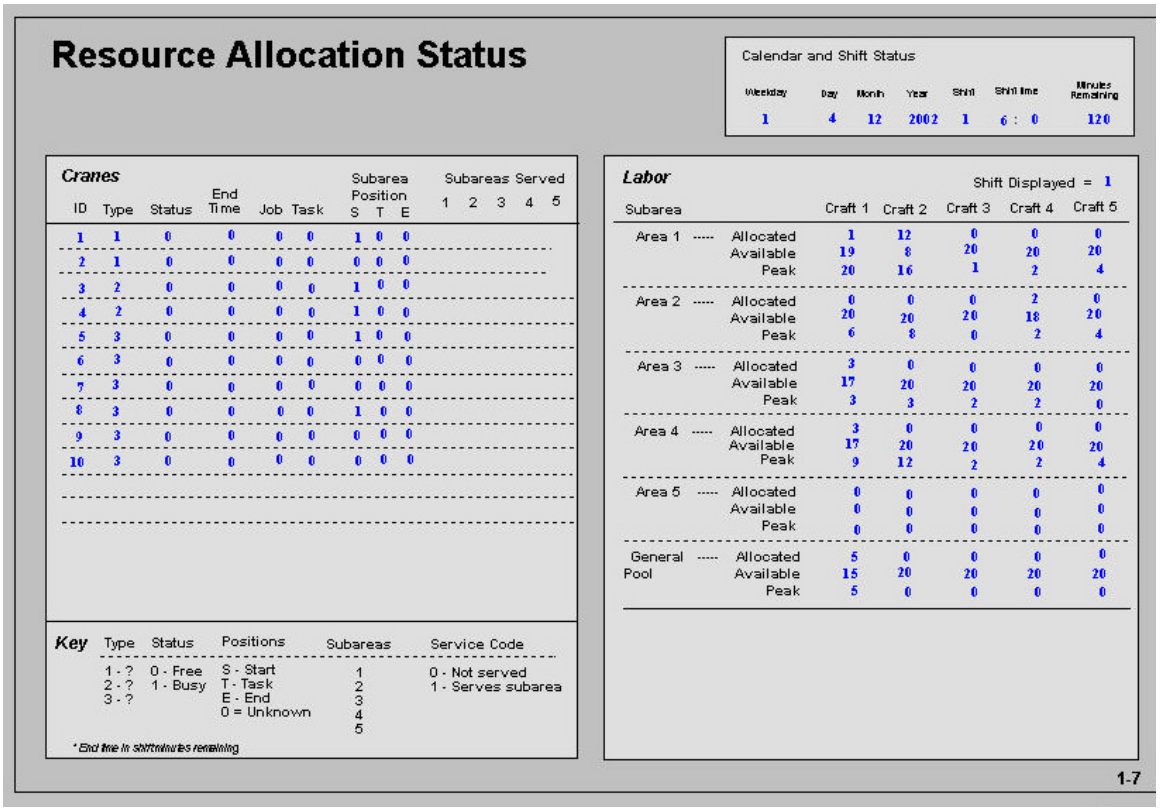


Figure 10: Resource Allocation Status Screen

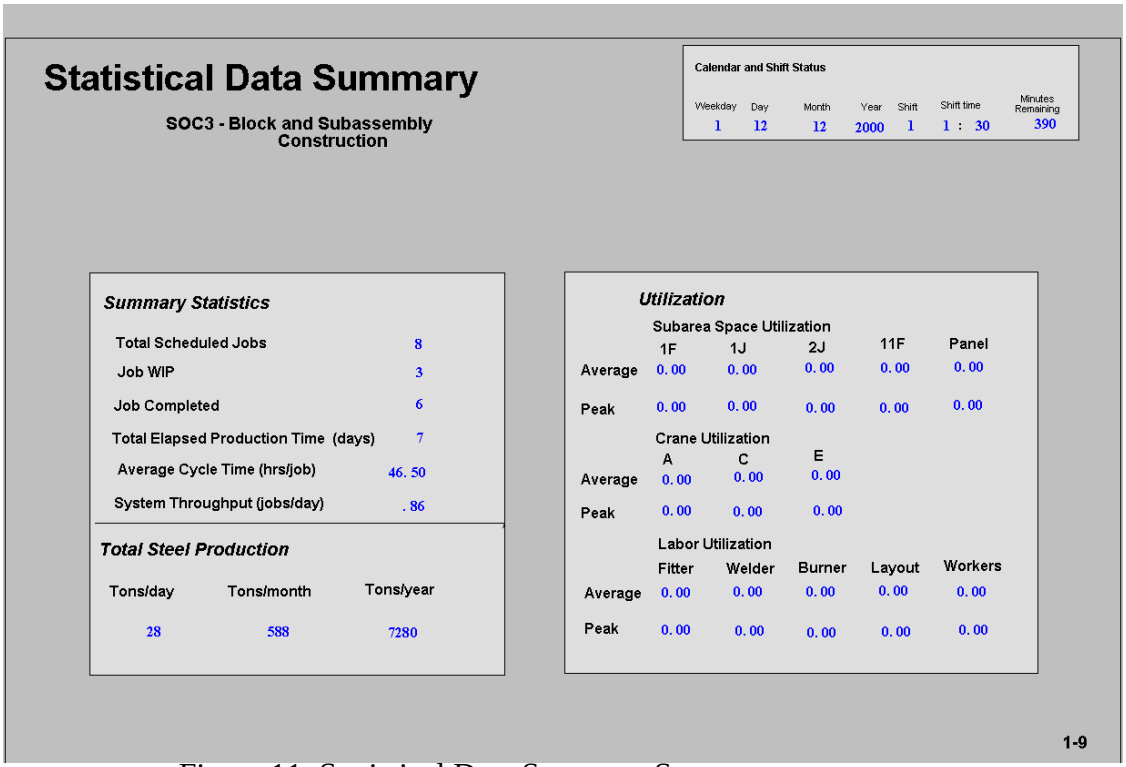


Figure 11: Statistical Data Summary Screen

7 USER INTERFACE MENUS AND DIALOG BOXES

Figure 12 is the menu that allows users to select any available screen to view the status. Figure 13 is the menu that allows users to select any available subarea to view its allocation status for the particular subarea. Figure 14 is the main menu that allows users to select any available submenu. Figure 15 is the form that allows users to enter the job identifier of interest to view the job information. Figure 16 is the form that allows users to enter the task identifier of interest to view the task information. Figure 17 is the form that allows users to enter the shift identifier of interest to view related information. Figure 18 is the feedback message that instructs users how to cycle through all jobs or tasks to view the relevant information. Figure 19 is the error feedback messages that indicate the users have entered an invalid job identifier and provides a correct range for the user to continue. Figure 20 displays an error feedback message that indicates the user has entered an invalid task identifier and provides a correct range for the user to proceed. The messages showed in Figure 19 and Figure 20 provide accurate error correction information to help the users.

