Using Transcription and Replay in Analysis of Collaborative Applications

Seth Landsman, Ph.D. – The MITRE Corporation

Richard Alterman, Ph.D. – Brandeis University

MITRE

Overview

- Motivation and Problem Statement
- Analysis of Collaboration
- Software Model for Transcription and Replay
  - Instrumentation
  - Generation
- Conclusions and Ongoing Work

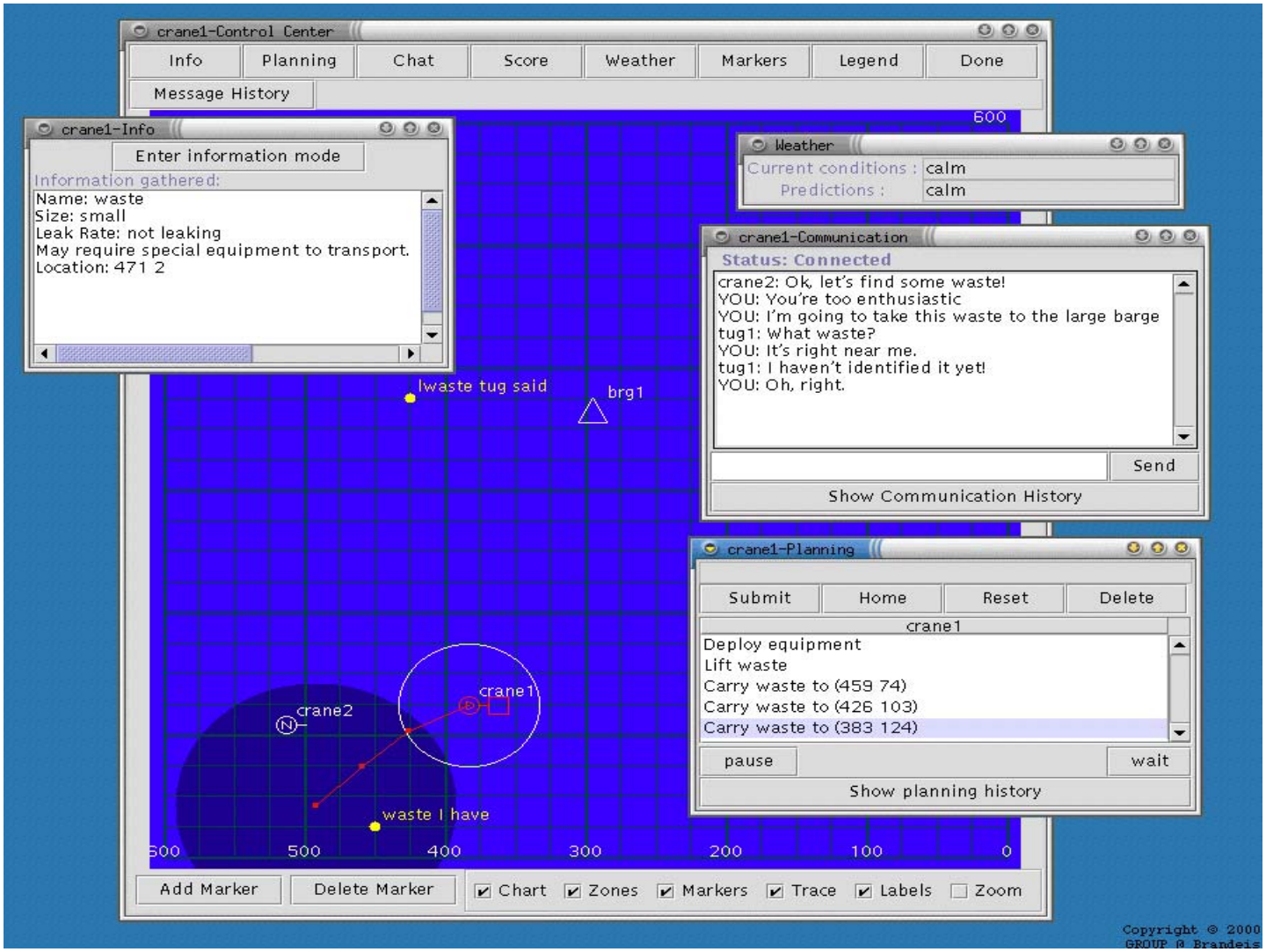
Motivation

- Successful execution of a C2 operation is increasingly a distributed enterprise mediated by a software system
- Success of an operation, therefore, depends on
  - Success of the underlying software and task environment
  - Success of the collaboration mediated by the software system
- How can we increase the likelihood of success in a collaborative environment?
  - How can we ensure that collaboration is tailored to the task environment and user community?
- Being able to understand how the user community uses the C2 system may be able to aid in the engineering of more successful systems
- Our approach is the in system replay of collaborative activity to support ethnographic analysis of the users' actions

VesselWorld

- The VesselWorld system is a simple, synchronous collaborative application used to study challenges in software mediated collaboration
  - Three users work to solve a cooperative problem
    - Each user has a different role in the problem solving
      - Each role has different capabilities
      - Explicit coordination of activity is required to complete the task
- To understand the collaboration as mediated via the application, ethnographic analysis was performed
  - Analysis indicated that users structured their communication over domain objects and planning
  - Enhancements were added that provided tracking of domain objects, and short term and long term planning
- Long development cycle followed by imprecise analysis
  - Experimenter notes, observations of collected data

VesselWorld



Lessons Learned from VesselWorld

- Adding new capabilities to VesselWorld was expensive:
