VIS 2003

Visualizing a Mars Rover with RSVP

Scott Maxwell

Mars Exploration Rover (MER) Project

Jet Propulsion Laboratory

http://mars.jpl.nasa.gov/mer/

The Mars Exploration Rover Project

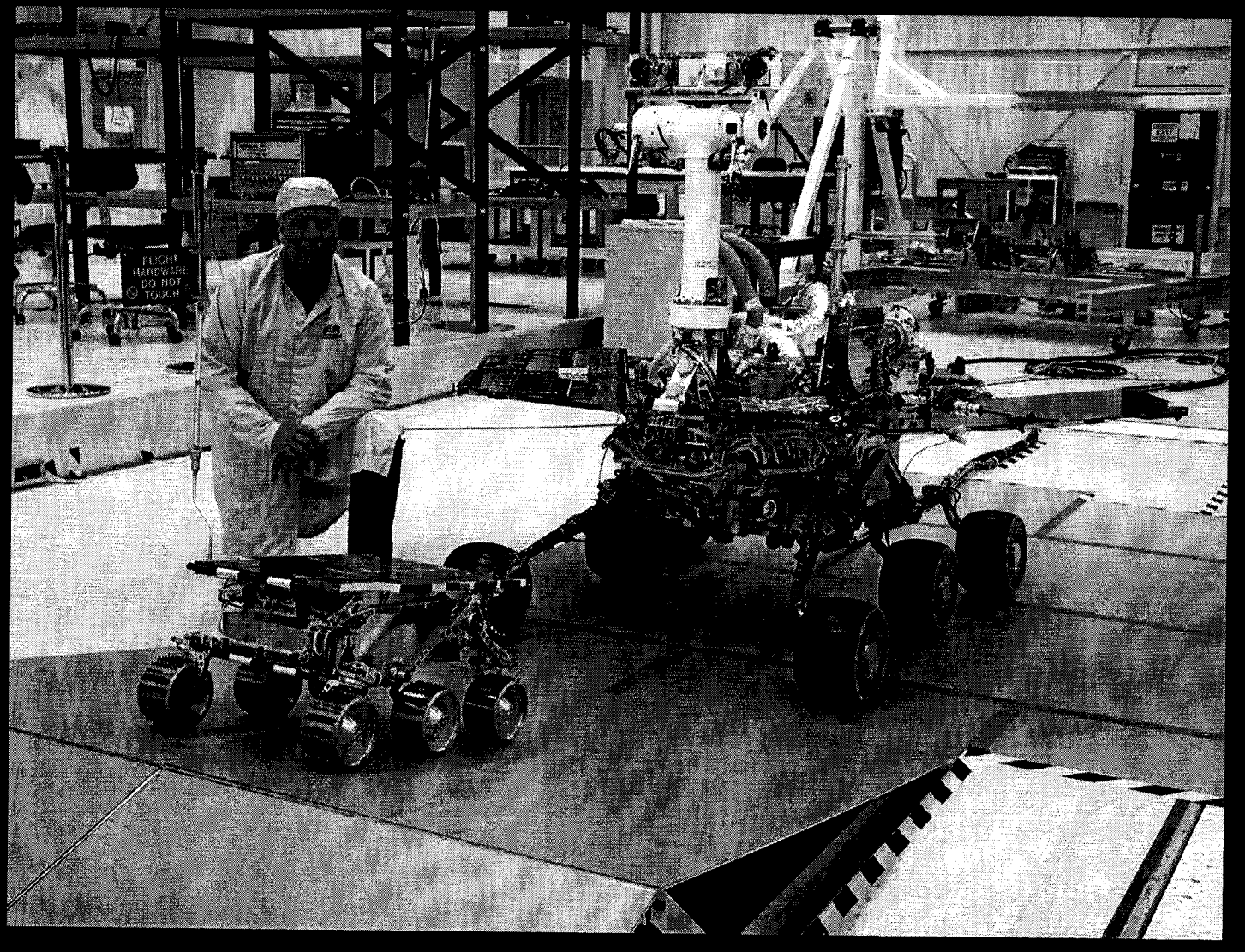
- Twin Mars rovers
- Currently en route to Mars
  - Land January 4 and January 25, 2004
  - Operate for 90 sols (90 Martian days, ~ 92 Earth days) each
  - Drive up to 1 km each

Mars Roving, Then and Now

- Mars Pathfinder (MPF)
  - One Rover (Sojourner)
  - Microwave oven
  - One site
    - Separate lander & rover
    - Rover on wireless "leash"
  - No arm
  - ~100 commands/day
  - 1997 technology

- Mars Exploration Rovers (MER)
  - Two rovers (Spirit and Opportunity)
  - Golf cart
  - Multiple sites
    - "The lander *is* the rover"
    - Rovers unleashed!
  - Arm ("Instrument Deployment Device") — like elephant's trunk
  - 1000+ commands/day/rover
  - 2003 technology

