Demonstration: Liaison Agents for Distributed Space Operations

D. Schreckenghost, P. Bonasso, D. Kortenkamp, C. Martin, T. Milam, C. Thronesbery

NASA Johnson Space Center
Texas Robotics & Automation Laboratories (TRACLabs)
1012 Hercules, Houston, TX
ghost@ieee.org

1 Research Overview

Future manned space operations are expected to include a greater use of automation [Cooke and Hine, 2002] This automation will function without human intervention most of the time. However, humans will be required to supervise the automation, and they must be on-call to respond to anomalies or to perform related tasks that are not easily automated. In such an environment, humans perform other tasks most of the time, and their interaction with the automation may be remote and asynchronous. As automation becomes more prevalent, better support for such interaction is needed. The Distributed Collaboration and Interaction (DCI) environment, being developed at NASA, investigates the use of software agents to assist humans in this type of remote, distributed space operations.

The DCI approach has been applied for use by control engineers at the Johnson Space Center (JSC) who are investigating advanced technology for life support such as the water recovery system, or WRS [Schreckenghost, et al, 2002]. The WRS recycles wastewater through biological and chemical processes to remove impurities and produce potable water. Managed by an autonomous control program called 3T [Bonasso, et al, 1997], the WRS ran unattended in a continuous 24/7 integrated test from January 2001 through April 2002 [Bonasso, et al., 2002]. WRS control engineers periodically monitored for network, hardware, or power failures from remote locations, while spending the majority of their time carrying out their daily tasks on unrelated projects. The current prototype of the DCI environment uses a simulation of the WRS 3T system for both demonstration and continuing development. The DCI implementation creates an environment in which humans and the 3T control automation together form an integrated team to ensure efficient, effective operation of the WRS.

2 Approach: DCI Architecture

The DCI approach uses intermediate *liaison agents* called Attentive Remote Interaction and Execution Liaison (ARIEL) agents to provide an interfacing layer between the human and the control automation. In the DCI system, these liaison agents provide a variety of services, which together support (1) human supervision of automated control systems, (2) direct human control of processes such as crew life support, (3) activity tracking and coordination among humans and automated systems interacting with the same process, and (4) asynchronous information exchange among distributed, remote humans and automated systems.

In addition to liaison agents, DCI provides *augmenting software'.* This software includes new capabilities used by all ARIEL agents (e.g., software to detect events from a control agent) or DCI tools used to integrate existing software into the DCI environment (e.g., an interface to a centralized planner). In Figure 1, the Event Detection Assistant (EDA) and the Conversion Assistant for Planning (CAP) are representative pieces of augmenting software. The EDA monitors data produced by the control automation for data patterns that are of interest to humans supervising this system. The CAP augments a centralized hierarchical task net (HTN) planner, (the prototype uses AP [Elsaesser and Sanborn, 1990]), for interfacing to humans and their ARIEL agents. The entities with black backgrounds in Figure 1 (the human, the WRS system and its control software, and the centralized planner) participate in but are not part of the DCI environment.

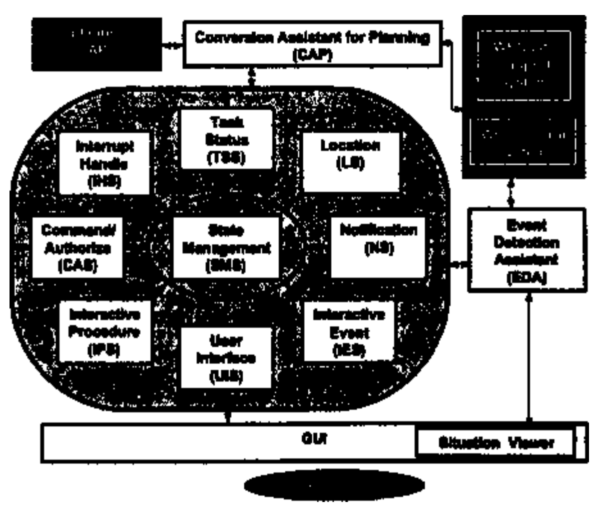

Figure 1. DCI Architecture

ARIEL agents, as pictured in Figure 1, provide the following services for and on-behalf-of their human users:

- the State Management Service maintains a model of the user's current context, including task, role, and location;
- the User Interface Service manages different modalities, such as display, pager, or email, to present information;
- the Notification Service extends communication protocols to support distributed, assynchronous collaboration and to ensure the proper routing of information to users based on their roles, location, etc.;
- the Task Status Service tracks human activity and provides completion status to the automated planner; and
- the Location Service provides human location information for use in tracking the completion status of user activities, determining how to notify the user of events, and customizing the presentation of information.