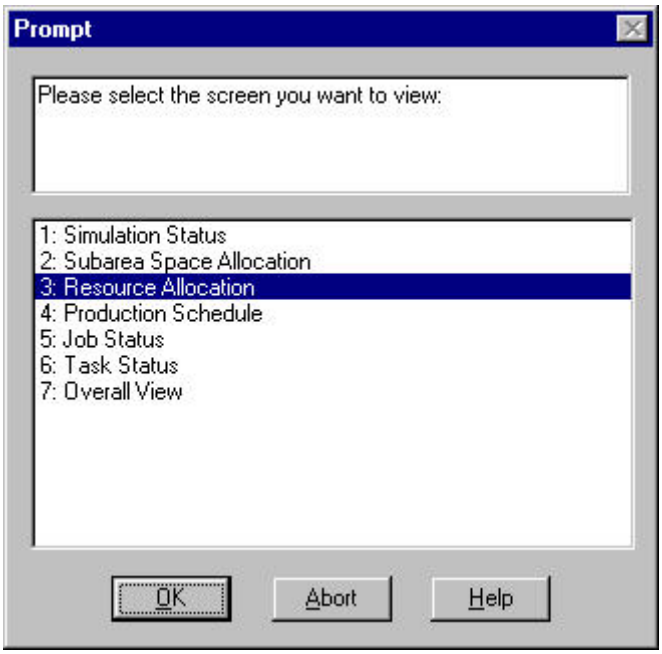


Figure 12: Menu for Screen Selection

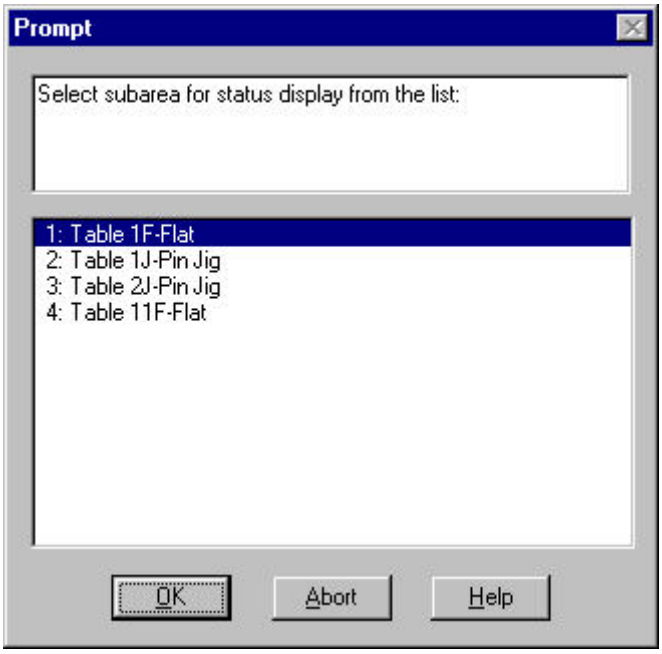


Figure 13: Menu for Subarea Selection

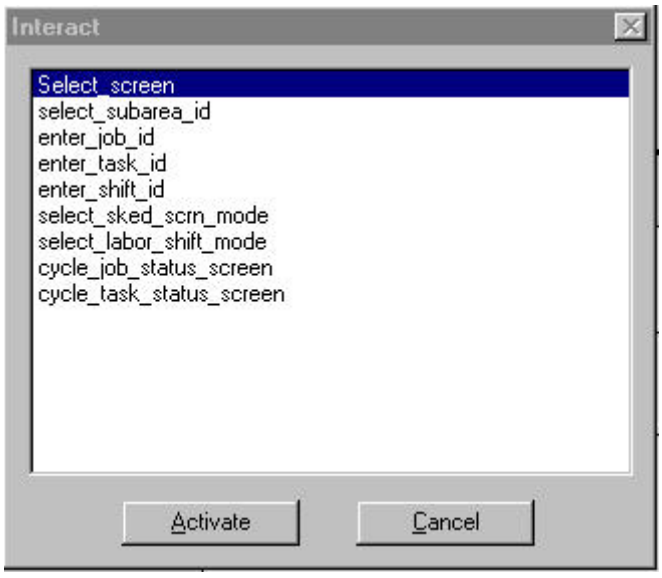


Figure 14: Main Menu

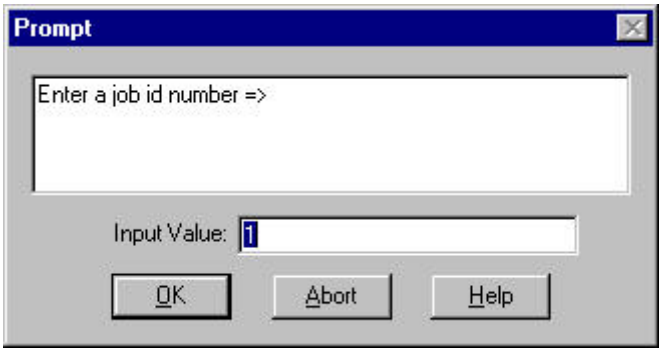


Figure 15: Form for Job Identifier Entry

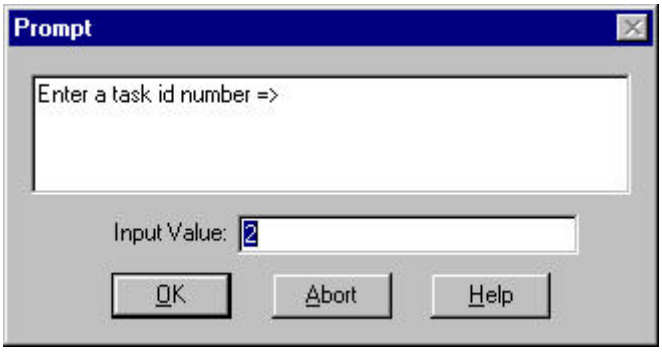


Figure 16: Form for Task Identifier Entry

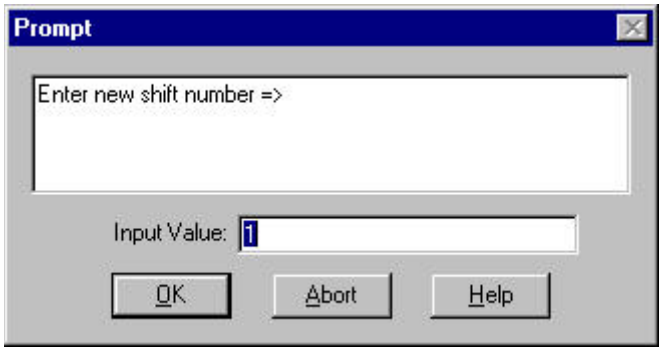


Figure 17: Form for Shift Identifier Entry



Figure 18: Feedback Message for Cycling



Figure 19: Error Message for Wrong Job Identifier Entry

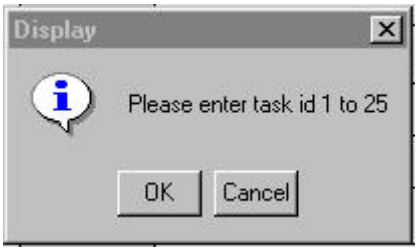


Figure 20: Error Message for Wrong Task Identifier Entry

8 STEPS TO USE THE SIMULATION INTERFACE

The following steps are the guides to user the simulation user interface:

1. Launch ProModel software
2. Open simulation model Shipbuilding_UI.MOD
3. Use F10 or go to the pull down menu "Simulation" (the fifth one from left) → then select Run
4. Click "Interact," a pop-up window will show up.
5. Choose "Select_screen" to activate, the window becomes a prompt window and it asks you to select from one of the following nine screens you wish to view:
    - *Simulation Status*
    - *Subarea Space Allocation*
    - *Location status*
    - *Resource Allocation*
    - *Production Schedule*
    - *Job Status*
    - *Task Status*
    - *Statistical data summary*
    - *Overall View*
6. After you highlight one of the screens and hit OK, that screen will be displayed.

