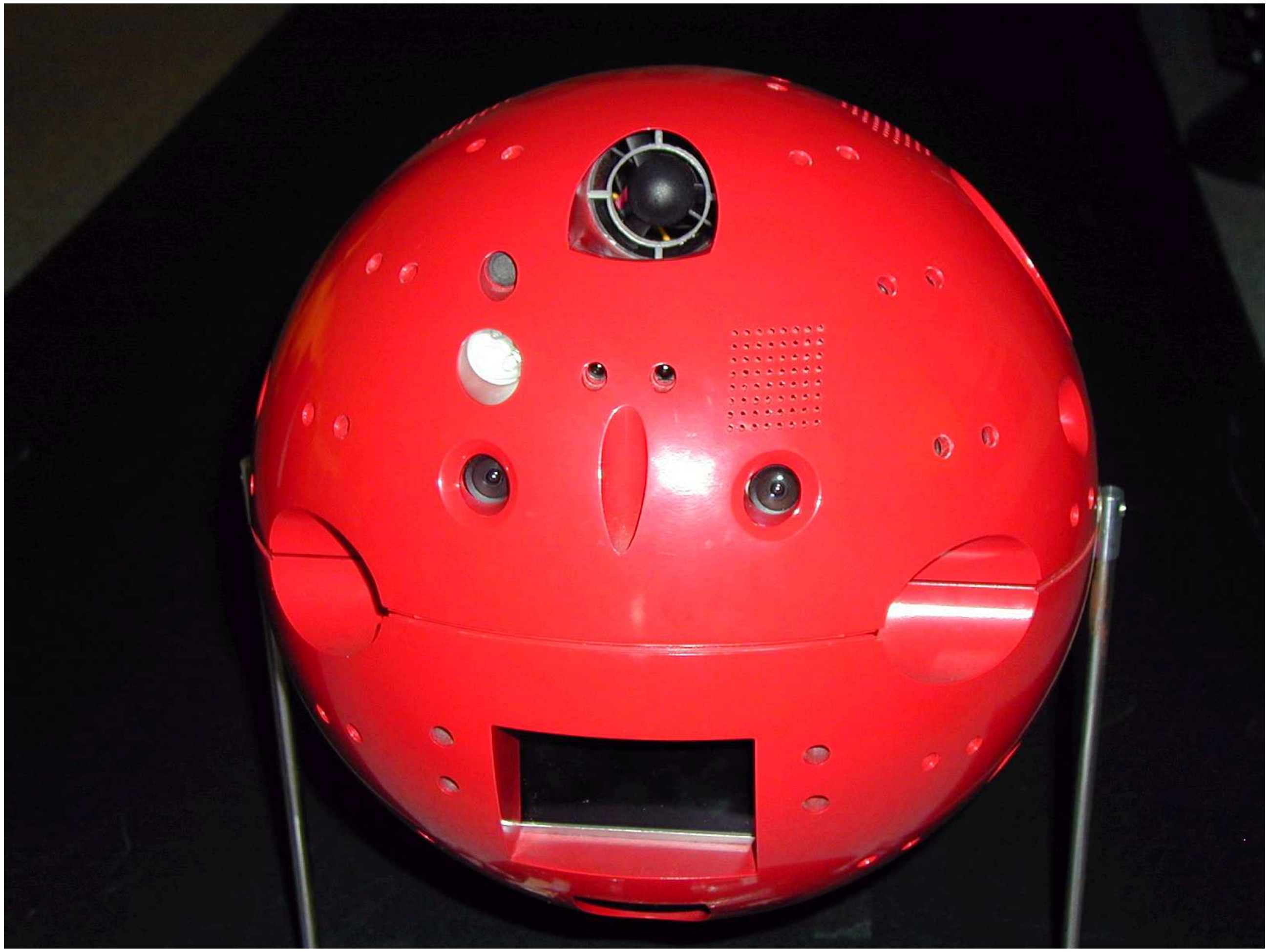
Automated Reasoning FY02 Workshop: Spacecraft Mobile Robot Project



Task Lead: Gregory A. Dorais

Motivation

- ►The demand to perform space activities exceeds our ability to safely and economically deploy the people to accomplish the tasks.
- ►The capabilities of mobile robots continue to increase such that they could significantly increase mission success by regularly performing scientific, diagnostic, monitoring, maintenance, and repair tasks inside and outside spacecraft, both manned and unmanned.
- ►Spacecraft mobile robot testbeds are useful for developing a wide range of intelligent system technologies including multi-agent planning/scheduling, adaptive control, intelligent state estimation and diagnosis, and human-robot interaction.

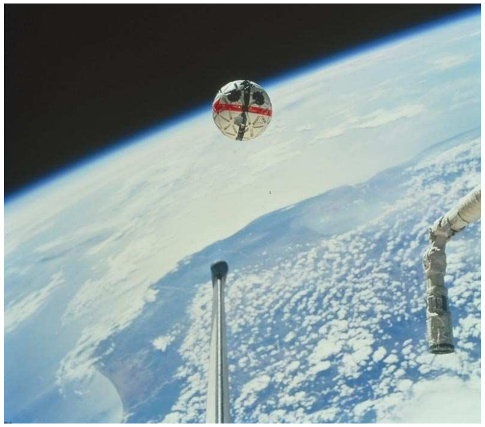
Two Classes of Space Robots

Ames Research Center

FOCUS OF THIS PROJECT

- Spacecraft Mobile Robots (SMRs) for operating in engineered environments for:
  - Sensing/Diagnostics
  - Science/Production
  - Maintenance
  - Construction
  - Crew Support

