ISE Advanced Technology

Space Station Evolution
Beyond the Baseline

August 6-8, 1991

Barry R. Fox
McDonnell Douglas Space Systems Co.

Scheduling is Hard

- The true nature of a project is not always immediately obvious!

Scheduling is Hard

- Space Station scheduling problems are diverse
  - Assembly Plan
  - Flight Manifest
  - Avionics
    - development
    - test
    - integration
  - Training
    - Shuttle Astronauts
    - Station Astronauts
    - Control Center Operators
    - Integrated Training
  - Facilities
    - Shuttle Simulators
    - Zero-G Simulators
    - Tele-Robotics Simulators
    - Avionics Development Facility

Scheduling is Hard

- Space Station scheduling problems are diverse
  - Station Operations
    - Core processes
    - Payloads
    - Real-time schedule evaluation & revision
  - Ground Operations
    - Control Center staffing
    - Payload Operations
  - Publications
    - Flight Plans
    - Schedules

Scheduling is Hard

- It is difficult to build purely automated scheduling systems
  - It is impossible to reduce our measure of a good schedule to one number that can be used in our search for an optimal schedule:
    - Everyone has different opinions
    - Our priorities depend upon circumstances
  - All algorithms that attempt to find a good or optimal schedule are based upon search:
    - Even familiar algorithms, for instance square root, are based upon search.
    - However, in scheduling problems the time required to solve a given problem increases exponentially with the number of activities and resources to be scheduled.

Scheduling is Hard

- It is difficult to build interactive scheduling systems
  - The implementation of interactive scheduling systems requires a disproportionate investment in the graphical user interface.
    - Estimates now range between 40 and 60 percent of effort.
    - The effort required is likely to increase as user expectations become elevated.
  - User interface standards are rapidly evolving.
    - A standard without concensus is not a standard.
      - Windows 3.0
      - Macintosh
      - X-Windows
      - OpenWindows
      - Motif

Scheduling is Hard

- Familiar, commercial project scheduling software cannot effectively handle the complex "resource constrained" and "state constrained" scheduling problems that are characteristic in the Space Station domain.
- Commercial, off-the-shelf products are not generally available in source code for customer modification.

Scheduling is Hard

- State-of-the-art scheduling research performed at the various NASA centers, including JPL, NASA-ARC, GSFC, and in the research laboratories of the major NASA contractors, including McDonnell Douglas, Martin-Marietta, and Lockhead, is based upon LISP
  - LISP is not generally accepted for production operations.
  - The data structures and algorithms are not easily translated into conventional languages.

Scheduling is Hard

- Extrapolation from the level-of-effort required to support Space Shuttle opeations, indicates that the level-of-effort required to schedule Space Station operations will be extremely challenging.

Background

- January 1989
  - Three Tasks
    - Develop generic planning and scheduling technology (NASA HQ, Code MD)
    - Develop technology that will enable real-time, onboard, schedule evaluation. (NASA HQ, Code MT)
    - Investigate technology that will enable real-time, onboard, schedule revision (MDSSC-SSD, WP-2)
  - Synergistic Combination
    - A full-function scheduler requires both schedule evaluation and revision.
    - Schedule evaluation requires generic scheduling technology
    - Schedule revision requires generic scheduling technology and schedule evaluation technology

Background

- Prior to 1989
  - Prototype, interactive, "job-shop" scheduler
    - Wedge
    - LISP
    - Symbolics, 3600 family of computers
  - McDonnell Douglas Research Laboratories, IRAD

Objective

- Develop and Demonstrate advanced scheduling techniques targeted towards SSF applications
  - Develop a library of scheduling software
    - Written in Ada with an X-Windows user interface
    - That supports both interactive and autonomous scheduling
    - That is generic, yet suitable for the development of specialized applications
  - Demonstrate the capabilities of this software in an interactive scheduling system
  - Develop the peripheral systems that will enable this software to become a stand-alone, turn-key, off-the-shelf product.

