Toward the Design of Web-Based Planning and Scheduling Services

Stephen F. Smith, David W. Hildum, David Crimm

The Intelligent Coordination and Logistics Laboratory
The Robotics Institute
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh PA 15213 USA
{sfs/hildum/dcrimm}@cs.cmu.edu
fax: 412.268.5569

Abstract

This paper introduces COMIREM (Continuous, Mixed-Initiative Resource Management), a system for collaborative, incremental development of plans and schedules in dynamic, resource-constrained domains. COMIREM is designed to provide web-based planning and scheduling services and is capable of interacting with a variety of interfacing and visualization tools using a standard Internet browser. The objective is to deliver a wide range of mixed-initiative problem-solving capabilities (e.g., specification of activities and requirements, commitment/de-commitment of resources, manipulation of problem constraints) through lightweight plug-in applications that can be selected according to present circumstances. We present a summary of the current functionality and an outline of the developing system architecture.

Keywords: collaborative planning and scheduling, lightweight web-based planning and scheduling components, incremental mixed-initiative resource management

1. Introduction

The explosion of the Web in recent years has increased both the opportunity and the need for collaborative planning and scheduling tools. On one hand it has provided an infrastructure that offers unprecedented potential for putting planning and scheduling capabilities in the hands of mobile and physically distributed decision-making agents. Planning and scheduling tools and technologies can now be coupled with widely accessible browser interfaces and increasingly standardized data formats and application communication protocols. On the other hand, the emergence of this technological infrastructure has placed increased emphasis on dynamic, real-time tracking of execution status and on efficient, ongoing management of plans/schedules as circumstances evolve. Among the perceived keys to more effective supply chain management, for example, are greater visibility of the real-time status of pending orders and projected demand, the ability to generate promise dates as a function of actual shop status, and the ability to rapidly re-arrange production to capitalize on spot market demand. In the domain of Special Operations Forces (SOF) planning, alternatively, plans and schedules must be developed rapidly on the basis of available information and then continually refined and revised as new information is accumulated and execution circumstances necessitate change. To further complicate matters, planning and executing agents are typically mobile over time and sometimes operating with limited computational capabilities (e.g., PDAs).

This need for dynamic, real-time planning and scheduling capabilities is at direct odds with the capabilities of current planning and scheduling tools. Current tools tend to be batch-oriented, monolithic in their design and scope, and provide little provision for controlled incremental change. In some cases, the result is an inability to keep pace with execution, and ultimately a forced separation of the planning function from operational control. The Enterprise Resource Planning (ERP) systems of many manufacturing organizations exhibit this problem; their sluggishness typically relegates planning to an overnight process, and they are simply not designed for the sorts of real-time support demanded by E-business models. A second problem stemming directly from the monolithic design philosophy that underlies current tools is lack of software mobility. In domains such as SOF planning as well as in many E-business contexts, planning and scheduling services may be best provided in an on-demand fashion. Moreover, if computational resources are limited, it becomes necessary to provide only selected services (e.g., plan interrogation service without generation service, lower fidelity but computationally cheaper planning/scheduling models, etc.), and allow the possibility to dynamically reconfigure. In any context where planning and scheduling services are being delivered in an on-demand fashion, the design of lightweight components becomes essential.

In this paper, we summarize current research aimed at the design of collaborative, web-based planning and scheduling services. We describe a prototype system for resource management called COMIREM (Continuous, Mixed-Initiative Resource Management), developed to support incremental plan development and management in the SOF planning domain. COMIREM is designed to operate within a standard Internet browser environment; the user interface is realized from an assembly of dynamically generated HTML documents, Java applets and a Shockwave application. Core resource management services provided by COMIREM leverage an underlying model of planning and scheduling as an incremental change process, and hence provide direct support for incremental, mixed-initiative solution development and management. Following emerging standardization trends, data integration and communication between system components is achieved via XML-based formats and HTTP (HyperText Transfer Protocol). In the following we summarize current functional capabilities and the underlying system architecture.