7. If you are viewing the Subarea Space Allocation screen, you can repeat step 4, and choose the *select_subarea_id* to activate; and then select one of the four subareas to view information.
8. If you are viewing the Resource Allocation Status screen, you can repeat step 4, and choose *enter_shift_id* or *select_labor_shift_mode* to activate it to view the information.
9. If you are viewing the Production Schedule screen, you can repeat step 4, and choose *select_sked_scrn_mode* to activate, and then select the sorting method to view the information.
10. If you are viewing the Job Status screen, you can repeat step 4, and then choose from one of the following:
    - Choose *enter_job_id* to activate, and then type in a job identifier to view its information.
    - Choose *cycle_job_status_screen* to activate, then cycle through all the jobs in the project to view the job information
11. If you are viewing the Task Status screen, you can repeat step 4, then choose from one of the following:
    - Choose *enter_job_id* to activate, and then type in a job identifier to view its information.
    - Choose *enter_task_id* to activate, and then type in a task identifier to view its information.
    - Choose *cycle_job_status_screen* to activate, then cycle through all the jobs in the project to view the job information.
    - Choose *cycle_task_status_screen* to activate, then cycle through all the tasks in current job to view the job-task information.
12. You can also repeat step 4, Choose “Select_screen” to activate, and then select the *Overall View* screen to view all nine screens together.
13. Error message: when you enter an invalid job or task identifier, a message will pop up and tell you what the correct number range is.

Before the execution of the simulation model, the user can view each screen by using the “hot key” Ctrl + F function keys, for example for the simulation status screen, use Ctrl + F1, and so on.

CONCLUSIONS

This paper has provided a brief overview of the user interface of the simulation of shipbuilding operations developed as a part of the NSRP Project by staff of the NIST Manufacturing Interoperability Program. The run time animated and graphical display of work location, resources, production flows, and simulation results help shipyards to make better production management and resource-allocation decisions. Using discrete event simulation software to design an effective user interface is a challenge. However, we still

applied the user-centered strategy for the development of the shipbuilding simulation user interface and received positive response from end users.

REFERENCES

Cooper, Alan 2002. About Face, the Essentials of User Interface Design, Hungry Minds, Inc.

Galitz, O. Wilbert 2002. The Essential Guide to User Interface Design, Second Edition, An introduction to GUI Design Principles and Techniques. Wiley Computer Publishing, John Wiley and Sons Inc.

McLean and Shao 2001. Simulation of the Shipbuilding Operations, Proceedings of the 2001 Winter Simulation Conference. New Jersey: Institute of Electrical and Electronics Engineers. Or via ‹http://www.informs-cs.org›

ProModel Manufacturing Simulation Software Reference Guide, 1996, Orem, UT

ProModel Manufacturing Simulation Software User's Guide, 1996, Orem, UT

Shneiderman, Ben 1998 Designing the User Interface, Strategies for Effective Human Computer Interaction, third Edition. Addison Wesley Longman, Inc.

Storch, Hammon, Bunch and Richard C. 1995. Ship Production, second Edition. Cornell Maritime Press, Inc.

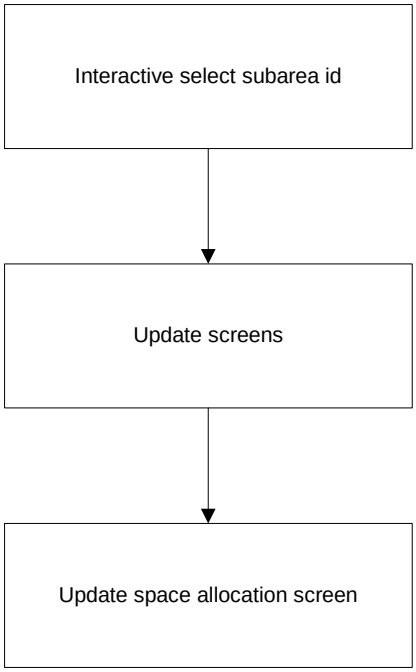
The National Shipbuilding Research Program (NSRP) WebPages <http://www.nsrp.org>

