POWER SYSTEM AGENTS: THE MOBILE AGENTS 2006 FIELD TEST AT MDRS

William J. Clancey
NASA/Ames Research Center
Intelligent Systems Division MS269-3
Moffett Field, CA 94035
and
Florida Institute for Human and Machine Cognition, Pensacola, FL
William.J.Clancey@nasa.gov

Maarten Sierhuis
USRA/RIACS (NASA/Ames)

Richard L. Alena
NASA/Ames Research Center

John Dowding
UC/Santa Cruz, CA (NASA/Ames)

Michael Scott and Ronnie van Hoof
QSS Group, Inc. (NASA/Ames)

ABSTRACT

During the Mars Society's Mars Desert Research Station (MDRS) Rotation 49 (April 23-May 7, 2006), we field tested and significantly extended a prototype monitoring and advising system for the habitat crew that integrates data from the MDRS power system. A distributed, wireless network of functionally specialized agents interact with the crew to provide alerts, access and interpret historical data (through voice command), and display troubleshooting procedures. In practical application during two weeks, the system generated speech over loudspeakers and headsets to alert the crew about the need to investigate generator-inverter-battery problems that had not yet come to their attention. This field test accomplished a milestone for the "CEV Agent-Mediated Situational Awareness" task in NASA's Exploration Technology Development Program 12B, Human-Systems Interaction.

INTRODUCTION

The Mobile Agents Project has developed and field tested over five years a general technology for integrating diverse hardware and software in a wireless network. The purpose of the project has been to develop an Extra-Vehicular Activity (EVA) communications system that automates data management, workflow between crew and robots, and alerting and advising for navigation, scheduling, life support, and astronaut health.

The Mobile Agents system is based on the Brahms multiagent work practice simulation language[1,2]. As previously reported, design of Mobile Agents and its particular configurations is based on historical analyses, baseline studies of field science, experience in simulated missions, and previous MDRS field tests[3,4,5]. In particular, field tests during 2002-2005 were conducted in Arizona (Desert RATS) and Utah to demonstrate and experiment with prototype human-robot operations concepts[6], such as collaborative planning with a remote science team, reconnaissance mapping by robot prior to an EVA, voice commanding of the robot and software systems, having the robot follow and watch designated astronauts, and using the robot as an EVA workbench (e.g., to carry samples and a printer). For 2006 the Mobile Agents Project team was asked to focus on spacecraft or surface habitat "situational awareness." Based on previous experience at MDRS, we chose to develop an agent system that would help the crew understand the state of the MDRS electrical system.

The configuration and experiments during MDRS49 demonstrate the generality and flexibility of using Mobile Agents for systems integration, namely "agent-based systems integration." In particular, telemetry from multiple electrical sources is logged and analyzed, such that the crew may use voice commanding to inquire about current and historical data. Inquiries are possible anywhere in MDRS through a wireless (Bluetooth) headset. The crew could set alarms for events (e.g., when the amperage used by the habitat exceeded a certain value) and receive either personal or loudspeaker broadcast alerts. Procedural advice for handling power system problems was displayed automatically when the system detected common error conditions (e.g., low battery voltage). Based on the first week's experience, the crew designed and improved new capabilities that were demonstrated during the second week, including a new command to send voice mail to other crew members.

BACKGROUND: AGENT-BASED SYSTEMS INTEGRATION

Since 1992, the Brahms Project has been developing a computer language (including a model-building tool and compiler) and simulation environment (including an agent-viewer timeline for displaying agent interactions) for modeling and simulating how software "agents," people, tools and facilities interact in practice in real-world settings. A model of practice represents behavioral interactions (e.g., meeting someone on the stairs and having a conversation) in contrast with task and process models that abstract away such details to represent the (idealized) functions that workers accomplish.

The Mobile Agent Architecture, under development at NASA since 2001, applies the Brahms language in a real-time environment, providing a flexible toolkit for interoperability of diverse system components (e.g., robots, instruments, biosensors, cameras, automated life support systems, displays). The agents in the model are software programs that enable people to receive data from and send commands to automation systems in the world (e.g., an inverter). Application is relevant to integrating habitat and surface crew operations, across a wide range of time scales and distributed interactions, including: crew planning, systems monitoring, human-systems communication, and supervisory functions (e.g., the work of CapCom in Apollo).

Through a variety of experimental configurations (see Figure 1) including graphic animation, integration with other simulations, graphic planning and scheduling, robotic and device control, integration with databases and email, and—during MDRS49—integration with a physical life support system (the MDRS power system), we have come to realize that the Mobile Agents technology provides a general solution to the well-known problem of systems integration. In particular, a distributed agent system constitutes a "data exchange layer" with the following features:

- The system integration is *model-based*, namely using semantic representations of crew activities, crew communications, and component configurations.

- The agent-based integration capability is *dynamically reconfigurable* (i.e., components can be added or deleted while the system is running).

- Most significantly, by incorporating crew activity plans, real-time activity monitoring, and location data, on top of the *data flow automation* (e.g., linking devices to databases), the resulting system provides *proactive work flow management* (e.g., scheduling and navigation assistance).

Agents have been used to implement context-sensitive protocols for monitoring what people and robots are doing, and transmitting alerts aurally, by email, and by posting on the web (e.g., RSS feed).