2. COMIREM System Overview

As just indicated, COMIREM is a prototype system designed for interactive development, refinement and revision of plans in dynamic, resource-constrained domains and currently being applied in the domain of Special Operations Forces (SOF) planning. SOF planning is representative of a class of problems where (1) planning must proceed on short notice using incomplete information, with additional information coming in incrementally and a piecemeal fashion over time, (2) execution typically commences before planning is complete and the dynamic nature of the environment requires a tight coupling between execution and (re) planning processes, and (3) plans inevitably do not execute as expected; there is a constant need to be able to quickly identify critical constraints and decision tradeoffs, and for rapid evaluation of alternative recovery options. The problem is further complicated by the fact that planning is often taking place in a distributed fashion, by mobile decision-makers with limited computational resources and under executional duress.

2.1 Functional Summary

Toward the development of planning tools for this class of application, the current COMIREM prototype provides a basic set of core resource management capabilities:

- Interactive commitment/de-commitment – Given an activity or task network at some level of precision, a user can selectively commit or constrain the set of resources to be used for a given activity or activities (or vice-versa s/he can de-commit or broaden possible resource choices). The underlying resource model in COMIREM utilizes the notion of *capabilities* to flexibly define the mapping between various types of resources and the tasks they are equipped to perform. Capabilities are organized hierarchically to enable commitments to be made at various levels of specificity, perhaps as a function of the amount of information available at the time the decision needs to be made. In the SOF domain, for example, a given deployment task may require a `light-transport` capability, which may be supportable by a range of different aircraft types. If it becomes known that only helicopters are to be made available, then this task's set of resource alternatives can be narrowed to `rotor-aircraft` to better assess resource capacity constraints.
- Manipulation of problem constraints – The ability to interactively add, delete or adjust problem constraints is also provided. Temporal bounds (e.g., milestone constraints) may be imposed on specific activities, and ordering constraints may be posted between sets of activities. Likewise, resource quantities and capacities can be increased or decreased over specific intervals of the execution horizon.
- Generation of scheduling options – As specific commitments/de-commitments are made or as changes are made to the current problem constraints, time and resource constraints are propagated and sets of feasible scheduling options are updated. A Simple Temporal Network (STP) representation of the current plan is used to incrementally manage temporal bounds on individual activities and detect temporal constraint conflicts. Allocation decisions and changes in resource availability are propagated through currently unassigned activities to maintain sets of possible resource assignments.
- Undo – Any of the above actions can be taken and interleaved opportunistically and then subsequently undone if determined to be undesirable.

The above capabilities are provided in COMIREM through a web-based user interface. Figure 2.1 depicts the basic "plan-oriented" visualization that provides the backbone of the overall interface. Within this visualization, which has been adapted from other current SOF planning tools, the horizontal bars are used to indicate an activity (or more generally sequences of activities) taking place at a particular location over time, and diagonal bars are used to indicate movement (or transport) between locations. Bars (activities) are color coded to indicate such characteristics as activity type (e.g., actual transport activity versus resource positioning or de-positioning flights), scheduling status (assigned or unassigned), and decision criticality (a function of the number of feasible resource alternatives that remain). Purple pincushions are used to highlight specific milestone constraints that are currently associated with different activities.