Example: Sprint AERCam (1997)



- Mobile Robots for operating in natural environments for exploration and material transport:
  - Rovers
  - Atmospheric Flyers
  - Subsurface Drillers and Submersibles

Example: Sojourner (1997)

Challenges of Spacecraft Mobile Robots

- Operate in close range in complex, dynamic, structured environment in 3 dimensions.
- Recognize, and in some cases manipulate, many engineered objects
- Observe nominal and diagnoses off-nominal situations
- Interact with people in a number of ways:
  - People are commanders (at various levels from high to low)
  - People are agents instructed by robot to achieve goal
  - People are dynamic obstacles to avoid
  - People are dynamic objects to track
  - People are peers to collaborate on achieving joint goals

Project Goals

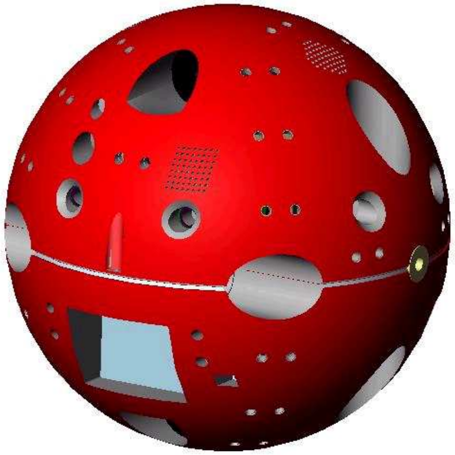
Ames Research Center

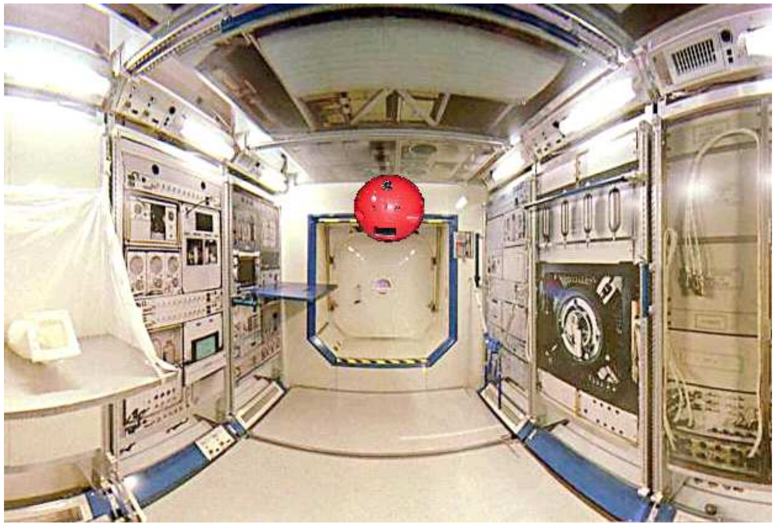
- Develop low-TRL framework for using spacecraft mobile robots to autonomously perform failure assessment, operation documentation, maintenance, and fault recovery on remote spacecraft

- Demonstrate Integrated Vehicle Health Management (IVHM), including autonomous fault diagnosis and cooperative diagnosis & repair, and autonomous space operations to improve crew safety and productivity in the International Space Station *via* an autonomous mobile robot with environment sensing and video-communication capabilities