What are the essential features of the Mobile Agents architecture that have made possible this wide-ranging systems integration? Our experience over six years integrating five robots, biosensors, cameras, etc. with a variety of graphic interfaces and software systems, including planning and scheduling tools, suggests that key to success has been implementing modular functions, which are essentially programs, as software agents that have the following features:

- Located on distributed, mobile, networked machines.

- Store data about the world and previous interactions as *Beliefs* (object-attribute-value proposition).

- React to a shared model of the world (*Facts* about *Objects*) that is geographically partitioned (thus only accessible according to the location of the agent) and affected by agent actions.

- Have conditional actions (*Workframes*) and inference rules (*Thoughtframes*) organized into chronological behaviors (*Activities*). These may change the agent's Beliefs and world Facts. The Brahms Executive manages the real-time application of actions and inferences through a relatively complex composition, prioritization, and interruption scheme.

- Communicate with each other via *Asking* about and *Telling* beliefs, using a well-defined agent communication language.

- Integrate with arbitrary software and hardware via *Communication Agents* that are implemented in JAVA and interact with the external system's API.

- Are networked via a *Directory Service* that enables communicating with remote agents (i.e., running on other platforms).

- Assume the network is unstable and deal with communication outages.

Integrating a new system, such as the Brand Electronics OneMeter power monitoring system, involves the following process:

- Specify interaction modes between people, systems and agents (e.g. speech dialogue interactions), which will determine required capabilities of the agents and systems.

- Identify or develop the API protocol for the hardware/software system that will be integrated (i.e., functions and parameters for accessing external data or controlling the external system).

- Specify functions that the agent communicating with the external system (the "assigned Agent") must provide. In most cases, this involves working backwards from the desired voice commands or the graphical user interface (GUI) such as a button display by which a person will access data and give commands.

- Write Communication Agent in JAVA to mediate between the external systems API and assigned Agent, that is, to provide the data and control functions required by the agent.

- Formalize the utterances, including alternative phrasings, that will be used to access data and give commands. Integrate these utterances in the RIALIST system through a tuning and compilation process[7].

- Relate Personal Agents (for crew members) with functional agent assistants (e.g., Navigation Assistant) to the assigned Agent via communication actions in Workframes (i.e., conditional actions that receive data or make requests).

POWER AGENT SYSTEM CONFIGURATION

The MDRS 49 power system consisted of an external generator, batteries under the habitat, solar panels, and inverter (Figure 2). MDRS Crew 47 had previously deployed the OneMeter (Brand Electronics) electric metering system (Figure 3) to instrument the various power sources.

The Mobile Agents configuration consisted of six laptop computers (Figure 4). One networked computer (HabCom) was placed on the upper deck, connected to loudspeakers. Four other laptops were paired with wireless (Bluetooth) headsets that enabled crew members to interact with their personal agents from anywhere in the hab (plus in one test as far as 10 meters outside). A fifth laptop functioned as a telenet server for the OneMeter device, constituting the Power Support System.

Referring to Figure 4, the Power Support System contains four agents:

1. One Meter Communication Agent (OneMeter CA)—serves as a telnet server. The OneMeter connects to this telnet server every 15 minutes (a fixed property of the OneMeter firmware) and sends the data for all channels. The agent parses and packages the data and sends it to the Power Monitoring Agent. The data is also stored in a MySQL database, which is used for responding to historical queries.

2. Power Monitoring Agent—processes incoming data from the OneMeter system, as well as answers queries about historical data

    a. Incoming Data: On every new reading received from the OneMeter Comm Agent, evaluates the values for the channels using a set of rules. The agent is designed to detect five anomalous events (e.g., impending shut-down of inverter due to low battery voltage). If an anomalous event is found, the agent sends the condition data to the Problem Resolution Agent for analysis. It forwards the alert with the procedure to the Notification Assistant in the Habitat Central Command System (HabCom laptop – see Figure 4), which distributes the alert and procedure to the Personal Agents of the crew and Habcom systems that have subscribed to this alert. The procedure is extracted by these Personal Agents and sent to their respective Procedure Assistants, which display the procedure on the associated display. The alert message is also sent to the respective Dialog Agent for verbal notification.

    b. Historical Queries: On receiving a power system inquiry from a crew member or HabCom Personal Agent, sends the query to the OneMeter Communication Agent, and returns the desired data (e.g., "what was the maximum habitat amperage since yesterday?").

3. Power Problem Resolution Assistant—Receives an error condition event from the Power Monitoring Agent; determines the procedure to be followed to resolve the problem; and sends that procedure back to the Power Monitoring Agent.

4. Xantrex Communication Agent (Xantrex CA)—on request, provides parameter settings for the Xantrex Inverter to the Power Monitoring Agent.

The MDRS49 configuration also included the activity plan loading and tracking functionality used for previous EVA simulations. In particular, crew members had daily schedules that indicated what activity they would normally be doing at a given time. Crew members could inquire about anyone's current activity. This was part of a related investigation of how a capability such as the Power Monitoring System might be augmented by information about the crew's activities.

The capability for crew members to send voice notes to each other was developed during MDRS49. Voice notes are recorded as usual (by a command and confirmation, with a pause causing termination of recording). Voice notes are sent by the Dialog Agent to the Science Data Assistant on the respective crew member system (Figure 4), which sends the file to the Data Collector Communication Agent, which transmits it to the Data Manager (on the HabCom laptop). The Data Manager sends the file to the Compendium system for storage (in other field tests, the ScienceOrganizer database at Ames Research Center was used instead, enabling access from remote support teams, a capability that was not relevant to MDRS49 experiments).

