

The MER/CIP Portal for Ground Operations

Louise Chan^{*}
Sanjay Desai^{*}
Matthew D'Ortenzio[†]
Robert E. Filman[‡]
Dennis M. Heher^{*}
Kim Hubbard[†]
Sandra Johan[†]
Leslie Keely[†]
Vish Magapu^{*}
Ronald Mak[‡]
Quit Nguyen^{*}
Tarang K. Patel^{*}
John A. Schreiner[†]
Elias Sinderson[⊕]
Joan D. Walton[†]

^{*} SAIC
[†] NASA
[‡] RIACS
[⊕] CSS

NASA Ames Research Center
Moffett Field, CA 94035

Contact author:

Robert E. Filman,
NASA Ames Research Center M/S 269-2,
Moffett Field, CA 94035.
1+650-604-1250
rfilman@mail.arc.nasa.gov

Focus issues: Operations and sustainment concepts

The Mars Exploration Rover

In June and July, 2003, NASA launched two robotic vehicles to explore Mars. These Mars Exploration Rovers (MER) are scheduled to land on opposite sides of the planet in January 2004. The primary goal of the MER mission is to research the history of water on Mars. After a parachute-, retrorocket- and airbag-eased landing, the rovers will roll over the Martian landscape. Panoramic camera images radioed to scientists on Earth will present possible exploration targets; commands radioed back will direct the rovers to move to particular sites and use their instruments (panoramic camera, miniature thermal emission spectrometer, Mössbauer spectrometer, alpha particle X-ray spectrometer, magnets, microscopic imager, and rock abrasion tool) to gather data. The rovers are planned to drive up to 40 meters a day; the planned duration of the mission is three months.

Back on earth, a small army of about 250 scientists and engineers, both at JPL and other sites, will work around the clock analyzing the collected data, determining activities for the next day and carefully composing the command sequences to realize these goals. The scientists and engineers must work closely together to balance scientific objectives with engineering constraints—the goal of the project is science, but science requires functioning rovers.

Mars is an unpredictable environment. We issue commands to the Rovers. However, we don't know if the commands will execute successfully, nor which of the objects sensed by the Rovers is worthy of further examination. The steps of what, to a scientist, are conceptually individual experiments may be intermixed and scattered over a large number of activities. While the scientific staff has an overall strategic idea of what it would like to accomplish, concrete activities are planned daily. The data and surprises of the previous day need to be integrated into the negotiations for the next day's activities, all synchronized to a schedule of transmission windows. "Negotiate" is the operative term, as different scientists want the same resources to run possibly incompatible experiments. Many meetings plan each day's activities. There is a lot going on: situational awareness is critical to the participants, communication windows drive hard deadlines, and everything is synchronized to the Martian day.

Collaborative Information Portal

We developed the Mars Exploration Rover/Collaborative Information Portal (MER/CIP) to facilitate MER operations. MER/CIP provides a centralized, one-stop delivery platform integrating science and engineering data from several distributed heterogeneous data sources. Key issues for MER/CIP include

- **Scheduling and schedule reminders.** Meetings drive operations planning. Participants need dynamic information about where they must be and cross-correlation to activities on Mars. Rather shortsightedly, COTS calendar tools presume a 24 hour day; A Martian Sol is 24.66 hours. Scheduling with respect to Mars time is critical, for the Rover is powered by sunlight.
- **Tracking the status of daily predicted outputs.** The outputs of experiments are transmitted to earth daily. Experimenters want to know what is planned to be done on a given Sol and what actually happened. They need to be informed when their data products have arrived.
- **Finding and analyzing data products.** This includes searching through the data-products space and analyzing the data files found there. The data is stored in existing heterogeneous structures, developed independently and obliviously to the needs of the portal.
- **Collaboration.** Users want to share information including not only data products, but also analyses and annotations.
- **Announcements.** MER/CIP serves as a primary mechanism for broadcasting to the staff announcements of events such as changes in schedules of meetings.
- **Personalization.** User interfaces, data product awareness and access rights to data must be personalized to the preferences and rights of each user.

From the user's point of view, MER/CIP is an integrated application composed of a collection of tabs/tools. Figure 1 presents the user interface graphics of several of these tools, including

- **Clocks** to enable the user to view any time zone, including Mars time zones. Clocks displaying Mars times (LST-A and LST-B in the figure) tick in Martian minutes, which are each nearly 2 seconds longer than an Earth minute.
- **Data navigator** for browsing, searching and previewing science data.
- **Reports navigator** for accessing and previewing mission reports.
- **Schedule viewer** for displaying operations and staffing schedules in various time scales, including Martian scales. Separate utilities (not shown) allow users who have the appropriate privileges to create and update the schedules.
- **Event horizon** for tracking important meetings and other events. Each event displays a countdown of the minutes to the start of the event.

- **Broadcast announcements** for distributing important messages to mission personnel.
- **Time converter** for converting between time zones (both Earth- and Mars-based).