Benefits

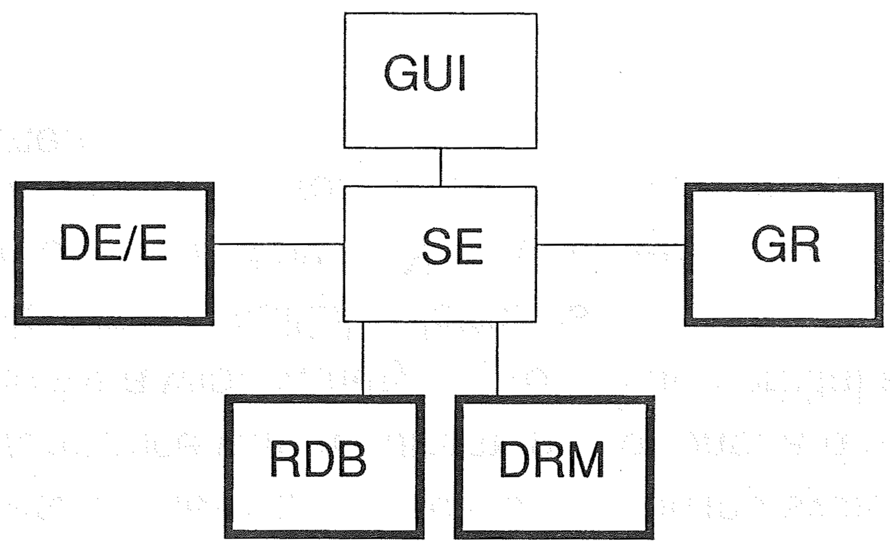
- Increased Productivity
  - By the groups that perform scheduling
- Increased Utilization
  - Of the people, resources, and facilities being scheduled
- Cost Avoidance
  - Avoid the staffing increases that may be necessary when station becomes fully operational.
- Enhanced Safety
  - Scheduler can guarantee that the resulting schedules satisfy all relevant contraints
  - Evaluator can guarantee that constraints remain satisfied at execution time.

Benefits

- NASA owned, state-of-the-art, interactive scheduling system
  - that is in conformance with requirements for onboard systems
  - that is suitable for a wide variety of ground and orbital applications
  - that is portable across multiple platforms
  - that can be freely shared by all NASA centers and contractors
  - that can serve as a cost-effective platform for continuing research and development

Approach

- Architecture
  - Centralized Scheduling Engine
  - Distributed Resource Managers
  - Graphical User Interface
  - Interactive Data Entry, Editting
  - Graphical Reports
  - Relational Database Connections



Approach

- ❑ Scheduling Engine
  - Incremental Scheduling
    - provides the ability to create or edit the schedule by scheduling and unscheduling one item at a time
  - Non-Chronological
    - provides the ability to create or edit the schedule in much the same way that you might draw and erase I-beams on a piece of paper.

Approach

- Graphical User Interface
  - Current User Interface
    - Based upon X-Windows
    - Enables remote execution
      - Macintosh with X-Windows at JPL
      - Scheduling engine in Houston
  - Next Generation User Interface
    - GENESIS
      - GENEric Scheduling Interface System
    - Jointly designed with JPL
    - Based upon X-Windows
    - Will become a seperate product that can be used by a variety of scheduling engines and applications.
    - Based upon the most general concepts of interactive scheduling.

Approach

- ❑ Software Engineering
  - ■ This software is has been designed for distribution and re-use:
    - Modular (packages)
    - Data Abstraction
    - Information Hiding
    - Side-Effect Free
    - Advanced Data-Driven Testing
    - With emphasis on
      - portability
      - maintainability
      - reusability
      - adaptability

Approach

- NASA ownership
  - Periodic contribution to COSMIC
  - Available directly from the Software Technology Branch
- NASA wide collaboration
  - New user interface was designed through a series of video conferences with JPL
  - Standards for data interchange will be developed through a similar effort
  - Work on distributed resource managers continues with the collaboration of the the COOPES development team
  - Planning is nearly complete for the second Planning and Scheduling workshop to be held September 24-26, whose purpose is to chart a program of cooperative research and development

Baseline Integration

- Actively working with several customers
  - COMPASS is being used to build schedules and resource profiles for the Design Reference Missions.
  - COMPASS has been selected as the basis for ADF and MRMDF scheduling.
  - COMPASS is being evaluated for use in several other application areas including:
    - Space Station Training Office
    - Systems Engineering Simulator
    - Ground Operations Support