Appendix A

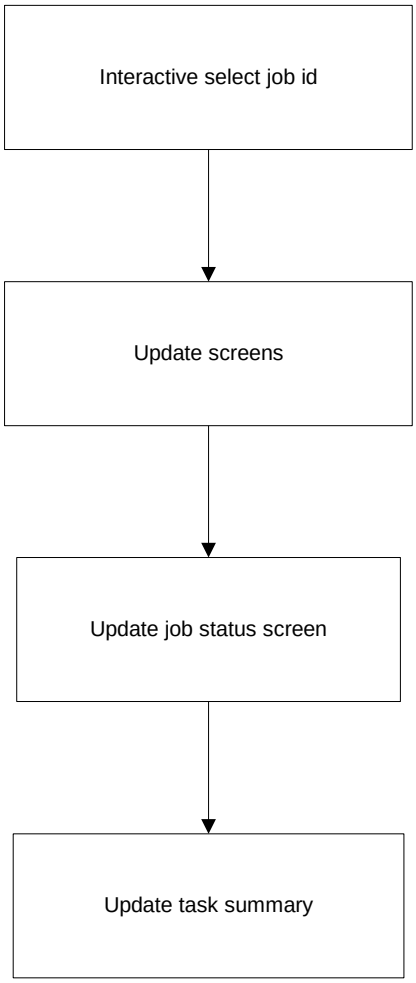
Logical Diagram for the Screens Display



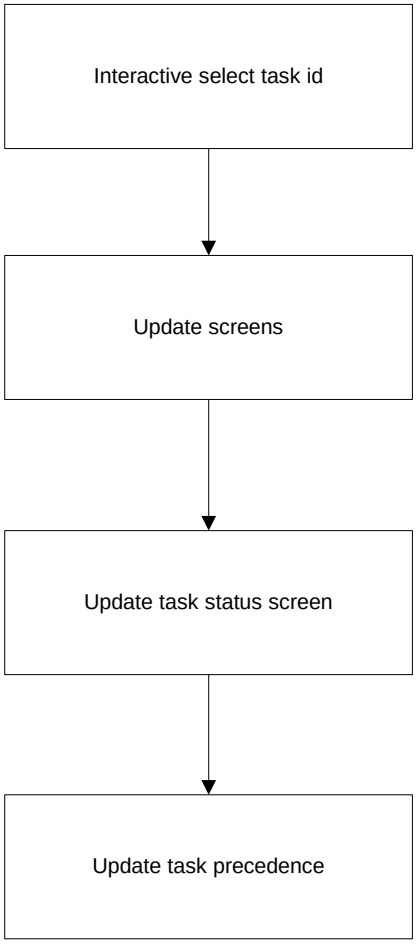