Three services included in the DCI design have not yet implemented:

- the Command and Authorization Service will assist a human in commanding the automation by reconfiguring the automation when transitioning between manual and automated commanding and by detecting and resolving potential command conflicts among distributed users;
- the Interactive Procedure Service will allow a user to specify new operations by modifying an automated procedure and triggering the control automation to perform this procedure; and
- the Interactive Event Service allows a user to interactively define temporary or new operational events and to control automated monitoring for these events.

ARIEL services work together to support an individual human's interaction with automation software and with other humans that are geographically and/or electronically distributed.

3 Demonstration

The DCI demonstration shows the potential benefits of the DCI system and ARIEL agents in the following typical scenario involving the WRS and its three control engineers: the "Prime" engineer, who is the first called in for WRS problems, a "Backup" engineer, and a "Coordinator," who oversees the work of the other two.

1) A loss of controls communication in the WRS control software requires a human to reinitialize the software.

2) The Prime engineer's ARIEL notifies him about the problem and the assignment of a new WRS repair task.

3) The Prime engineer is offline and doesn't respond to the pager notification in a timely manner.

4) The Prime engineer's ARIEL re-issues a task assignment acknowledgement request with increased urgency.

5) The Prime engineer still does not respond, and his ARIEL indicates to the planner that the task assignment has not been acknowledged. The planner reassigns the repair task to the Backup who is located off-site from JSC.

6) The Backup engineer's ARIEL notifies him about the assignment of a new WRS repair task.

7) The Backup engineer is online and responds to the request to acknowledge the change In his schedule.

8) The Backup engineer travels on-site to JSC, to fix the problem in the WRS.

9) In the Water Processing Facility, the Backup engineer reviews a summary of the anomaly situation via a notice previously logged by his ARIEL. Based on that review, he determines how to respond.

10) Once the problem is fixed, the Backup engineer's ARIEL notifies him when the water system has returned to normal and he leaves the Lab.

11) When the Prime logs into the DCI environment later, he reviews his ARIEL'S notifications about how the control team coordinated to resolve the situation and what if any impacts the problem had on his schedule.

12) Throughout this interchange, the Coordinator relies on his ARIEL to inform him of events in the WRS and the response of the control engineers on his team.

The DCI demonstration showcases the potential to provide unprecedented support for distributed collaboration among humans and automation in the context of future manned space operations. The DCI approach also holds future promise for extended application because similar increases in the use of automation can be seen in other domains ranging from process control to smart houses.

Acknowledgements

We want to acknowledge the support of Dr. Michael Shafto, the manager of Human-Centered Computing in NASA's Intelligent Systems Program, under which this work was done.

References

[Bonasso, et al., 1997] Bonasso, R. P., Firby, J. R., Gat, E.. Kortenkamp, D., Miller, D. P., and Slack, M. G. 1997. Experiences with an Architecture for Intelligent, Reactive Agents. *Journal of Experimental and Theoretical Artificial Intelligence* 9: 237-256.

[Bonasso, et al., 2002] Bonasso, R. P., Kortenkamp, D., and Thronesbery, C. 2002. Intelligent Control of A Water Recovery System: Three years In the Trenches. *AI Magazine* 24 (1).

[Cooke and Hine, 2002] Cooke, D. and Hine, B. 2002. NASA's New Era in Space Exploration. *IEEE Intelligent Systems* 17 (2).

[Elsaesser and Sanborn, 1990] Elsaesser, C. and Sanborn, J. 1990. An Architecture for Adversarial Planning. *IEEE Transactions on Systems, Man, and Cybernetics* 20(1): 186-194.

[Schreckenghost, et al., 2002] Schreckenghost, D., Thronesbery, C, Bonasso, R., Kortenkamp, D., and Martin, C. 2002. Intelligent Control of Life Support for Space Missions. *IEEE Intelligent Systems* 17 (5): 24-31.