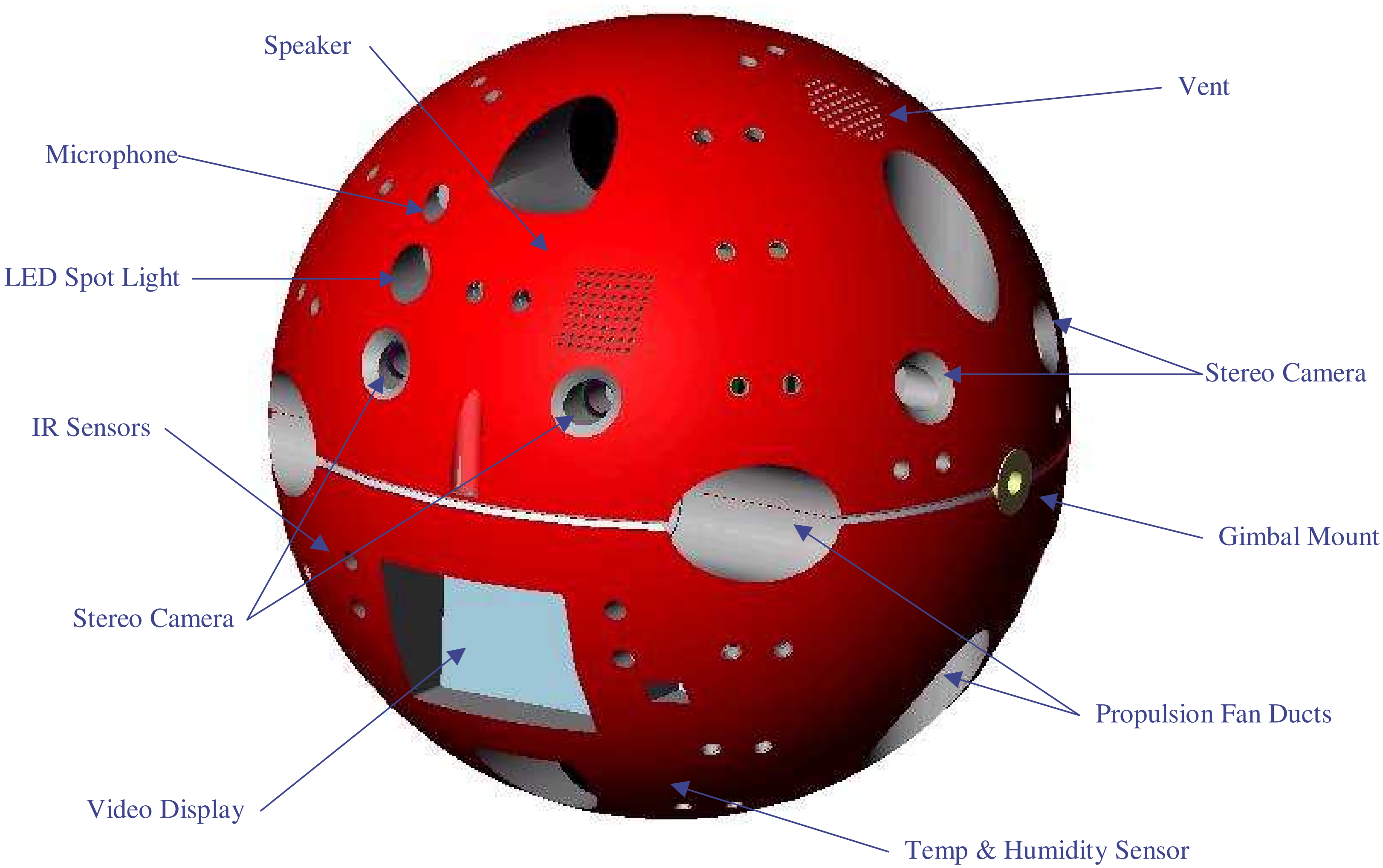
Task Objectives

- Develop a Spacecraft Mobile Robot (SMR): self-propelled robot with cameras, proximity and atmospheric sensors for navigation, monitoring its environment, and video conferencing– *(done in cooperation with ECS program)*
- Develop an adjustably autonomous control software supporting mixed-initiative interaction at plan and execution time and validate on SMR
- Demonstrate autonomous fault diagnosis in internal spacecraft-like environment (physical ISS mockup and simulator) using a SMR in conjunction with an in-situ control system (e.g., Life Support Simulator).