The originating Dialog Agent also sends the voice note to the Voice Mail Client, which transmits the file to the Voice Mail Server (on the HabCom laptop). The Server sends the voice note to the intended recipient at the specified time.

VOICE COMMANDING

Using voice commanding through the headsets, as well as by speaking directly into the HabCom computer, crew members could at the spur of the moment ask any of the following questions about the MDRS power system ("|" designates alternative terms or phrases; "{}" designates optional terms or phrases):

- What is the {battery | generator} {volts | amps | volts and amps}?
- What is the status of the {generator | inverter | battery | solar panel}?
- Are the batteries charging?
- What is the hab{itat} {power usage | volts | voltage | amps | volts and amps}?
- What is the low battery cut {off | out voltage}?

Note that the phrasings given here are intended for experimental use, not our recommendations. Different people might prefer alternative phrasings, and the system could be augmented to allow these alternatives. We expect that voice commands, responses, and other aspects of human-agent interactions would be refined through more extended practical use in an analog environment like MDRS.

Before arriving at MDRS we tested the system (Figure 4) with simulated power system data. During the first week, we completed the system integration with the OneMeter system and verified the operation of the voice commands.

From direct experience maintaining the diesel and gas generators during the first week, we designed extensions to the system to allow historical inquiries, providing data not available directly at the inverter or OneMeter displays:

- What was the average hab{itat} {amps | volts | voltage} since <#> {AM | PM}?
- When did the {generator | batteries} change status?
- When did the batteries start {dis}charging?
- What was the {hab{itat} | battery} {amps | volts | voltage} at <#> {AM | PM}?
- What was the maximum hab{itat} {amps | volts | voltage} {today | this morning | this afternoon | this evening | yesterday | last night}

Other extensions included more flexible alerting, set by any crew member ("when" alerts the first time; "whenever" alerts every time):

- Tell {me | <person> | everyone} when{ever} the generator goes offline.

- ❖ Tell {me | <person> | everyone} when the hab{itat} {amps | volts | voltage} {exceeds | drops below} <#>.

We also made it possible to know in general what other people were doing, according to the predetermined activity model stored in the agent system:

- ❖ What is {my | <person's>} current activity?

Based on our experience in the hab, we added an ability to send voice messages to each other:

- ❖ {Send | Take | Record} {a} voice note {{for | to} <person>}
- ❖ Send voice note <#> to <person> {at <time>}

Like text messaging, the voice messaging capability allowed us to exchange information without interrupting what we were doing. The "time" option allows setting reminders for oneself or other people.

In the original Mobile Agents configuration (2002-2004), every command required confirmation. As response time because short and functionality reliable, we eliminated this confirmation for non-critical actions (e.g., creating a sample bag) or those requiring possible immediate action (e.g., telling a robot to halt). We also added additional checking for more dangerous actions (e.g., telling the Scout rover to move). However, we found that some confirmation was still required that the system had interpreted an utterance as being a command (recall that the RIALIST Speech System uses the "open microphone" mode, i.e., it is continuously processing what the person says, attempting to recognize a command).

As part of our experiments with the Dialog Agent during MDRS49, we implemented a short tone that indicates a crew member's statement has been interpreted as being a command and that a response is pending. We found this immediate feedback to fit the three common situations when response occurs almost immediately as perceived by the person, response requires a second or two, or a response is delayed.

RESULTS

The overall configuration and voice commanding, all several levels, including OneMeter-Agent integration, Bluetooth wireless networking, and the six platforms (Figure 4) worked very well. The new voice commands were functional by Thursday of the second week of MDRS49 and fully exercised in a variety of situations in the upper-deck galley and work area, outside by the generator (about 50 meters south, using a wireless relay), and by the inverter on the lower deck. We demonstrated a scenario whereby the engineer (Alena) investigated the generator-habitat performance and loads over the past day while inspecting the inverter on the lower deck, went outside to restart the generator, and communicated with the commander using voice notes about his findings and the system status.

The only serious limitation to the system, restricting our ability to exercise the functionality, was the 15 minute frequency for receiving data from the OneMeter system. This meant for example that if a crew member overloaded the electrical system, an alert would not be received until as much as 15 minutes after a problem occurred. Obviously, a sensor for an electrical system should transmit data immediately to monitoring and controlling systems. During MDRS49 we received a firmware update from the vendor that changed the time delay to 5 minutes, but we decided it was unwise to modify the OneMeter system in the middle of our testing.

For the Mobile Agents team, the habitat scenario and Bluetooth headsets provided the first opportunity to use the system, which had of course been developed over the years for scientists to use during EVAs. We found the ability to interact with a Personal Agent extremely satisfying, as it provided information anywhere (e.g., while reading in a stateroom) using a pleasing and intuitive interaction. The crew laptops were all located on a table on the lower deck, hence one had a hands-free and computer-free experience—even better than carrying the computer in your backpack, as had been required in our prior EVA field tests. Indeed, one had no impression of interacting with a computer at all; the experience is much more like talking to someone on a telephone, though without having to dial a number to get attention—you simply ask and get an answer, wherever you are, whatever you are doing.