MER/CIP Architecture

Figure 2 shows the MER/CIP architecture. MER/CIP is a three-tiered architecture

- **The Client.** The client tier presents the application to the user. It is a Java/Swing application and runs on Windows, MacOS X, Linux and Solaris. The client is responsible for transforming the clicks and keystrokes of users into web services requests to the middleware, and formatting the responses into visualizations such as the windows of Figure 1. Since the overall architecture is based on a “pull” model of web services, each client has a component that periodically polls to check for asynchronous messages.
- **The Middleware.** The middleware runs on a WebLogic, J2EE, Enterprise JavaBeans (EJB) server. It provides XML/SOAP-based web services to the client, including various services for accessing the back-end database, streaming data, static web pages, servlets for dynamic web pages, verifying user credentials, providing current Earth and Mars times, generating announcements, and handling JMS-conforming, publish-and-subscribe asynchronous messaging.

- **The Data Tier.** The data tier includes not only databases, but also access to the overall mission file system. In general, science data from the Rovers is asynchronously and unpredictably added to the mission file system. A MER/CIP daemon periodically scans the interesting parts of that system, and, on discovering new data files, adds information about them to the “meta” database and informs the middleware of their existence. The middleware uses this information to inform clients that have subscribed to files like those of their arrival.

Lessons Learned

The process of developing MER/CIP illustrates a variety of lessons for developing operational systems, particularly one-of-a-kind operational systems for and by a research environment.

We first note that, on the verge of operational deployment, MER/CIP seems to be a success. It has progressed from being viewed as a minor frill for the operating environment to being a central, critical resource. It appears to satisfy an otherwise unfulfilled need for knowledge of what is happening in the operating environment—current information on meetings and other scheduled events, data arrivals, and even answering what is ordinarily a simple question, “What time is it?” with extraterrestrial precision. MER/CIP has also been an implementation success, being one of the first examples of a full-fledged, successfully operational, three-

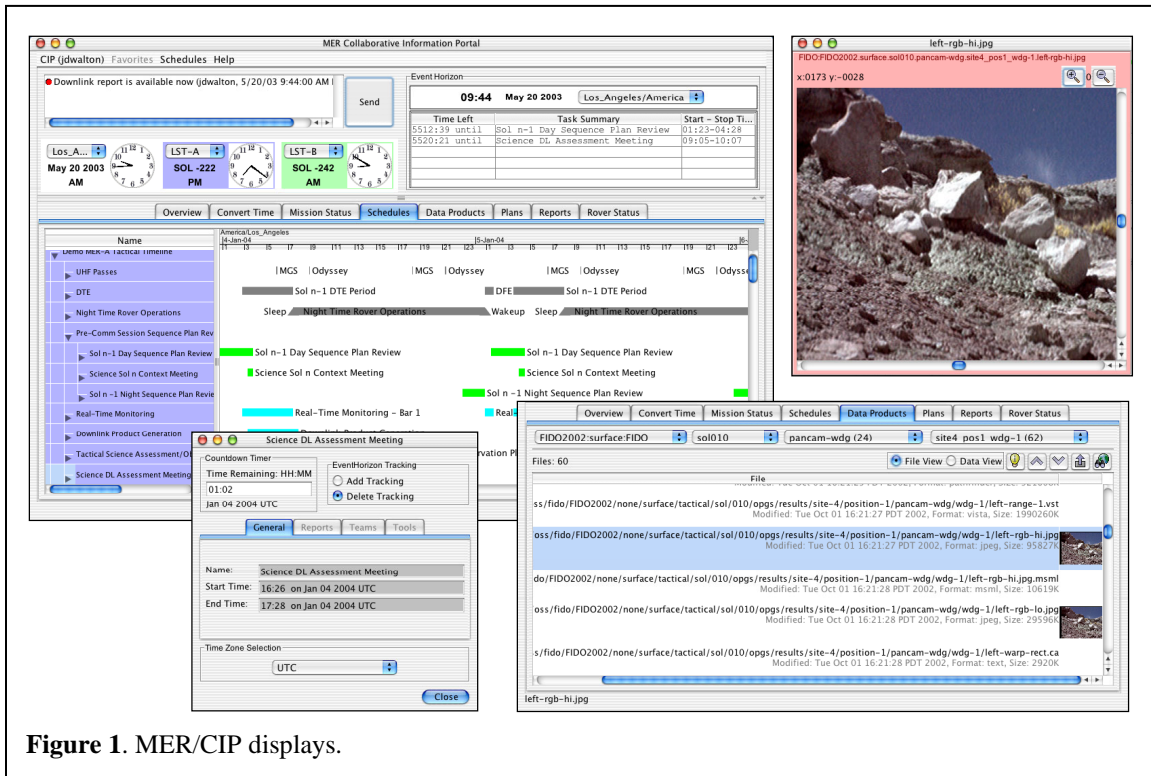


Figure 1. MER/CIP displays.