PSA Model 2 Prototype

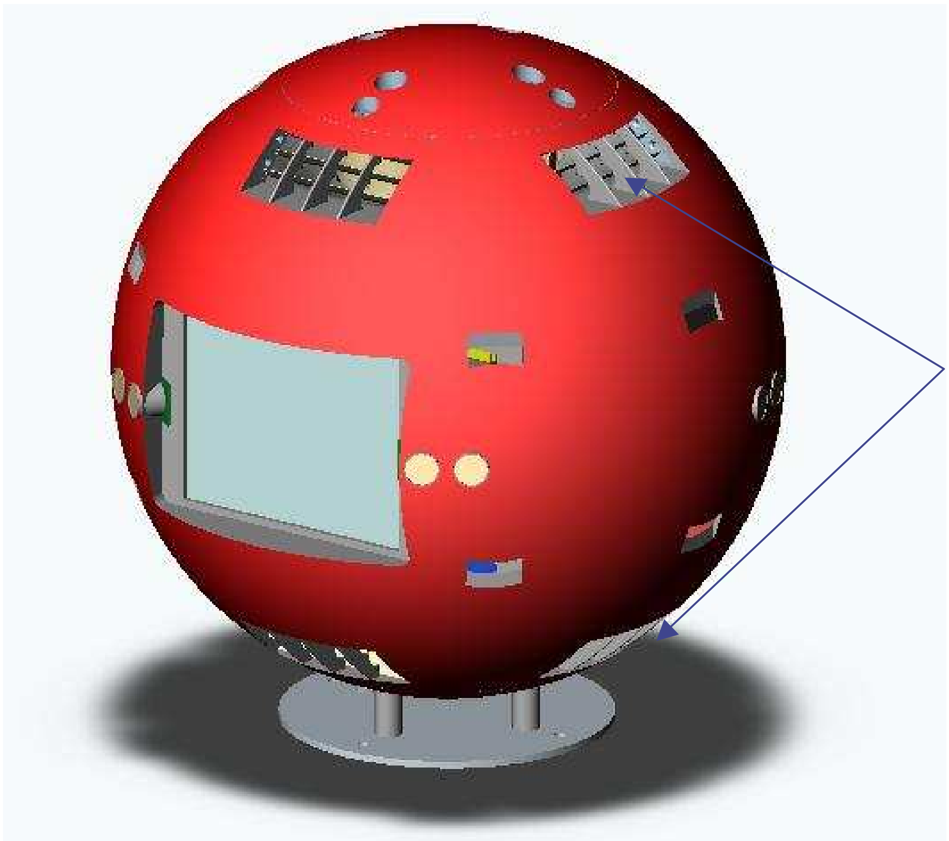


Next-generation Prototype

Ames Research Center

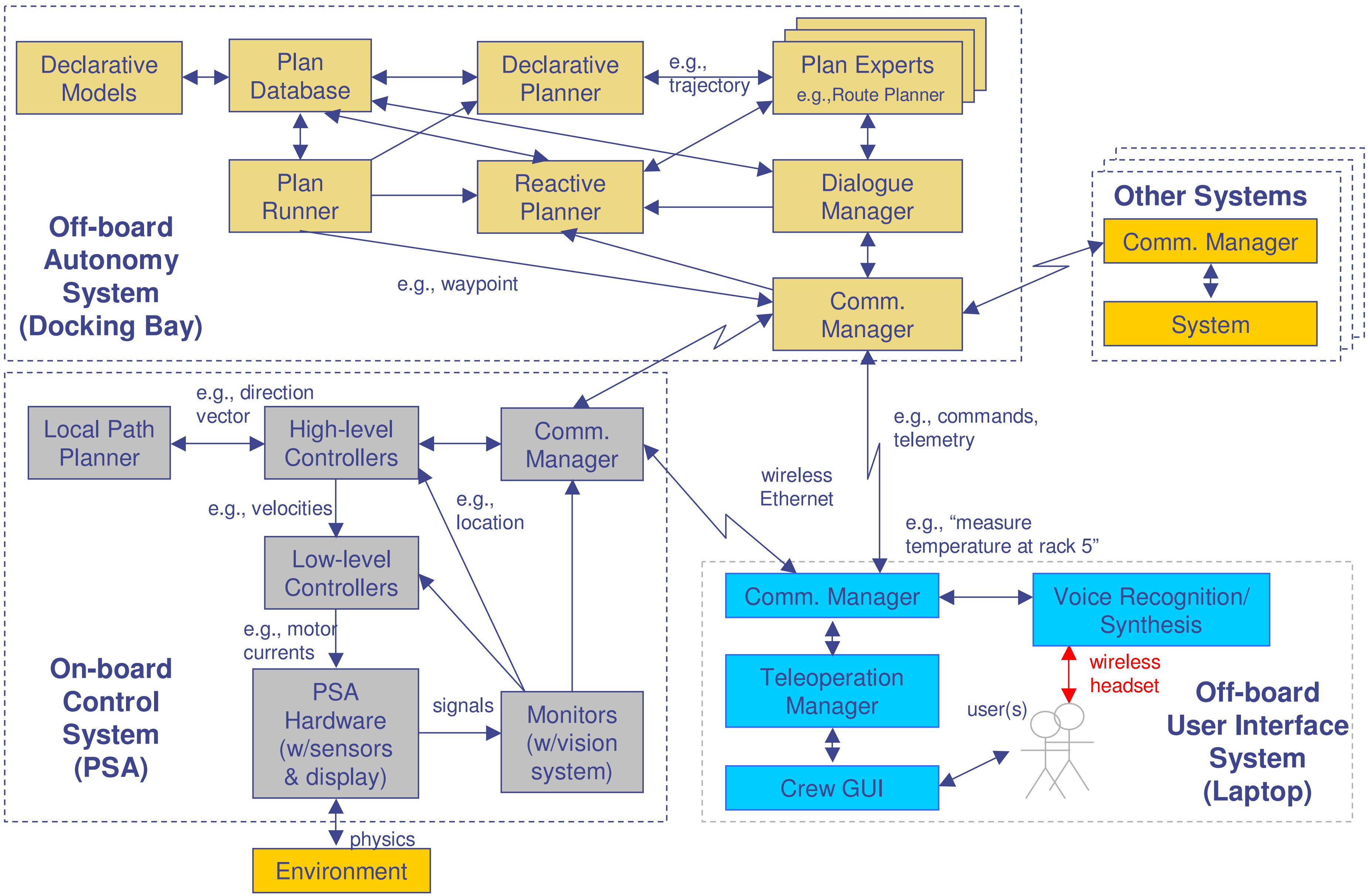
PSA Model 3 Propulsion & Attitude Control Testbed Design

Contains internal reaction wheels for orientation control



Blowers

PSA Top-level Software Framework



Sample Executed Plan Fragment

Ames Research Center

<table>
<tr><td>Scenario F: INITIAL_STATE</td><td colspan="27"></td></tr>
<tr><td>Time:</td><td>0</td><td>4</td><td>8</td><td>12</td><td>16</td><td>20</td><td>24</td><td>28</td><td>32</td><td>36</td><td>40</td><td>44</td><td>48</td><td>52</td><td>56</td><td>60</td><td>64</td><td>68</td><td>72</td><td>76</td><td>80</td><td>84</td><td>88</td><td>92</td><td>96</td><td>100</td><td>104</td></tr>
<tr><td>Timeline Name</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td><td>|</td></tr>
<tr><td>Agent_Mode_SV</td><td colspan="27">Agent_mode(Nominal)</td></tr>
<tr><td>Planner_SV</td><td colspan="4">Planner_Idle()</td><td colspan="2">Planning(0,120,OK,1)</td><td colspan="12">Planner_Idle()</td><td colspan="2">Planning(0,240,OK,1)</td><td colspan="7">Planner_Idle()</td></tr>
<tr><td>Planning_Horizon_SV</td><td colspan="8">Planning_Horizon(Intermediate_Horizon,0,32,0,120)</td><td colspan="19">Planning_Horizon(Intermediate_Horizon,32,120,0,240)</td></tr>
<tr><td>Goal_Task_SV</td><td colspan="8">GOAL_IDLE()</td><td colspan="14">GOAL_SENSE_AT_LOCATION(USLab,Rack5,Locker1,1,undef,8s)</td><td colspan="5">GOAL_IDLE()</td></tr>
<tr><td>Goal_Position_SV</td><td colspan="4">GOAL_LOCALIZE_POSITION_(USLab,Rack7,Locker2,1)</td><td colspan="4">GOAL_MAINTAIN_POSITION(U.S.Lab,Rack7,Locker2,1,PSA_Dock)</td><td colspan="12">GOAL_MOVE_TO_POSITION([50 100],[-180 -120],[80 100],[-100 -50],[60 120],[-100 -80],</td><td colspan="2">GOAL_MAINTAIN_POSITION(8)</td><td colspan="5">GOAL_MAINTAIN_POSITION([-100 -5 360],[360 -360],[360 -360],U.S.Lab,R</td></tr>
<tr><td>Path_Planner_SV</td><td colspan="8">PATH_PLANNER_IDLE()</td><td colspan="4">PATH_PLANNING(...)</td><td colspan="15">PATH_PLANNER_IDLE()</td></tr>
<tr><td>Path_SV</td><td colspan="12">PATH_IDLE()</td><td colspan="8">PATH(...)</td><td colspan="7">PATH_IDLE()</td></tr>
<tr><td>Motion_Command_SV</td><td colspan="12">Station_Keep(100,-180,100,[-360 360],[-360 360],[-360 360])</td><td>WAYPOINT_COMMAND(1,1,2,51,-135,81,0,0,0,OK,1)</td><td colspan="4">WAYPOINT_COMMAND(1,2,3,-85,-43,-24,0,0,0,OK,1)</td><td colspan="2">WAYPOINT_COMMAND(1,3,4,-85,62,-90,0,0,0,OK,1)</td><td colspan="8">Station_Keep()</td></tr>
<tr><td>Sensed_State_SV</td><td colspan="2">SENSED_STATE_INIT()</td><td>SENSED_STATE(0,...)</td><td>SENSED_STATE(1,...)</td><td>SENSED_STATE(2,...)</td><td>SENSED_STATE(3,...)</td><td>SENSED_STATE(4,...)</td><td>SENSED_STATE(5,...)</td><td>SENSED_STATE(6,...)</td><td>SENSED_STATE(7,...)</td><td>SENSED_STATE(8,...)</td><td>SENSED_STATE(9,...)</td><td>SENSED_STATE(10,...)</td><td>SENSED_STATE(11,...)</td><td>SENSED_STATE(12,...)</td><td>SENSED_STATE(13,...)</td><td>SENSED_STATE(14,...)</td><td>SENSED_STATE(15,...)</td><td>SENSED_STATE(16,...)</td><td>SENSED_STATE(17,...)</td><td>SENSED_STATE(18,...)</td><td>SENSED_STATE(19,...)</td><td>SENSED_STATE(20,...)</td><td>SENSED_STATE(21,...)</td><td>SENSED_STATE(22,...)</td><td>SENSED_STATE(23,...)</td><td>SENSED_STATE(24,...)</td></tr>
</table>