As a result of this experience, it became obvious to the crew that this hands-free, display-free—no visible computer—mode is precisely how we would like to get and receive certain kinds of information. For example, we generated a variety of new commands (that were well beyond the scope and time of the MDRS49 effort) that would enable us to hear news, have text messages read to us, search and hear recent testimony in Congress, and so on. We also realized that there are limitations to speech as a medium, particularly for numerical data, which may need to be spelled out or confirmed on a display. For example, the video from our experiments had subtly different audio quality that makes it difficult at times to know what number the computer voice is saying (e.g., fifteen or sixteen?). In practice, an astronaut may want to glance down to a watch-like device to see a number simultaneously displayed as it is spoken.

The idea of voice notes turned out to be a serendipitous discovery and significant extension of the Mobile Agents concept. We used voice notes for communicating with crew members whose location was unknown and for filing reminders for ourselves. Unlike phone mail, one need not dial-up to hear a message, and may use the voice commanding system to control playback. In subsequent tests at Desert RATS 2006 (now called D-RATS) in Arizona in September 2006, an astronaut and other subjects using pressurized suits found the voice messaging capability to be extremely useful. In particular, voice notes provided a back channel for communicating with mission support in Houston and getting assistance from Mobile Agents personnel on site (during the EVA). Voice notes are more convenient than typing an email, less intrusive than a phone call, and will provide a friendly voice when conversations in distant space are not possible.

OTHER ACCOMPLISHMENTS DURING MDRS49

During this rotation we also accomplished the following:

- Successfully used iMAS (individual Mobile Agents System) for exploration EVAs by a geologist operating off the network and downloading data upon return to the habitat.

- Deployed and used the Crew Activity Analyzer developed by Foster Miller[4] to video and audio record the upper deck, synchronized with records of crew movements using an RF beacon-listener system ("Crickets"). Crew interactions with the agent system on the upper deck were thus recorded for five days.

- Represented the generator restart procedure in a formal procedural language and developed methods to integrate the agent system with an "executive" system interpreting the procedure. This method could provide a more rigorous method for tracking crew actions relative to procedures, especially for monitoring and advising a team of people whose actions are interacting.

- Using actual crew activities, developed a model that could be used in the future to help crew members and automated systems relate tasks and problems to where crew members are located and what they are doing.

Articles about MDRS Rotation 49 appeared in the *New Scientist*, *Technology News Daily*, *Space Daily*, *Flight International*, and Spaceref.com.

FUTURE MISSION SCENARIO

The following imaginary story is an example of how the Power Monitoring System could be deployed for lunar missions.

> A lunar exploration crew is about to enter the Lunar Habitat for a prolonged stay on the lunar surface. The PHALCON (Power, Heating, Articulation and Lighting Console) flight controller had commanded the Power Monitoring Agent (PMA) to start the power generator for the hab, standing a kilometer away, three days before the crew arrived. Over the next three days, the PMA monitors the habitat's power inverter. The PMA provides the PHALCON with an hourly status report, and the PHALCON can ask specific status questions as desired.
>
> When the crew arrives, they can ask the PMA status information via voice commanding, as well direct information to be displayed in preferred ways on any monitor. In particular, the engineer has configured his watch to present certain data via a touchscreen. The PMA monitors the crew's power consumption, informing them trends and answering questions about possible correlations.
>
> One night, while the crew is sleeping, the generator malfunctions briefly. The PMA communicates an alert to mission support on earth. After receiving diagnostic information, the PHALCON asks the PMA to reconfigure a circuit. The PMA communicates that everything is back to normal. The two-person "Gemini" support team in Houston decides to let the crew sleep. When the crew awakes, the PMA provides a night power report, explaining what went wrong and how it was resolved. The crew asks

the system to keep the crew informed about certain power parameters every hour and sets a new alert with a threshold chosen to detect another failure.

One important risk that needs to be addressed with such a distributed agent system is eliminating a potential single point of failure. Communication between ground, crew, and the PMA could be interrupted. When this happens, the distributed system needs to follow protocols and construct workarounds to continue to provide necessary functions. For example, if communication back to mission control fails, and power is dangerously low, the PMA may need to wake up the crew to handle the situation.

CONCLUSIONS

The MDRS49 crew demonstrated a paradigmatic use of an analog research station for prototyping software that could be used for future space flight. First, on the basis of actual experience living and working in the habitat (2002-2005), we recognized the need and opportunity for a power monitoring system. Recognizing the limitations at the time of the Xantrex API, Crew 47 deployed the OneMeter system and provided us with information to integrate it with the agent system. We developed a prototype of voice commanding and alerting using the Xantrex operating manual, and tested this system in simulation prior to arriving at MDRS. We used our time at MDRS to deploy, test, debug, improve, and demonstrate a Power Monitoring System that directly applied the systems integration capability of the Mobile Agents Architecture. The system, subject to the limits of the telemetry available, functioned beyond our expectations, and prompted many new ideas about the use of wireless headsets, voice commanded access to data, and voice messaging.