  - Time consuming to build
  - Hard to enhance once built
- How can we be more precise in the enhancements or changes made to a collaborative application?
  - Shorten the feedback loop between implementation and analysis
- How can we be more precise in how collaborative improvement and issues are observed
  - Improve the tools used to study collaborative activities
  - Improve the analysis methods we have available, as supported by the tools
- As the complexity of collaborative applications increases, the need for techniques to construct applications that are appropriate for the task and user community become more critical

Analysis of Collaboration

- Existing techniques help in the construction of collaborative application
  - Rapid development techniques (Roseman and Greenberg, 1992, Pedersen, et al, 1993, Li, et al, 1999)
  - Ethnography (Suchman and Trigg, 1991, Neal and Simons, 1983)
  - Analysis Techniques (Feinman and Alterman, 2003, Feinman, 2006, Lárusson, 2010)
- Each of these techniques provide a piece of the puzzle
  - How can we engineer applications quickly, figure out what information to collect, and do something with the information once it is distilled
- However, the fundamental question of how to collect and work with the user activity is unanswered

Software Model of Transcription and Replay

- Our approach is what we call a “within system perspective” of user activity
  - Compared to having a video camera focused on a user’s screen
  - Over the shoulder view of the user’s activities
- The user activity from the perspective of system events, not UI events, is captured and transcribed
  - Capture chat utterances or planning activity, not key presses and mouse clicks
- The result is that the user activity can be replayed from an individual user perspective or an omniscient perspective
- Our model is implemented into two frameworks
  - THYME is the collaboration construct toolkit that generates the transcripts
  - SAGE is the set of replay components that are applied to a THYME application