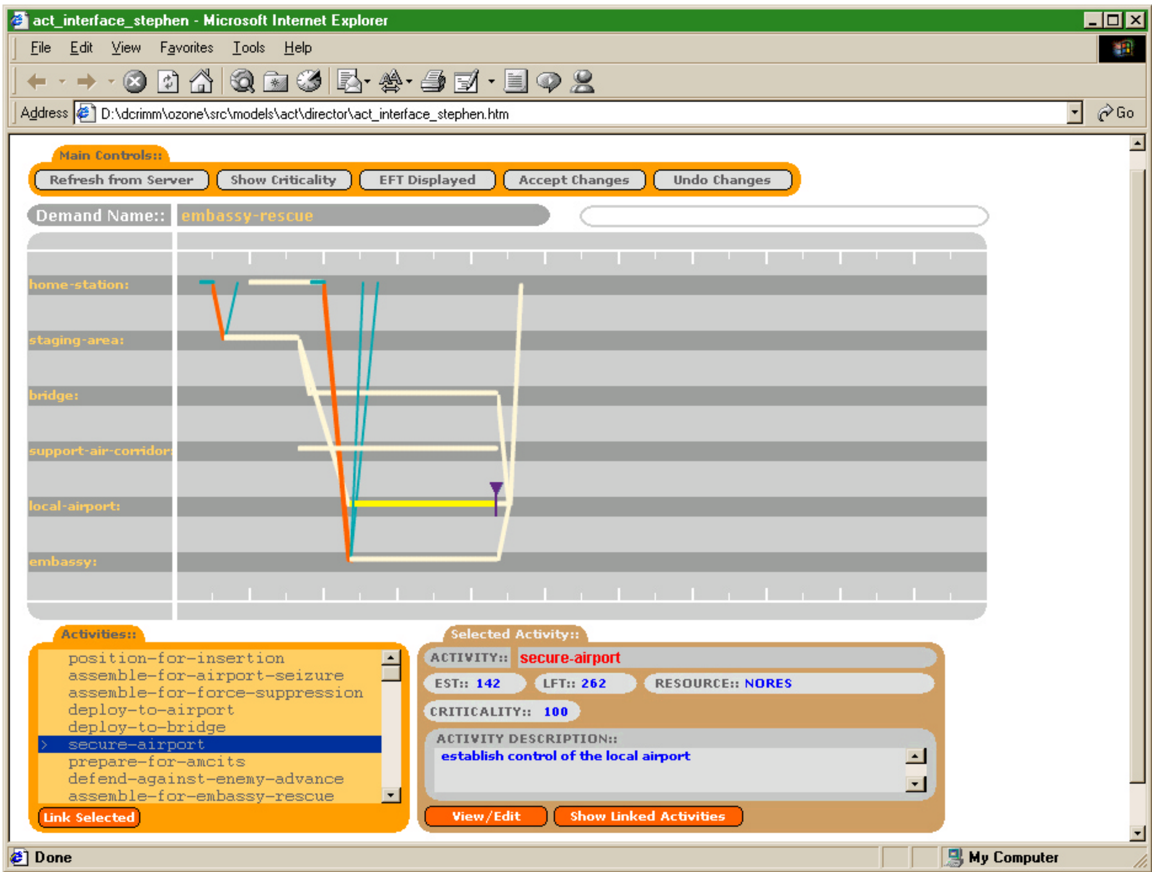

Figure 2.1 Plan Oriented View

Specific bars (activities) can be selected to access/edit their current properties and resource assignments. Figure 2.2 shows the current temporal bounds and feasible resource assignments for a particular activity in a plan. Figure 2.3 shows use of the basic plan visualization to characterize the impact of a particular constraint change on the evolving plan/schedule. A final interface component (not depicted here) gives a more conventional ‘Gantt’-style visualization of resource availability over time, which can also be used to manipulate current resource availability constraints.