One cannot help but notice how collaborations between MDRS rotations and accumulating improvements to MDRS are increasing the sophistication of the simulations we can carry out. Our work this year would not have been possible without the prior installation of the inverter and telemetry system by the Mars Society's engineering team and other crews. By building the MDRS facility, the Mars Society has as a side-effect created a community of volunteers across countries and disciplines, private and government. This developing network of friends and collaborators is far more important and lasting than any building we erect. We stand on each other's shoulders, sharing tools, methods and ideas, reaching ever closer to Mars.

ACKNOWLEDGMENTS

In addition to the authors of this paper, the MDRS Rotation 49 crew members included Paul Tompkins (NASA Ames), Brent Garry (Smithsonian Air and Space Museum), and Vandi Verma (NASA Ames).

We gratefully acknowledge the people who were in email contact before and during our stay in MDRS and helped us carry out this work:

- Emily Colvin & Jan Osburg (Georgia Tech) deployed the OneMeter system during Rotation 47.
- Ethan Brand (Brand Electronics, Inc.) provided the OneMeter API.

- Jim Murray and Judy Gertler (Foster Miller, Inc.) developed the Crew Activity Analyzer.
- Don Foutz (Whispering Sands Inn, Hanksville, UT) helped us keep the generators operating.
- The Mars Society Engineering and Mission Support volunteers monitored our activities and provided frequent advice for using the MDRS facility, particularly: Julie Edwards, Paul Graham, James Harris, Tony Muscatello, Shannon Rupert, Artemis Westenberg.

This project was funded in part by a NASA Research Announcement (NRA), "Intelligent Systems." Funding was also provided by the Exploration Technology Development Program (ETDP) within NASA's Exploration Systems Mission Directorate (ESMD): Exploration Systems Architecture Study (ESAS), Program 12B, "Human System Interaction" and ESAS 6E, "Spacecraft and Vehicle Autonomy."

For more information about Brahms and Mobile Agents see http://www.agentisolutions.com; papers and press releases are available at: http://WJClancey.home.att.net; MDRS49 reports at: http://www.marssociety.org/MDRS/fs05/

REFERENCES

[1] Clancey, W. J., Sachs, P., Sierhuis, M., and van Hoof, R. 1998. Brahms: Simulating practice for work systems design. *International Journal of Human-Computer Studies*, 49: 831-865.

[2] Sierhuis, M. 2001. *Modeling and Simulating Work Practice*. Ph.D. thesis, Social Science and Informatics (SWI), University of Amsterdam, SIKS Dissertation Series No. 2001-10, Amsterdam, The Netherlands, ISBN 90-6464-849-2.

[3] Clancey, W. J., Sierhuis, M., Alena, R., Crowford, S., Dowding, J., Graham, J., Kaskiris, C., Tyree, K. S., & van Hoof, R. 2004. The Mobile Agents integrated field test: Mars Desert Research Station 2003. *FLAIRS 2004*, Miami Beach, Florida.

[4] Clancey, W.J., Sierhuis, M., Alena, R., Berrios, D., Dowding, J., Graham, J.S., Tyree, K.S., Hirsh, R.L., Garry, W.B., Semple, A., Buckingham Shum, S.J., Shadbolt, N. and Rupert, S. (2005). Automating CapCom Using Mobile Agents and Robotic Assistants. *American Institute of Aeronautics and Astronautics 1st Space Exploration Conference*, 31 Jan-1 Feb, 2005, Orlando, FL. Available: [http://eprints.aktors.org/375].

[5] Pedersen, L., Clancey, W.J., Sierhuis, M., Muscettola, N., Smith, D.E., Lees, D., Rajan, K., Ramakrishnan, S., Tompkins, P., Vera, A., Dayton, T. (2006) **Field demonstration of surface human-robotic exploration activity.** AAAI-06 Spring Symposium: Where no human-robot team has gone before.

[6] Hirsh, R., Graham, J., Tyree, K., Sierhuis, M., and Clancey, W. J. 2006. Intelligence for human-assistant planetary surface robots. In A. M. Howard and E. W. Tunstel (Eds.) *Intelligence for Space Robotics***,** pp. 261-279. Albuquerque: TSI Press.

[7] Dowding, J., Alena, R., Clancey, W. J., Graham, J., and Sierhuis, M. 2006. Are you talking to Me? Dialogue Systems Supporting Mixed Teams of Humans and Robots. *AAAI Fall Symposium 2006: Aurally Informed Performance: Integrating Machine Listening and Auditory Presentation in Robotic Systems*, October, Washington, DC.