Select subarea id



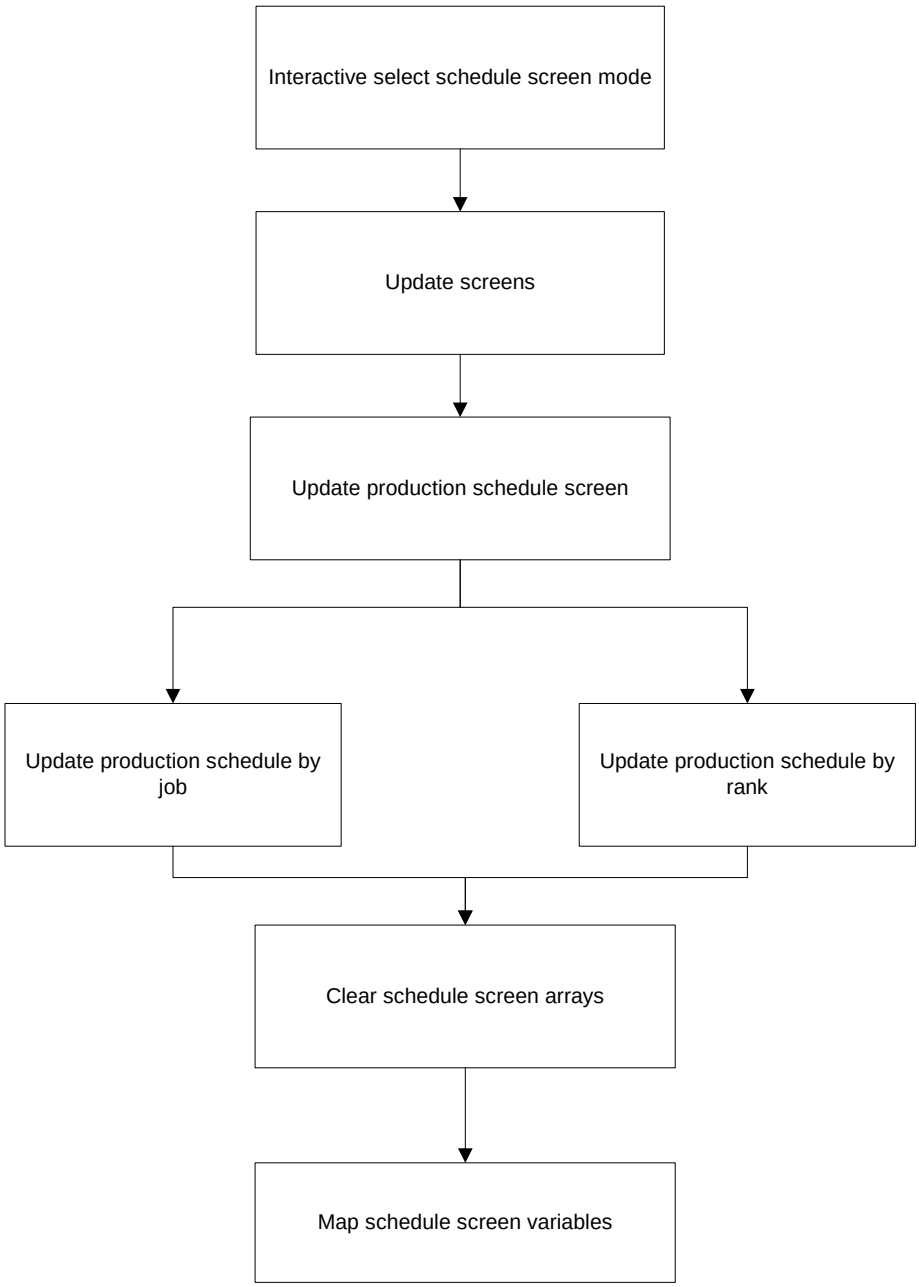
Select job id



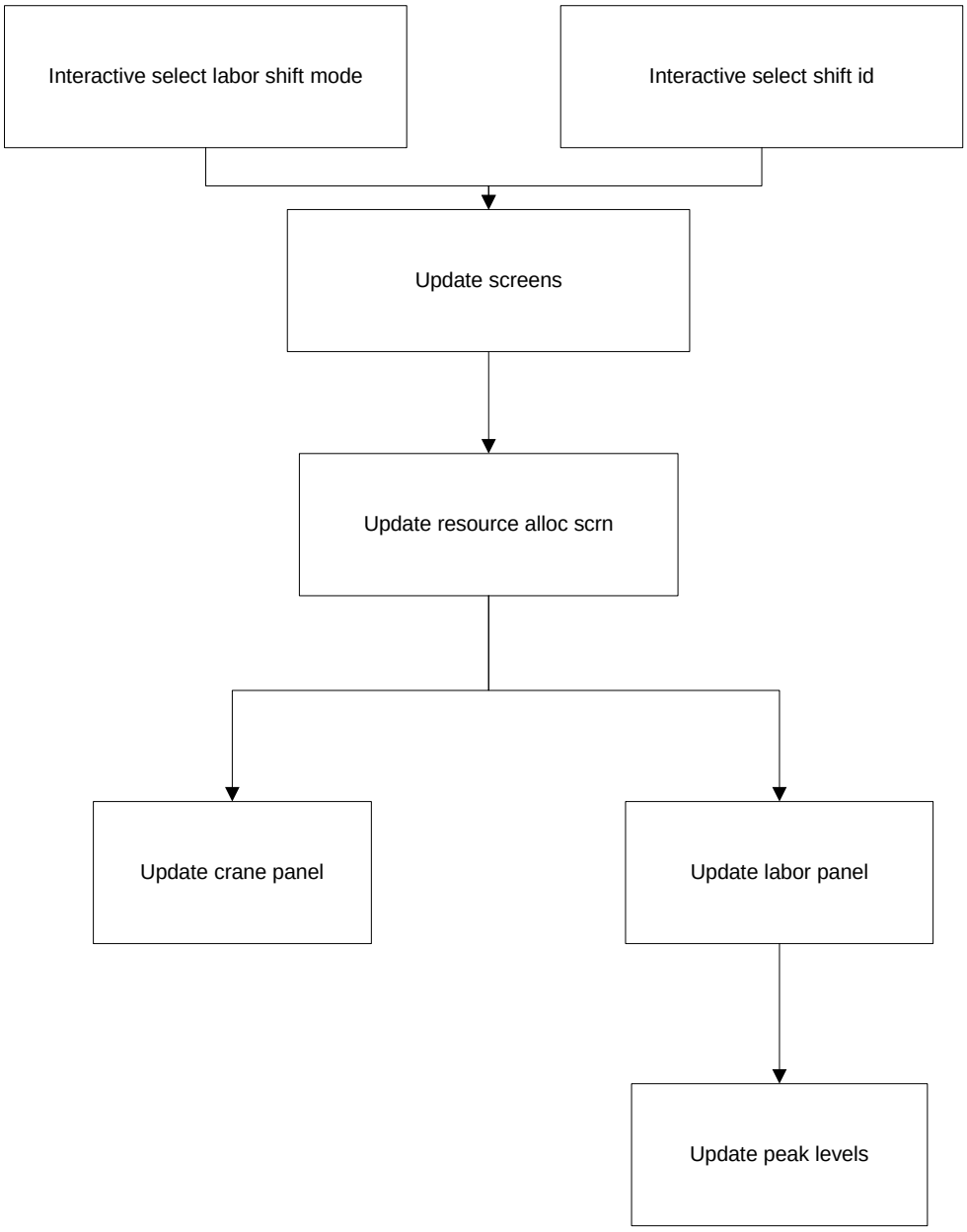
Select task id