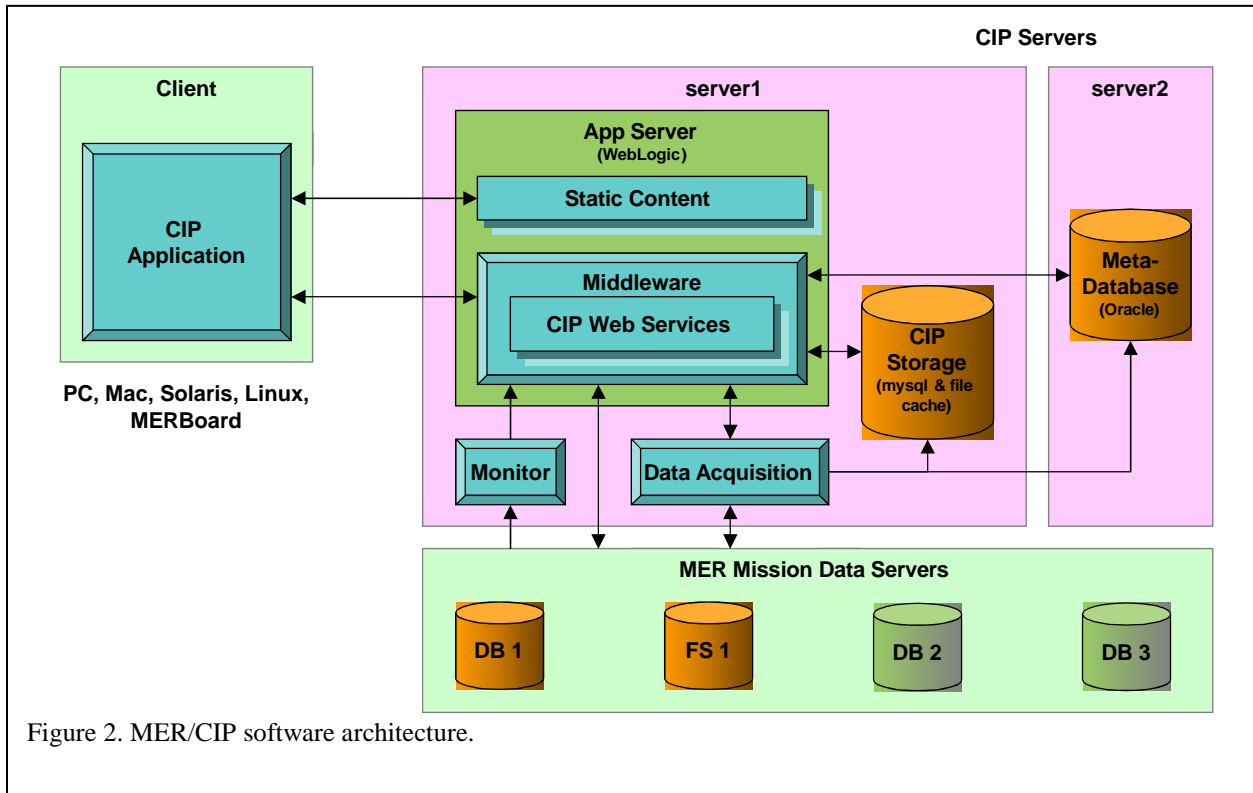


Figure 2. MER/CIP software architecture.

tier Enterprise Java Beans environment.

MER/CIP was developed in a research environment, which translates to customers who are unable to articulate at the beginning of the project what they will need in the end, and an underlying operating environment that has evolved during system development. Several lessons are implied by this situation, primarily the need for developing flexible systems and being able to adjust one's engineering assumptions. A primary mechanism for flexibility is data-driven software, where the task to be performed is described in some language in the system configuration data, rather than being hard-coded in the software. Thus, for example, the locations of interest for the data loader are described as configuration data (including positive and negative filters) rather than being encoded as source. Similarly, engineering assumptions such as the different frequency of operations can drive performance—the kind of caching strategy to be used in an environment with few and infrequent data changes is different than one with common changes; a system that can adjust to the actual operations will have advantages over one that cannot.

MER/CIP economized by integrating several commercial, off-the-shelf software products. This is a double-edged sword—one trades development on one's own part for a likely higher-quality component, though that component comes with a steep learning curve and inflexible behavior.

Research software is often “demo once and throw away.” For a tool to be relied on by others, it is critical

to include quality assurance as part of the development process; for a tool to be used by others, it is critical to include training and documentation in that process. Bringing in team members with these specialized skills early into the development process is a good idea.

A system that presents a complex user interface needs time for that interface to be tested and evolved. A particularly critical item we discovered was the “computer as oracle” fallacy—disclaimers about accuracy and precision are quickly forgotten in front of a computer screen. For example, our original mechanism for presenting Martian time, accurate to five minutes, needed dramatic revision when (despite such disclaimers) customers complained about its impression.

At its heart, MER/CIP is a tool about describing and finding data. A lesson for the developers of missions is that it is not good enough just to get the data back from space; one must also plan how one is going to organize it back on Earth. If there is the possibility of having one's data read or manipulated by inappropriate, malicious or careless individuals, then the system needs security, and the security policy ought to be thought out at the beginning of the process so that they can be uniformly realized in all further development.