FIGURES

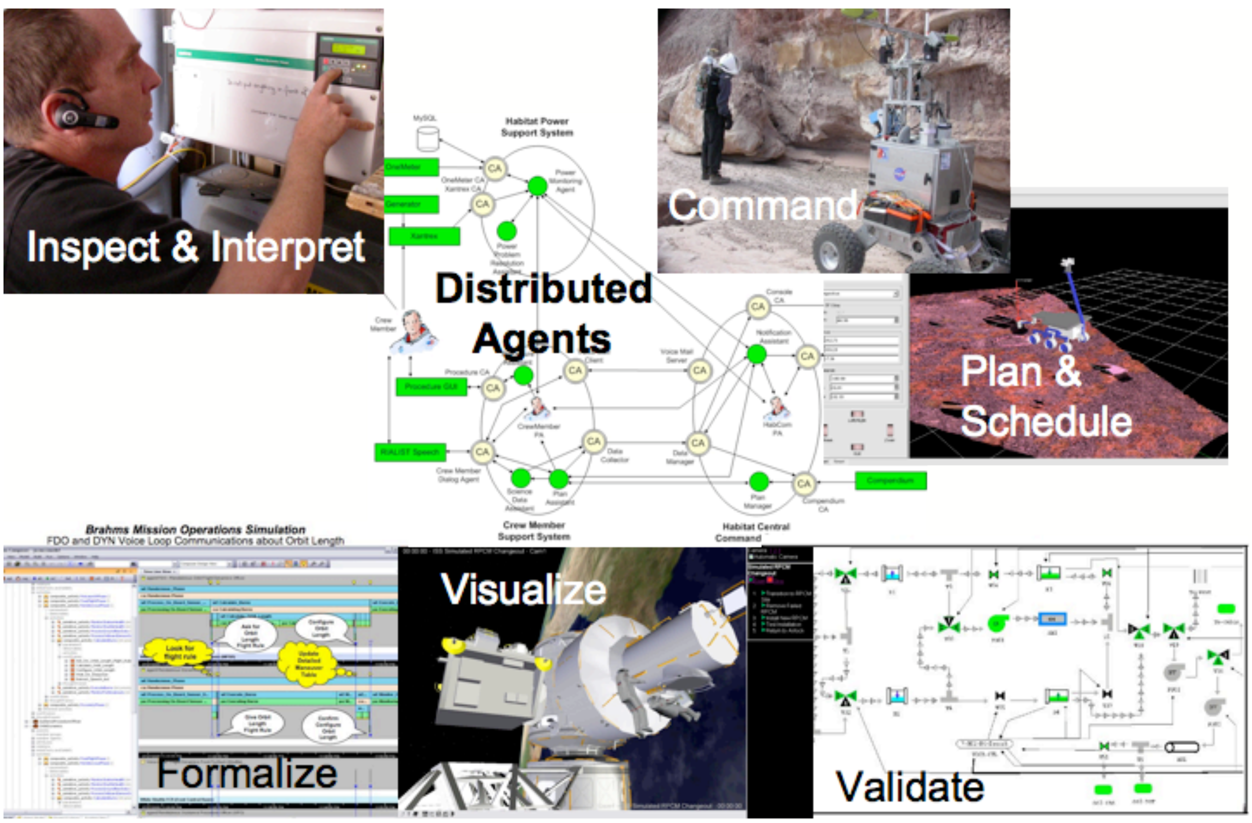

Figure 1. Agent-based system integration: A work system is modeled in the Brahms language of agents and activities (bottom left). As a simulation of work practice, Brahms has been used to drive visualizations and validate procedures by integration with other simulations (e.g., life support system). In the "real-time" configuration, called Mobile Agents, Brahms agents located on different computers interact with hardware and software systems in the real world and

communicate with each other using wireless networking. Examples include integration of a robot planning and scheduling system with graphic interfaces[5] and voice commanding a robotic assistant[4,6]. Photo in upper left shows an MDRS49 crew member making an historical inquiry via voice commanding while inspecting the inverter. (Graphic by Clancey).