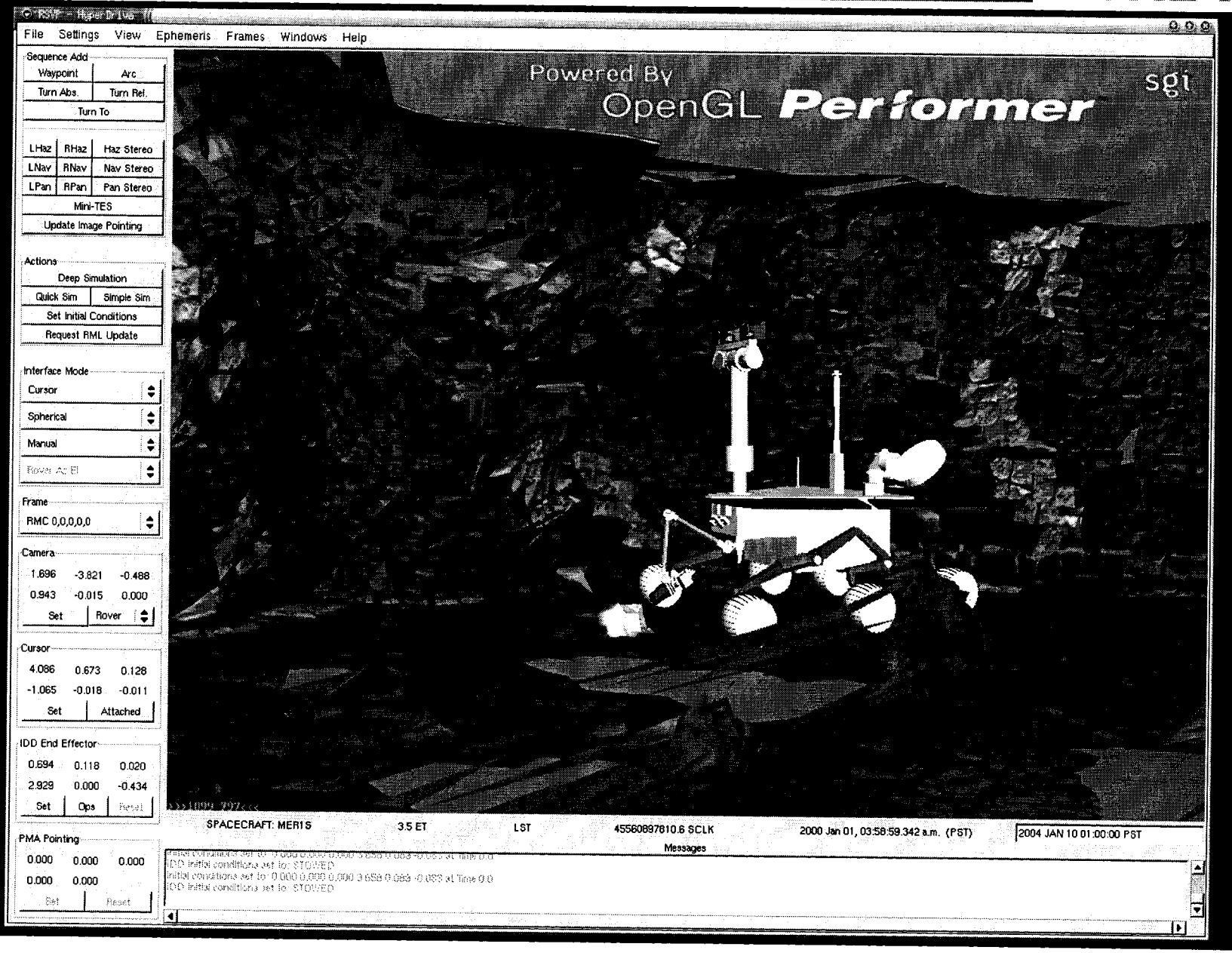
Two Generations of Rovers



RSVP Overview

- "RSVP" = Rover Sequencing and Visualization Program
  - Successor to Rover Control Workstation software used on Pathfinder
- Command-level editor for MER
  - Light-time delays force plan-and-wait model — send the rover its daily "to-do" list, visualize & simulate, then send & see what it did
- Two major components:
  - RoSE (Rover Sequence Editor) is a text-based editor for all rover commands
  - HyperDrive is the part they'll show on the news — cool 3-D visual editing for a subset of rover commands (mobility, IDD, & imaging)
    - Part of good visualization is knowing what to leave out ….
    - This presentation focuses almost exclusively on HyperDrive
  - RoSE and HyperDrive are linked by a message-passing executive that helps synchronize their "views" of the command load

Obligatory Screenshot



We Need Distributed Collaboration — But Can We Afford It?

