pared to a manual process, this script reduces human error and saves considerable man-hours by automating and streamlining the mission planning and sequencing task for the GRAIL mission.

Real-Time Multimission Event Notification System for Mars Relay

NASA's Jet Propulsion Laboratory, Pasadena, California

As the Mars Relay Network is in constant flux (missions and teams going through their daily workflow), it is imperative that users are aware of such state changes. For example, a change by an orbiter team can affect operations on a lander team. This software provides an ambient view of the real-time status of the Mars network.

The Mars Relay Operations Service (MaROS) comprises a number of tools to coordinate, plan, and visualize various aspects of the Mars Relay Network. As part of MaROS, a feature set was developed that operates on several levels of the software architecture. These levels include a Web-based user interface, a back-end "ReSTlet" built in Java, and databases that store the data as it is received from the network. The result is a real-time event notification and management system, so mission teams can track and act upon events on a moment-by-moment basis.

This software retrieves events from MaROS and displays them to the end user. Updates happen in real time, i.e., messages are pushed to the user while logged into the system, and queued when the user is not online for later viewing. The software does not do away with the e-mail notifications, but augments them with in-line notifications. Further, this software expands the events that can generate a notification, and allows user-generated notifications.

Existing software sends a smaller subset of mission-generated notifications via email. A common complaint of users was that the system-generated e-mails often "get lost" with other e-mail that comes in. This software allows for an expanded set (including user-generated) of notifications displayed in-line of the program. By separating notifications, this can improve a user's workflow.

SIM_EXPLORE: Software for Directed Exploration of Complex Systems

NASA's Jet Propulsion Laboratory, Pasadena, California

Physics-based numerical simulation codes are widely used in science and engineering to model complex systems that would be infeasible to study otherwise. While such codes may provide the highest-fidelity representation of system behavior, they are often so slow to run that insight into the system is limited. Trying to understand the effects of inputs on outputs by conducting an exhaustive grid-based sweep over the input parameter space is simply too time-consuming. An alternative approach called "directed exploration" (see figure) has been developed to harvest information from numerical simulators more efficiently. The basic idea is to employ active learning and supervised machine learning to choose cleverly at each step which simulation trials to run next based on the results of previous trials.

SIM_EXPLORE is a new computer program that uses directed exploration to explore efficiently complex systems
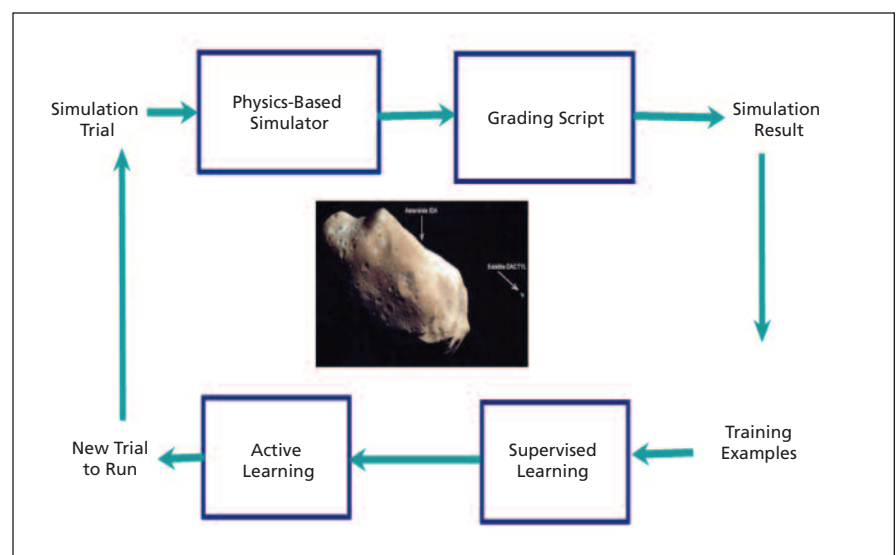

Illustration of the **Directed Exploration** approach in an asteroid collision application. The central image shows the Ida-Dactyl asteroid pair observed serendipitously by the Galileo spacecraft. Planetary scientists are interested in understanding how such systems form and more generally in how asteroid families form. Physics-based numerical simulations offer a means to gain insight into such systems; however, the simulations are so slow to run that a *directed exploration* strategy is required.