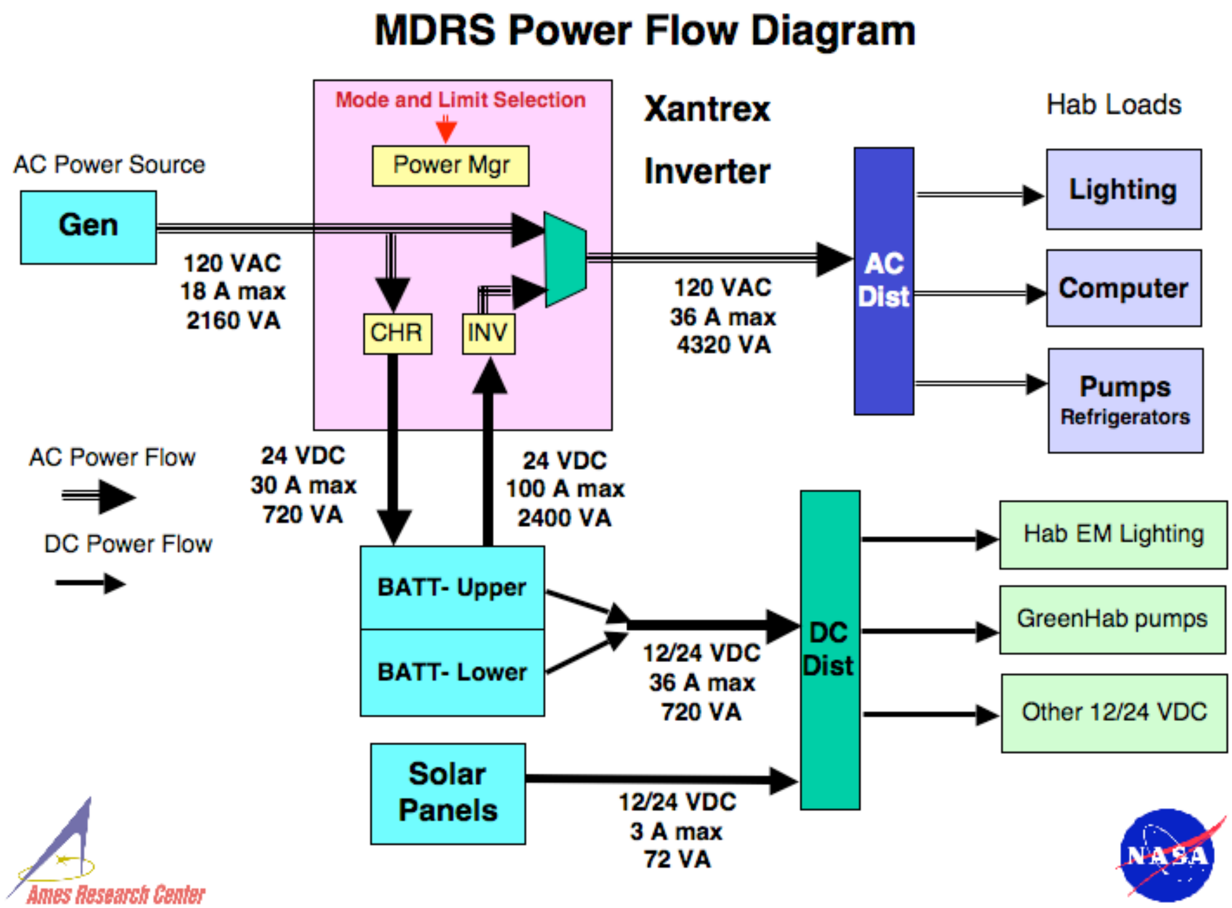

Figure 2. MDRS April 2006 Power Flow Diagram: Diesel or gasoline generator provides AC power to Inverter, which charges the batteries and automatically combines/converts and switches between alternative power sources. DC power distribution is supplemented by solar panels. (Graphic by Alena.)

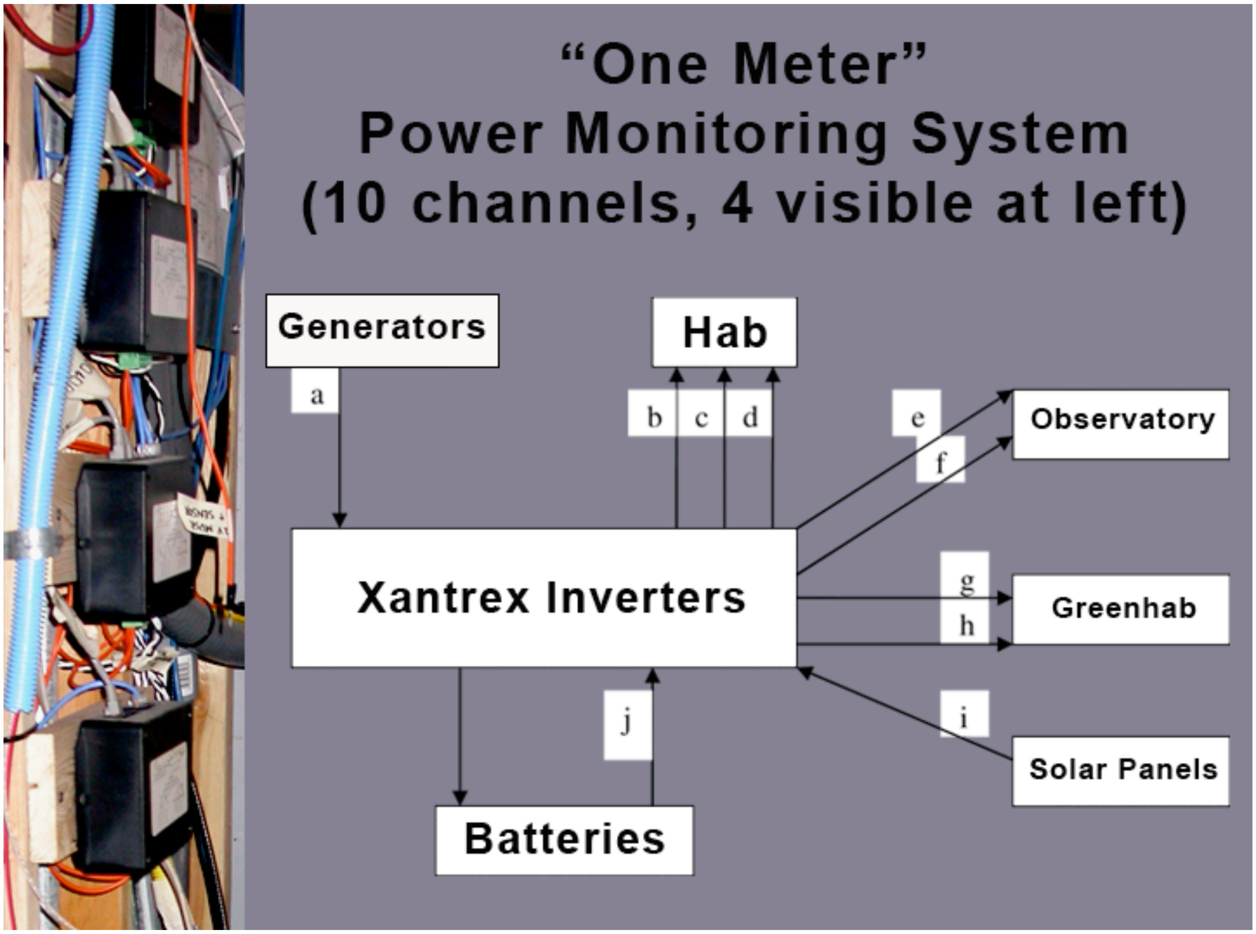

Figure 3. “One Meter” Power Monitoring System: Electrical connections providing voltage and amperage data are connected to ten “channels” (letters a through j), of which four are visible as black boxes on the left. These data are consolidated into a “one meter” box, which uses telenet to transmit data a set time intervals (15 minutes minimum allowed in April 2006) to a Mobile Agents computer (see Figure 4). (Graphic by Clancey and Covan.)