VIS 2003

- On Pathfinder, all rover commanding routed through one person
  - Science, engineering, mobility
  - Had one $200K SGI workstation (+ 1 backup) generously donated by SGI
- Won't scale up to MER (> 10x the work in the same time)
- On MER, each instrument team needs to build its own sequences, as do engineering and mobility teams — for both rovers simultaneously
  - No $ for $200K SGI workstation for everyone (and even SGI isn't *that* generous ….)

Distributed Collaboration: Our Solutions

VIS 2003

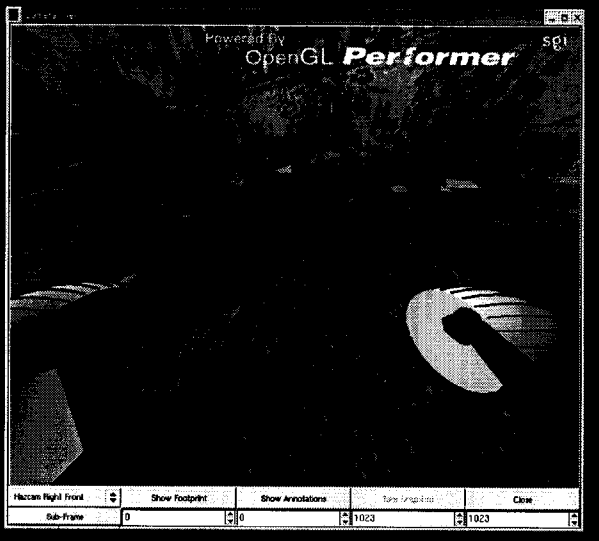
- GNU/Linux to the rescue!
  - "Cheap" ($5K-$7K), fast PCs — can buy enough for everyone
    - 2003 PC has more muscle than 1997 SGI (CPU & graphics)
    - 2x2.8GHz Xeons, 2-4GB RAM, NVIDIA Quadro FX 2000
  - Current capabilities > 4x what we had when we started in late 1999
  - Thank you, Moore's Law!
- Rover drivers get 2 new SGIs
  - Infinite Reality 4, dual-pipe for 60Hz/eye stereo, 4x650MHz MIPS R14000 CPUs, 4GB RAM (1GB texture RAM), $150K each
- Multiple-platform support issues straightforward to resolve:
  - Non-performance-critical code (RoSE) in "naturally portable" Java
  - Performance-critical code, especially visualization code, in portable C++
    - Base code on OpenGL Performer
    - Mostly developed on GNU/Linux, regularly built and tested on SGIs
  - Represent command load files using platform-neutral XML
- RoSE integrates XML command files produced by multiple teams

Simulation

- Light-time delay forces plan-and-wait commanding; can't "joystick"
- Fully simulating rover after every change is too slow
  - Three levels of simulation (trading speed for fidelity):
    - *Simple simulation* (very fast, very low fidelity)
      - Assumes perfect straight-line traversal between points
    - *Quick simulation* (reasonably fast, medium fidelity)
      - Simulated rover conforms to terrain, more realistic durations
    - *Deep simulation* (slow, high fidelity)
      - Runs actual rover flight software within virtual environment
- Simulation results (all levels) can be turned into a movie on the fly
  - Step through movie to see rover's predicted behavior
  - Computed kinematic state saved as XML file
  - Display movie in daily command reviews as part of uplink process
  - Valuable outreach products
  - Same interface can be used to play back telemetry (see what the rover did yesterday)

Getting the Telemetry You Want with the Rover's-Eye View

- Rover has 9 "eyes" (digital cameras)
  - 8 of the 9 are arranged as 4 stereo pairs for 3-D imaging (9th is a science instrument, the Micro Imager)
- Often want a specific image (of an interesting rock, of IDD workspace, etc.)
- HyperDrive can show the view from any camera at any point in the virtual environment
- Modeling is highly accurate and takes individual camera models into account
  - All cameras are the same resolution (1024x1024) but have different fields of view suited to their different purposes
  - Images on right are from a recent test — top image is predicted hazcam image, bottom is actual



High-Performance Visualization

- Spacecraft telemetry -> MIPL (JPL's Multimission Image Processing Laboratory) -> 3-D terrain mesh (.pfb file)
- Many challenges:
  - Rover model conforms to terrain as it moves
  - Rover body, suspension, mast, IDD all articulate accurately
  - Detect collision with terrain
  - Detect IDD collision with rover body
  - Detect unreachable IDD positions
  - May be in a new site each day
  - …

Lagniappe

- Different cursors to engage kinesthetic sense of terrain
  - 6-foot-tall “fashion-plate man” cursor shown to right

Acknowledgments

- JPL / NASA / The American Taxpayer!
- RSVP Software Team:
  - Brian Cooper (team lead)
  - Frank Hartman
  - Scott Maxwell
  - John Wright
  - Jeng Yen
- Additional software development:
  - Carlos Balacuit
  - Chris Leger
- And assistance from:
  - SGI
  - Richard Stallman, Linus Torvalds, Red Hat, the usual cast of thousands ....