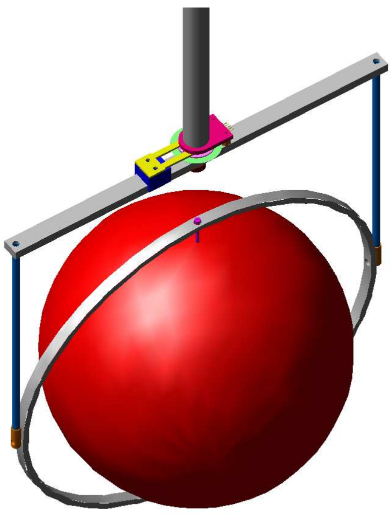
Faulty Sensor Scenario

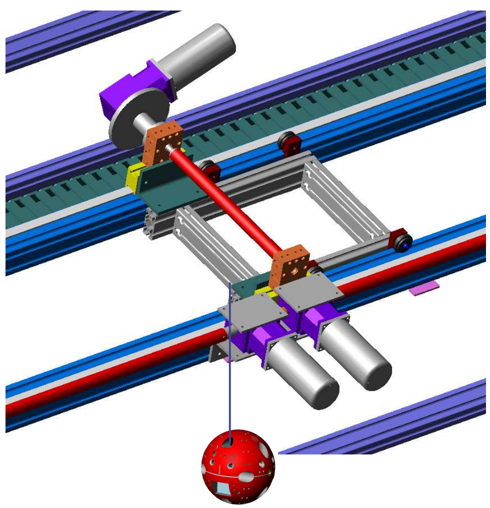
Time (s)	PSA	Operator	Life Support Agent (ECLSS)
T – 10	PSA starts planning upon receipt of command to go to the fixed sensor location and measure temperature		Detects a high heat signal from a fixed ISS node sensor and commands PSA to verify the temperature at that location
T – [10 1]	PSA completes plan		
T + 0	Starts executing plan		
T + [30 60]	PSA arrives at fixed sensor and begins collecting temperature data and sending it to ECLSS		
T + [60 90]			Determines fixed sensor is faulty, uses PSA as temporary sensor, requests crew to repair sensor
T +100		“Repairs sensors”	
T + 105			Fixed sensor signals actual temperature
T + 110		Requests PSA to compare fixed sensor reading with its own sensor	
T + 112	PSA queries ECLSS for fixed temperature sensor reading		
T + 114			Responds to PSA temperature query
T + 116	PSA replies to Operator query		
T + 118		Operator commands PSA to resume autonomous operation	
T + 120	PSA queries ECLSS for commands		
T + 125			Commands PSA to return to dock
T + [125 200]	Returns to docking locker		