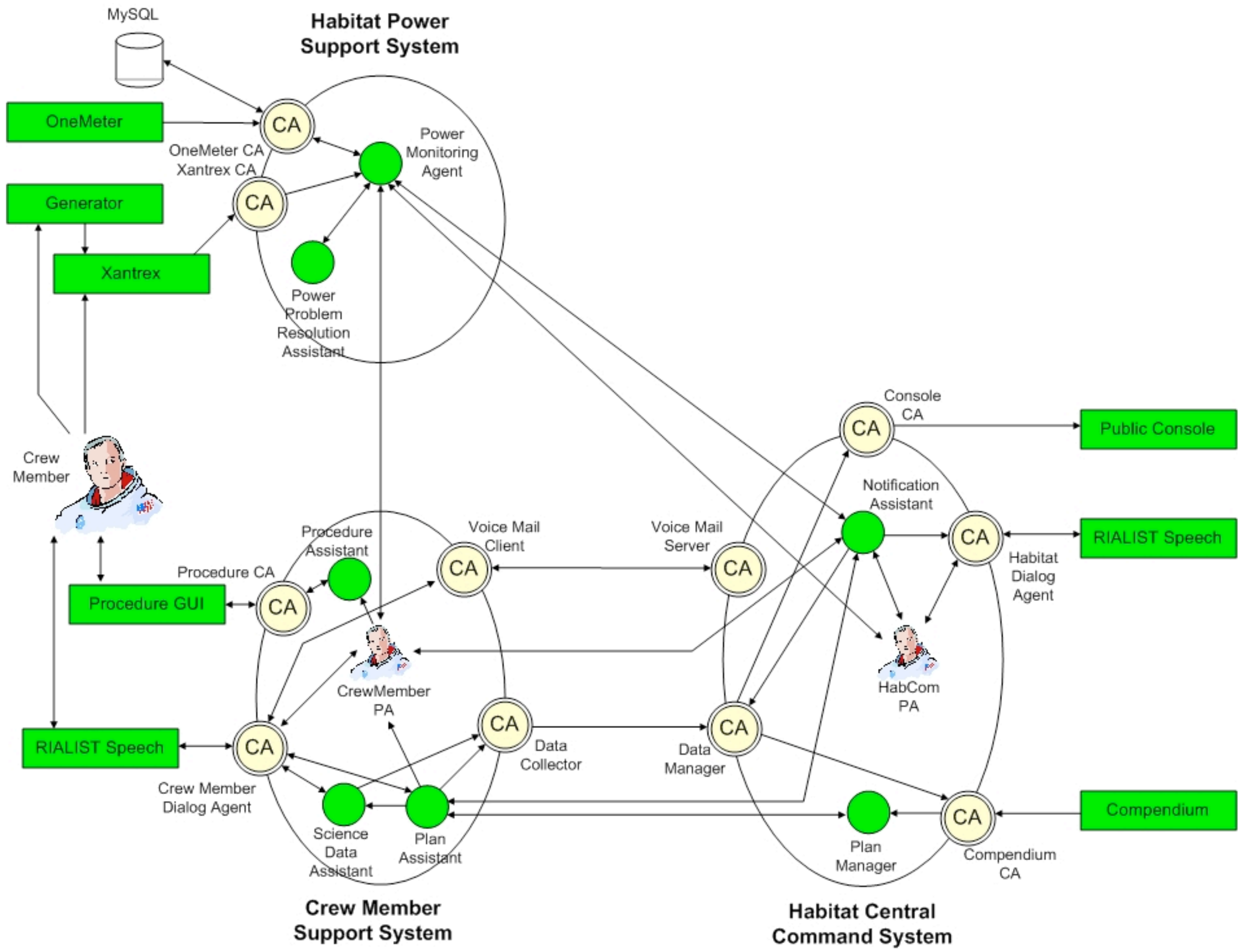

Figure 4. Mobile Agents MDRS49 Configuration uses six laptop computers: One for the power monitoring agent that connects to the One Meter telemetry ("Habitat Power Support System"), a central MDRS Command System ("HabCom") located on the upper deck, and one "Support System" laptop (located on the lower deck) for each of the four crew members who experimented with the system. The crew and HabCom systems include a "personal agent" for coordinating communications (command processing, alerting, and dataflow) with crew members. Specialized agents provide activity plan, science data, and troubleshooting procedure assistance. A plan manager in the HabCom system for initializing crew member plans. CA designates a "communication agent" which is implemented in Java and interacts with an applications programming interface (API) of some hardware device (e.g., OneMeter) or software system (e.g.,

MySQL database, dialog system). The link to the Xantrex inverter provided setup information (e.g., error thresholds); real-time data came only from the OneMeter system. (Graphic by van Hoof).