Transcription

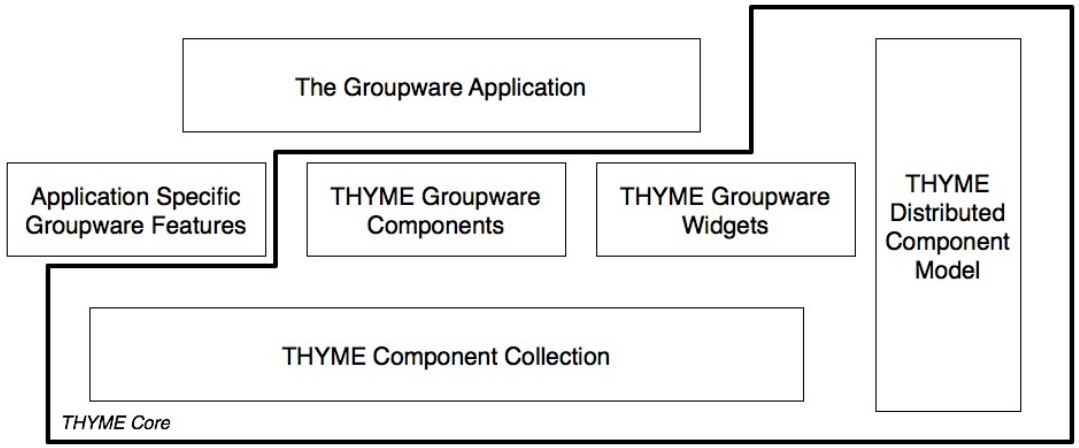
- Collection of interaction between the application and the users
- Features of the transcription capability influence the replay capabilities
  - Completeness
    - Both the amount of information and details
  - Types of information collected
    - Mouse events, chat events, etc
  - Transitions versus States
    - Each atomic unit in the transcript is the system state or an event
- Customized transcription gives most fidelity of information, but is expensive to implement on a per-application basis
  - Internal transcription is next best (e.g., Morse and Steves, 2000)
  - External transcription lacks information context (e.g., Suchman and Trigg, 1991)

Replay

- Allows ethnographic analysis of groupware application use
  - Online behavior can be captured and recreated exactly through a transcript
- Basis replay capabilities are similar to playing a video tape
  - Features enhance the analysis
    - Precision
    - Search
    - Annotation
- How can transcription and replay be accomplished without significant impact to deployment schedule?
  - Leverage system infrastructure
  - Make replay cheap

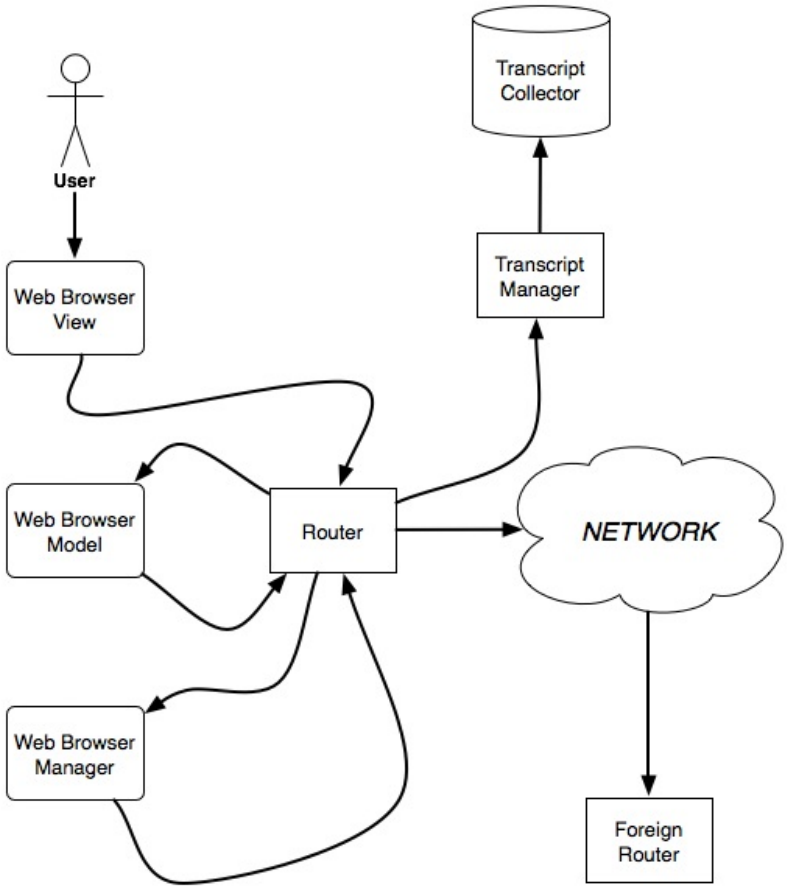
Frameworks

- THYME
  - Framework for building component-oriented groupware applications
    - Includes transcription capabilities
    - Model of development encourages localized changes
  - Rich library of groupware widgets and components
- SAGE
  - Class library for replaying THYME applications
    - Includes capability to generate replay applications from a THYME application



