Collaborative Editing for Mission Control

Robert C. Steinke
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109 USA
(818) 354-5506
robert.steinke@jpl.nasa.gov

ABSTRACT

Mission control of robotic space exploration (e.g. a Mars rover) involves several collaborative editing tasks such as generating command sequences to send to the robot. We are currently developing tools to allow remote participation in mission control activities over the internet. In this application, security and reliability requirements are paramount. We are interested in advanced techniques for data consistency in collaborative editing which include formal verification if possible. Our current work is based on shared memory consistency models since they have both formal and intuitive descriptions.

Keywords

collaborative editing, shared memory consistency models

POSITION STATEMENT

Mission control for robotic space exploration is an application traditionally performed in a centralized fashion. All participants are physically present at the mission control facility. This centralized approach has disadvantages as many participants have to relocate and there is a large facilities burden at the mission control site. The Web Interface for TeleScience (WITS) is a tool developed at JPL that enables remote participation in mission control activities over the internet.

There are several mission control activities that can be accurately described as collaborative editing tasks. The primary goal of mission control is to produce command sequences that are uplinked to the robot. Many different people are involved in selecting science targets, selecting activities to perform on those targets, and optimizing a command sequence to maximize science return while staying within engineering restrictions. All of those people's input must be incorporated to produce a single artifact, the command sequence. Other collaborative editing tasks include analyzing and processing science data to produce data that is improved or enhanced in some way.

The field of space exploration has extreme requirements for mission safety as each mission is very expensive and there is very little repair ability if something goes wrong after launch. Performing mission operations over the internet raises serious questions of security and reliability. The issue we believe this workshop can best address is the consistency of data in collaborative editing. For example, the mission manager might have to okay a sequence before it is uplinked. It would be a disaster if the version of the sequence okayed by the mission manager was not the same version uplinked.

Our current strategy for data consistency is based on shared memory consistency models. Consistency models can be expressed in terms of each process' view of the order of events in the system. This expression has a solid formal foundation and an easy to understand informal description. A good example to demonstrate this is a chat session.

A chat session can be modeled as a single variable of type string where each participant corresponds to a process writing to the end of the string. Sequential consistency requires that all writes appear to occur in the same order for every process. In our example, this means all participants see the exact same conversation. With weaker consistency models different participants may see the same statements in different orders, but each consistency model would still enforce certain desired properties. For example in causal consistency, an answer to a question must appear after the question it is answering.

Memory consistency models have some aspects that make them distinctly different from database serializability. Consistency models do not group operations into transactions, and there is no concept of aborting an operation after it is submitted. This may mesh better with collaborative editing. Also, there are several consistency models to choose from, and they can be formally related. It may be desirable to use different consistency for different participants, at different times, or for different types of interaction.

There is one aspect in which memory consistency and serializability are similar. Both expect a process to actively submit read operations. In a collaborative editing session, a participant may have information displayed on a screen which can be passively read. A memory consistency model based approach must assume that a participant is continually submitting read operations for anything displayed in such a fashion.

In conclusion, we are interested in data consistency techniques for collaborative editing suitable for distributed mission operations which has strong reliability requirements. Ideally, any such technique would be formally verified. We believe shared memory consistency models are a good starting point when looking for such techniques.