Growth and Evolution

- Current Capability
  - provides reasonable, but approximate models of activities and resources
- Development of Advanced Technology is needed to realize the the full potential of the Space Station and supporting groups
  - Interruptable Activities with Persistent Resource Requirements
  - State Constrained Scheduling
    - Methods for accomodating change-over-costs (required for the ADF)
    - Methods for maintaining state-transition time-lines and for scheduling against required conditions.
  - Schedule Optimization
    - Genetic Algorithms
  - Parallel Scheduling Algorithms
    - Pipelined Networks of Transputers

673

Growth and Evolution

- Advanced Applications
  - Distributed Resource Management
    - Enables the creation of high fidelity resource models
    - Enables access to resources managed at remote sites
    - Research performed cooperatively with the COOPES development team.

Growth and Evolution

- Advanced Applications
  - Distributed Scheduling
    - Geographically Distributed
      - Simultaneous access to central scheduler
      - Simultaneous operation of multiple schedulers with centralized data
      - Simultaneous operation of multiple schedulers with multiple sources of data

Growth and Evolution

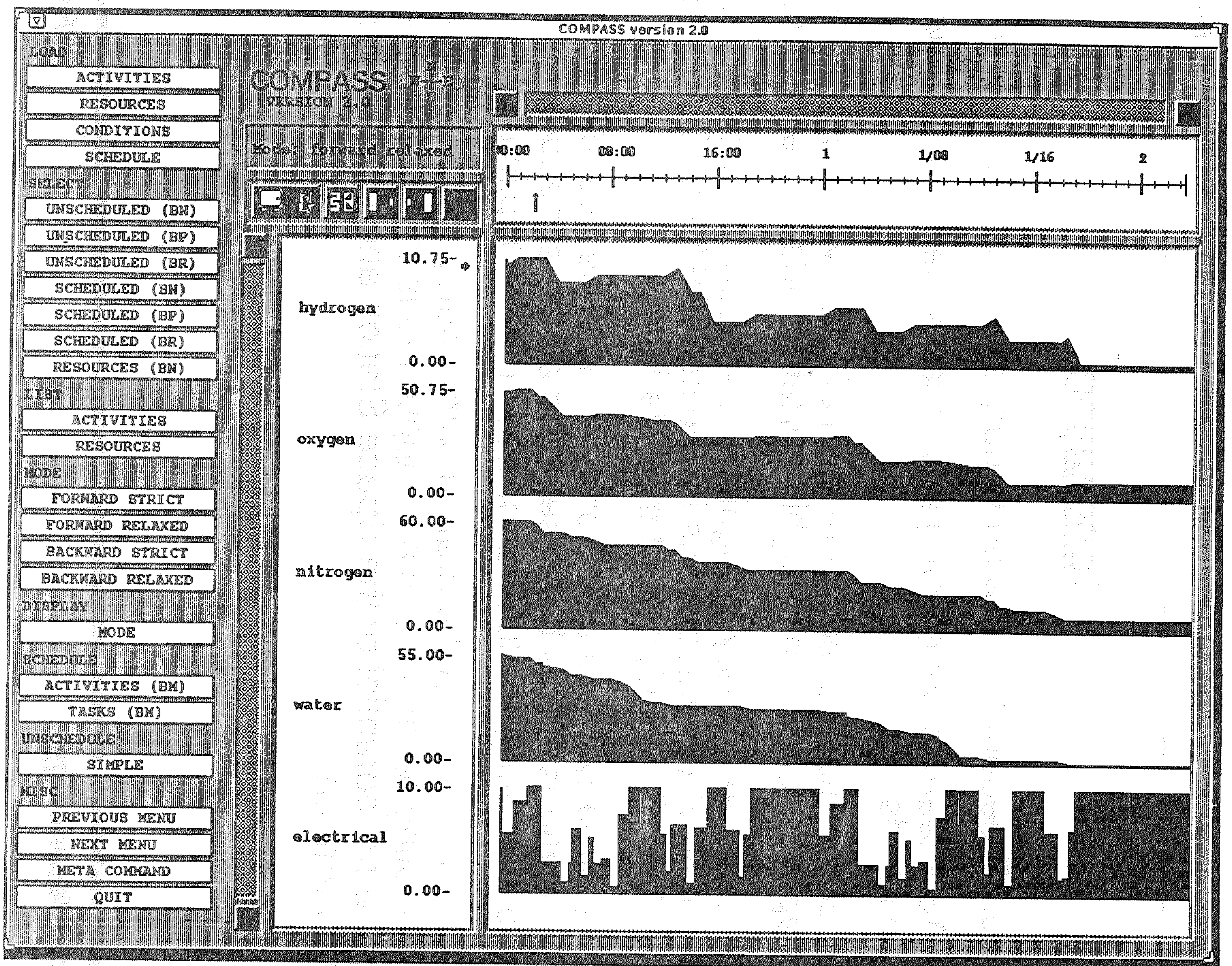
- Advanced Applications
  - Time-phased scheduling
    - Seamless integration of several phases of scheduling
      - Manifiest
      - Long-Term
      - Short-Term
      - Onboard/Detail