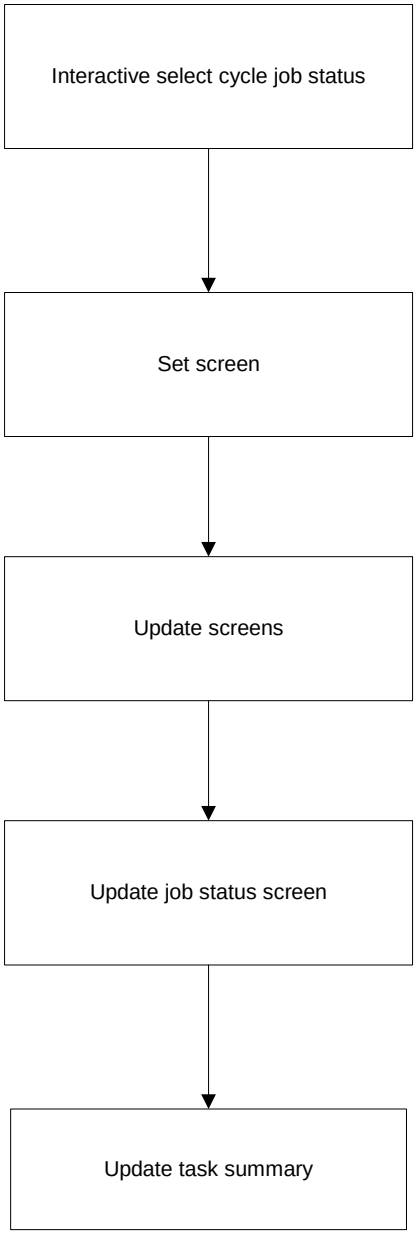
Select sked scrn mode



Select labor shift mode



Select cycle job status



Select cycle task status