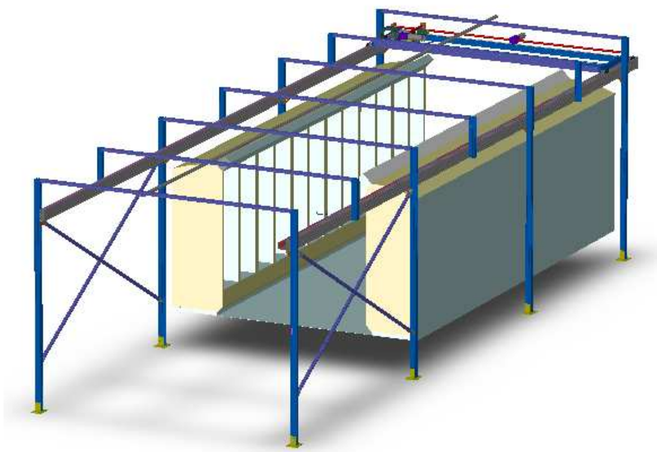
Micro-gravity Test Facility

Ames Research Center

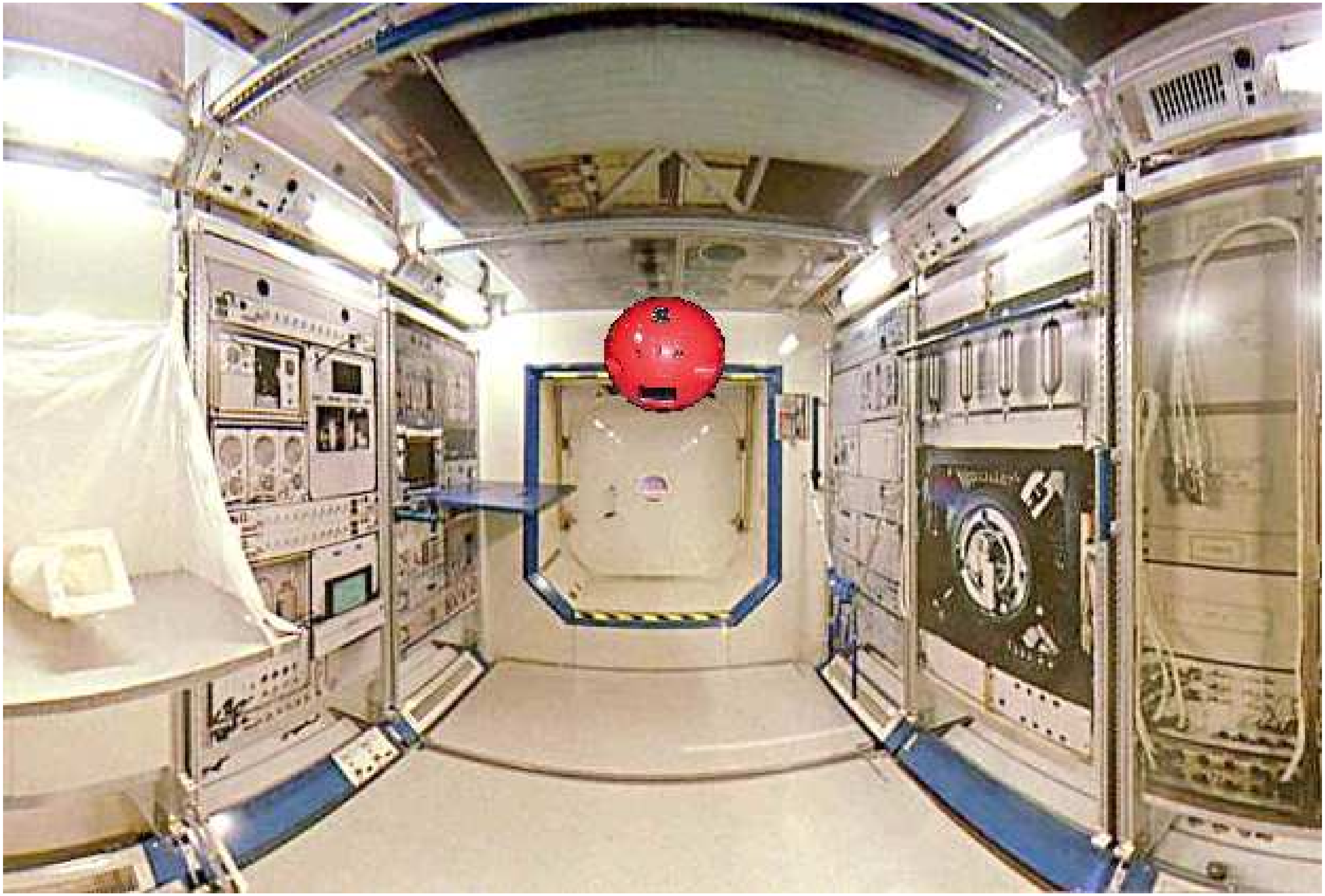
- Current Status
  - X&Y axes can be operated independently or in coordinated motion, from either remote or local control
  - A yaw-only gimbal with encoder feedback has been created to support the Model 2 sphere
- Work due 9/30/01:
  - Implement z-axis mechanism
  - Implement ISS racks
  - Implement static human sling