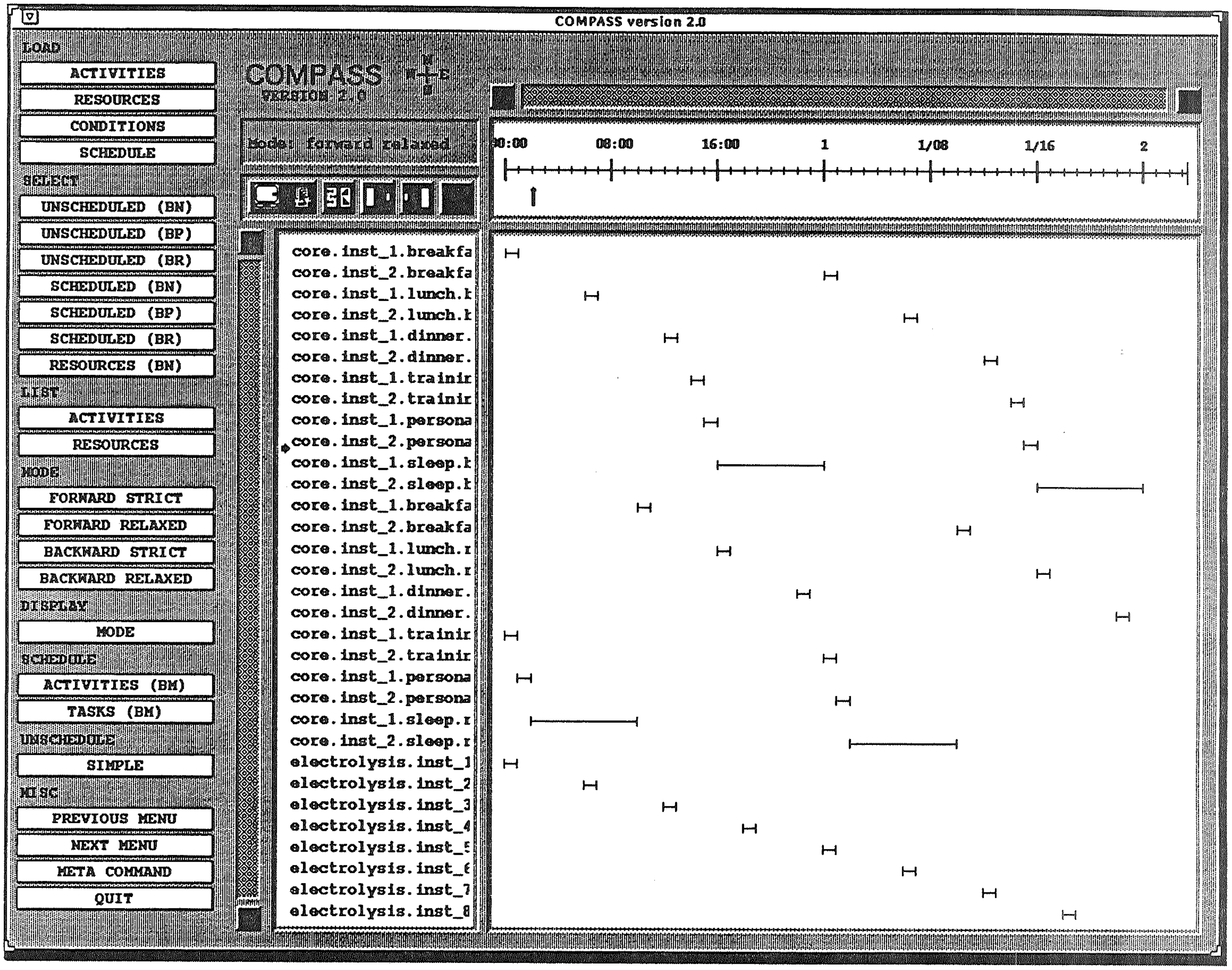
Growth and Evolution

- Advanced Applications
  - Real-Time
    - Evaluation
    - Propagation of delays
    - Interactive and Autonomous Revision