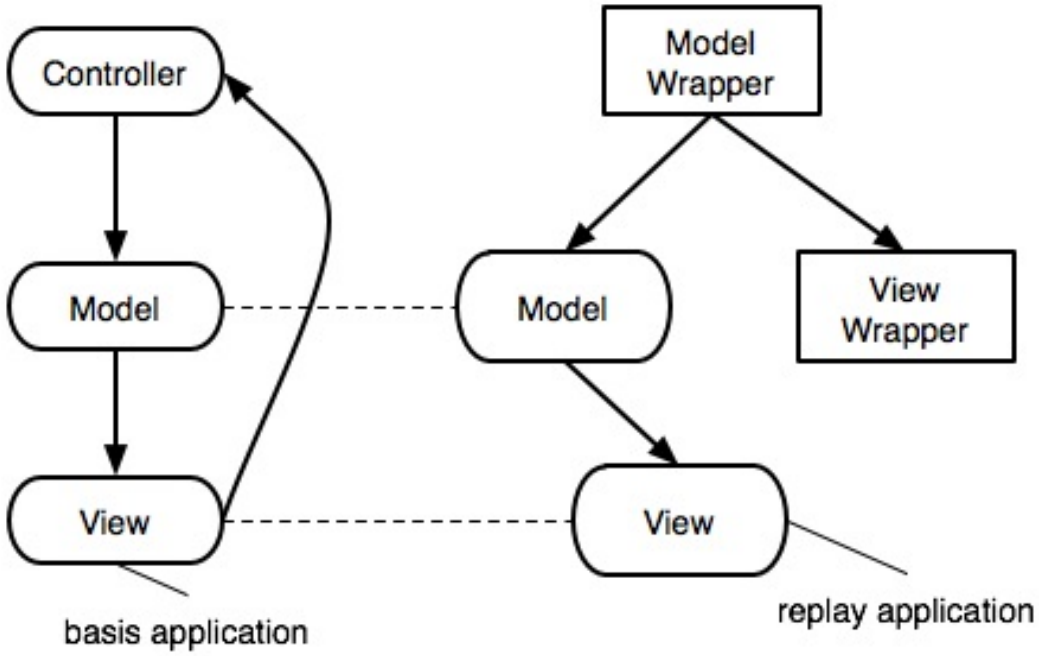
Instrumentation

- Interaction is collected into an ordered transcript of messages
  - Interaction between components
  - Interaction between the user and the system