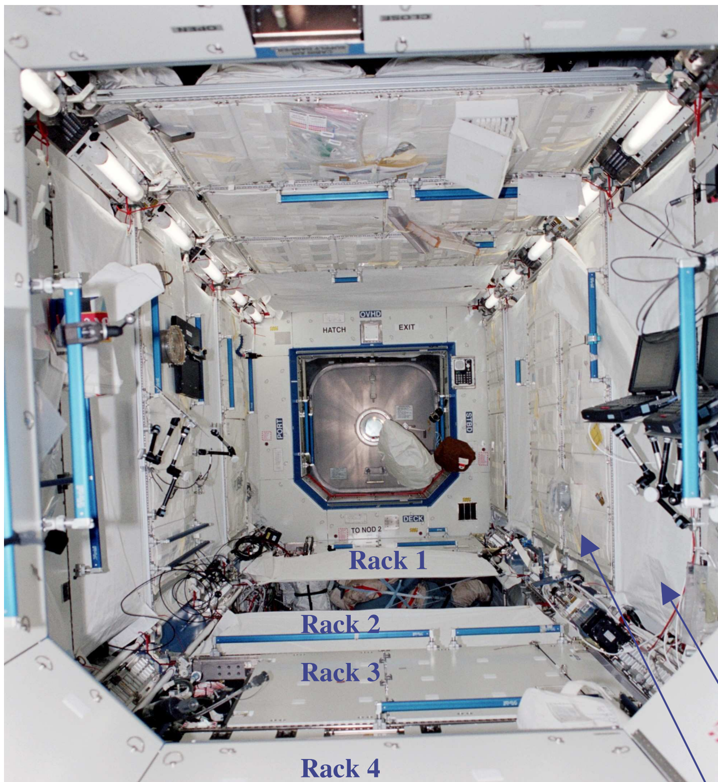


PSA in ISS Module Micro-gravity Test Facility



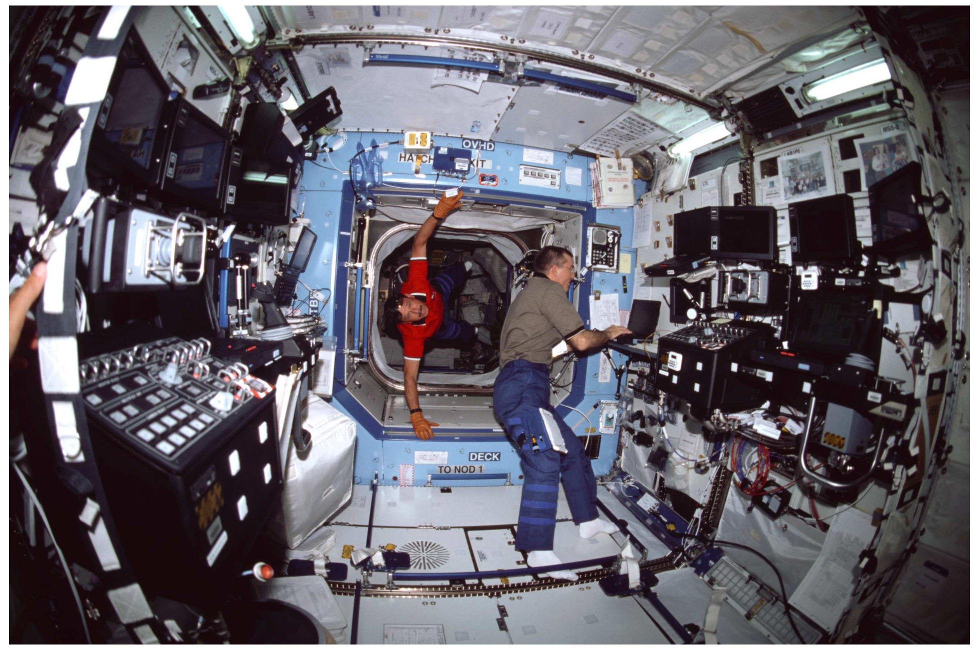
Actual U.S. Lab Module

Port Starboard



View looking to Node 2 - Forward

Starboard Port

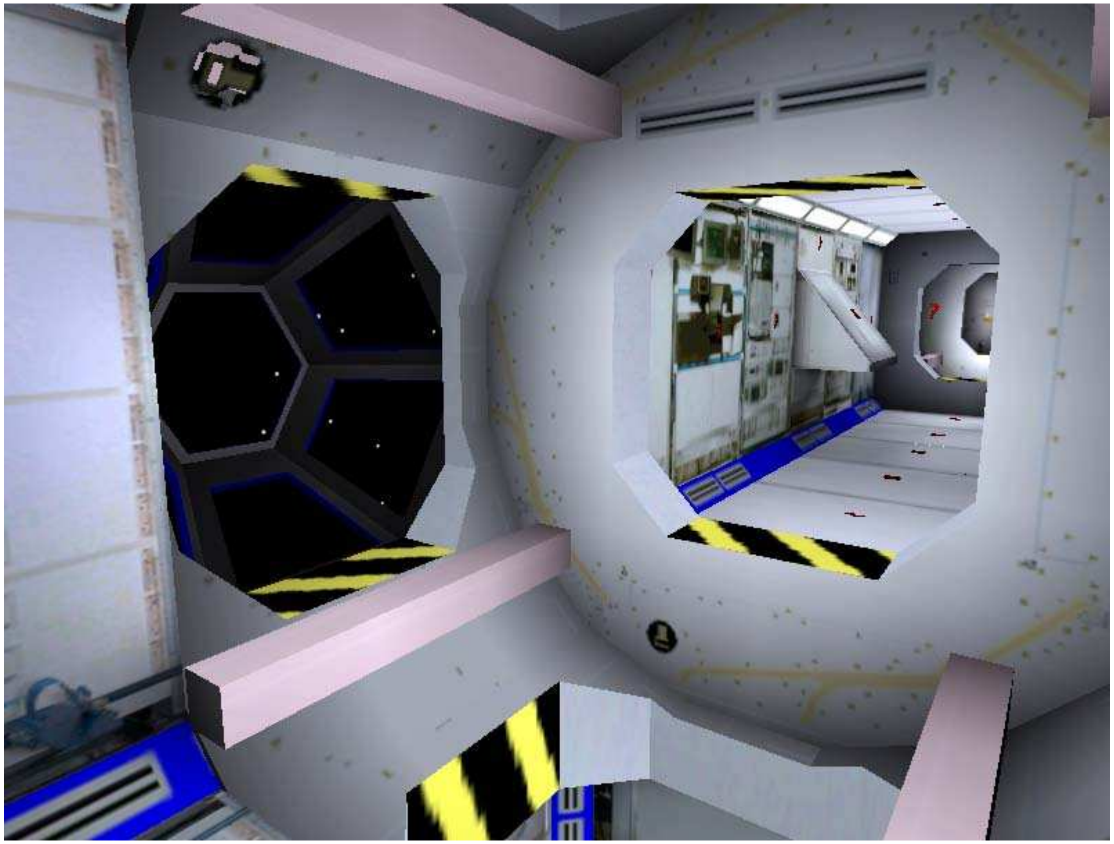


MSG

HRF

View looking to Node 1- Aft

PSA ISS Dynamic Simulator



May be made public given sufficient interest

FY02 Accomplishments

- Developed an initial multi-agent adjustably autonomous control system and implemented it on a spacecraft mobile robot prototype – *done in conjunction with IDEA and EUROPA projects*. Demonstrated simple autonomous control tasks in simulation and 3-DOF test facilities.
- Developed a spacecraft micro-gravity test facility for spacecraft mobile monitors. Currently operational in X,Y, and yaw dimensions with Z, pitch, and roll due by 9/30.
- Implemented a vision-based people-tracking algorithm on testbed.
- Developed 3D path planning algorithms for controlling an internal spacecraft mobile robot in the ISS.
- Implemented an internal ISS simulation with PSA and dynamic objects that integrates a commercial physics simulator with a 3D renderor – *done in conjunction with Skunkworks project and ECS program.*
- Enhanced GUI with 3D renderor

FY03 Scheduled Work

- Enhance initial multi-agent adjustably autonomous control system and demonstrate complex multi-agent tasks with the PSA Model 2 in the 6-DOF micro-gravity test facility and in the high-fidelity ISS simulation. The demonstrations will include :
  - Scenario A: Robust generation of an ISS node environment map (including operator interruption)
  - Scenario B: Multi-agent diagnosis of an ISS node fault
  - Scenario C: Fault Detection and Cooperative diagnosis and recovery with multi-agents and human interaction
  - Scenario D: Cooperative (human & mobile agent) Data Collection and Crew Instruction for Performing Interactive Mission Science Experiments
  - Scenario E: Long-term mixed-initiative planning and optimization