Conclusion

- Space Station Freedom has a wide variety of scheduling needs.
- The COMPASS team is developing advanced scheduling technology to satisfy these needs.
- This technology is maintained in the form of a software library and interactive scheduling application which is highly portable, adaptable, and re-useable.
- COMPASS is the beneficiary of significant support and collaboration from many different NASA organizations including NASA-HQ CODE MT, MD, and R, JPL, NASA-ARC, LeRC, and Martin-Marietta.
- COMPASS is already being used for analysis in several different Space Station organizations, it has been selected as the scheduler for two critical facilities, and it is being evaluated for use in other applications.
- The COMPASS team, in collaboration with others, continues to develop specific scheduling technologies and applications that are necessary in order to achieve the full potential of the Space Station and the organizations that support its operation.


COMPASS version 2.0
LOAD
ACTIVITIES
RESOURCES
CONDITIONS
SCHEDULE
SELECT
UNSCHEDULED (BN)
UNSCHEDULED (BP)
UNSCHEDULED (BR)
SCHEDULED (BN)
SCHEDULED (BP)
SCHEDULED (BR)
RESOURCES (BN)
LIST
ACTIVITIES
RESOURCES
MODE
FORWARD STRICT
FORWARD RELAXED
BACKWARD STRICT
BACKWARD RELAXED
DISPLAY
MODE
SCHEDULE
ACTIVITIES (BM)
TASKS (BM)
UNSCHEDULE
SIMPLE
MISC
PREVIOUS MENU
NEXT MENU
META COMMAND
QUIT
COMPASS
VERSION 2.0
Mode: forward relaxed
08:00
16:00
1
1/08
1/16
2
10.75-
hydrogen
0.00-
50.75-
oxygen
0.00-
60.00-
nitrogen
0.00-
55.00-
water
0.00-
10.00-
electrical
0.00-


COMPASS version 2.0
LOAD
ACTIVITIES
RESOURCES
CONDITIONS
SCHEDULE
SELECT
UNSCHEDULED (BN)
UNSCHEDULED (BP)
UNSCHEDULED (BR)
SCHEDULED (BN)
SCHEDULED (BP)
SCHEDULED (BR)
RESOURCES (BN)
LIST
ACTIVITIES
RESOURCES
MODE
FORWARD STRICT
FORWARD RELAXED
BACKWARD STRICT
BACKWARD RELAXED
DISPLAY
MODE
SCHEDULE
ACTIVITIES (BM)
TASKS (BM)
UNSCHEDULE
SIMPLE
MISC
PREVIOUS MENU
NEXT MENU
META COMMAND
QUIT
COMPASS
VERSION 2.0
Mode: forward relaxed
08:00
16:00
1
1/08
1/16
2
core.inst_1.breakfa
core.inst_2.breakfa
core.inst_1.lunch.
core.inst_2.lunch.
core.inst_1.dinner.
core.inst_2.dinner.
core.inst_1.trainir
core.inst_2.trainir
core.inst_1.persona
core.inst_2.persona
core.inst_1.sleep.
core.inst_2.sleep.
core.inst_1.breakfa
core.inst_2.breakfa
core.inst_1.lunch.
core.inst_2.lunch.
core.inst_1.dinner.
core.inst_2.dinner.
core.inst_1.trainir
core.inst_2.trainir
core.inst_1.persona
core.inst_2.persona
core.inst_1.sleep.
core.inst_2.sleep.
electrolysis.inst_1
electrolysis.inst_2
electrolysis.inst_3
electrolysis.inst_4
electrolysis.inst_5
electrolysis.inst_6
electrolysis.inst_7
electrolysis.inst_8