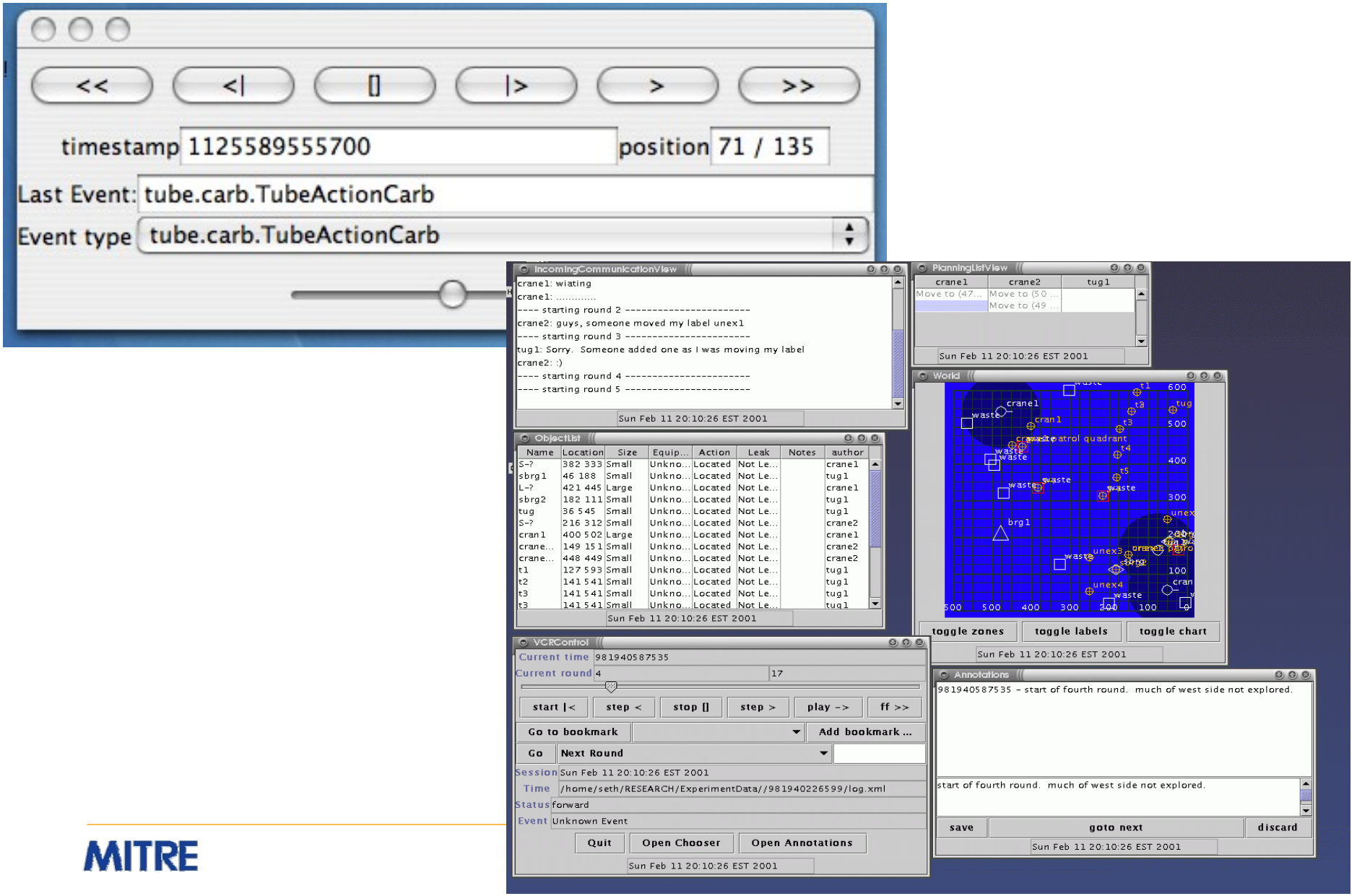


Generation

- Individual components from the basis THYME application are used in the SAGE application
  - Cheaper development
  - Ensures accuracy of the representation



SAGE for VesselWorld



MITRE

Ongoing Work

- There is demonstrated benefit to replay of collected usage data for improving collaborative activity
  - More examples in the paper
- However, doing so requires an investment
  - THYME and SAGE reduce that benefit, but it was still an upfront investment to build the frameworks
- Infrastructure has come a long way since we wrote THYME, specifically
  - More introspectable component architectures in J2EE, Microsoft Web Services, etc
  - More distributed architectures in ESBs and general messaging architectures
- How can we leverage these architectures to enable transcription and replay on more general systems?

Conclusions

- Analysis of collaborative applications is key need for building maintainable, adaptable, and usable applications
  - The application changes during its lifetime
    - Building the application is insufficient, it must be analyzed, modified, and redeployed
    - These activities must be factored into the engineering process
  - The proposed system shows how to accomplish the analysis task
- THYME and SAGE are example implementations of the software support necessary for this analysis
  - Automatic transcription of use
  - Generation of replay application
- This work is a first step on being able analyze and learn from a user community's behavior in situ