- Demonstrate 6-DOF localization and map registration *(jointly with ECS program)*
- Enhance GUI to support adjustably autonomous control of PSA

Spacecraft Mobile Robot

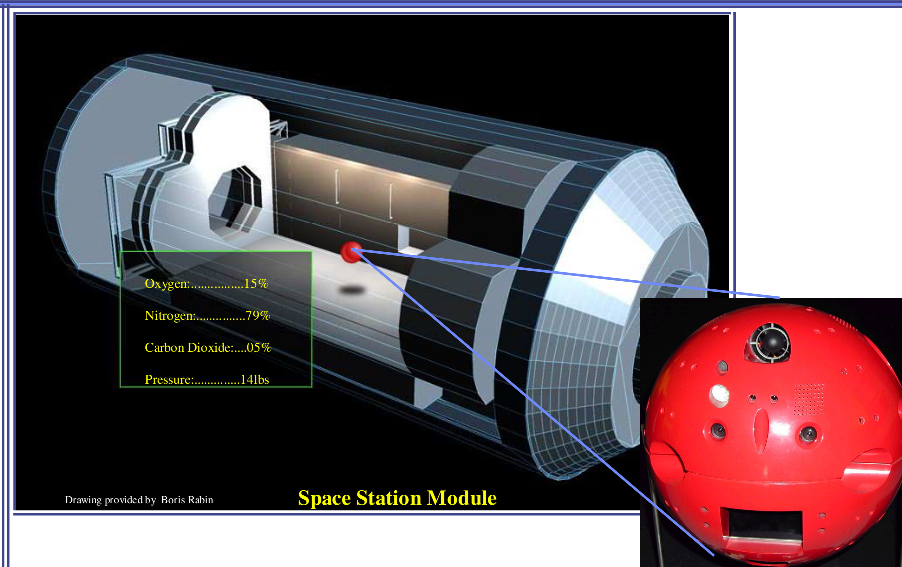
Gregory A. Dorais/ARC and Yuri Gawdiak/ARC/HQ

Ames Research Center

Goal: Demonstrate an active IVHM and crew support prototype system, including adjustably autonomous fault diagnosis & repair, capable of improving crew safety and productivity in ISS *via* an autonomous mobile robot with environment sensing and video-communication capabilities

Objectives:

- Spherical fan-propelled robot with cameras, proximity and atmospheric sensors for navigation, monitoring its environment, and video conferencing
- Adjustably autonomous control software supporting mixed-initiative interaction at plan and execution time
- Vision-based state estimation system



NASA Relevance:

The high-level autonomy and state-estimation software will be applicable to a wide variety of mobile and stationary control applications. On ISS, the PSA will increase crew performance and reduce mission risks by:

- performing routine environment monitoring tasks and playing a support role in detecting and responding to off-nominal situations, e.g., detect source of leak or heat
- acting as a mobile sensor suite for life support system and support Integrated Vehicle Health Management (IVHM)
- providing a mobile computer terminal, checklist support, task scheduling, and remote video camera for the crew

Accomplishments:

- *FY00* Developed PSA Model 1: teleoperated in 3-DOF
- *FY01* Developed PSA Model 2, demonstrated 3-DOF visual servoing, developed 3D low-fidelity simulator
- *FY02* Developed autonomous control system including IDEA & EUROPA. Demonstrated on PSA Model 2 in simulation and on micro-gravity test facility

Schedule:

FY03: Enhance autonomous control system and demonstrate multi-agent adjustably autonomous task execution in 6-DOF test facility and simulator in multiple scenarios