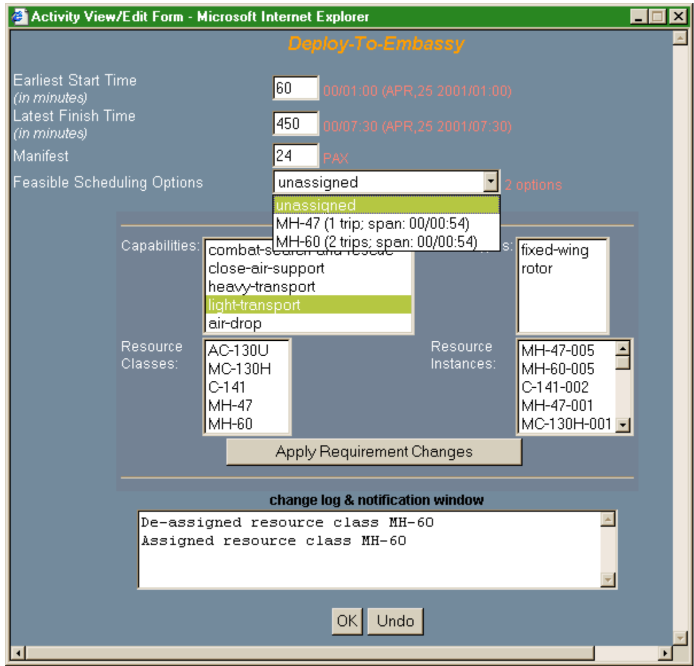

Figure 2.2 Generating Options

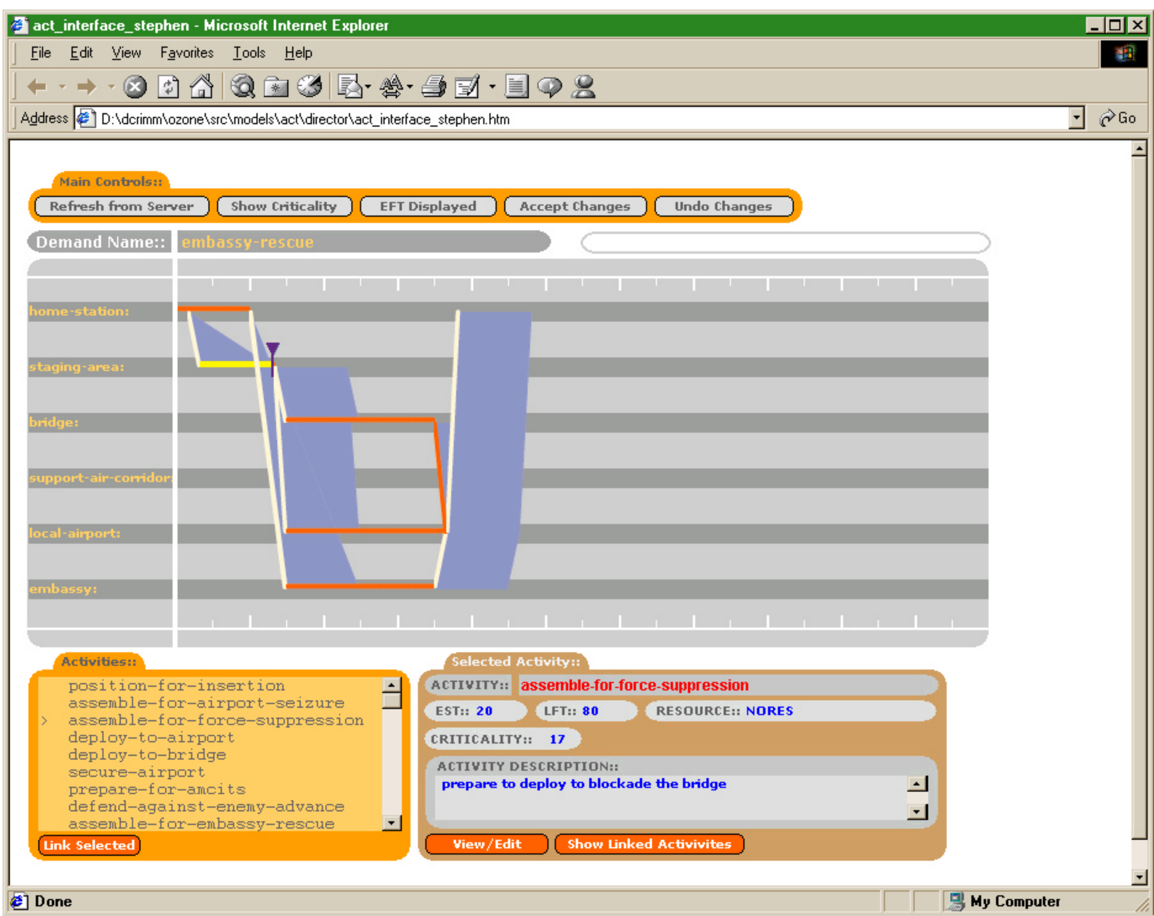

Figure 2.3 Relaxing a Deadline

2.2 System Architecture

An overview of the underlying COMIREM system architecture is presented in Figure 2.4. The planning/scheduling engine is implemented in Common Lisp as an extension of the Ozone scheduling framework [Smith et al. 1996]. It operates together (i.e., in the same Lisp process) with a Common Lisp HTTP server (either AllegroServe in Allegro Common Lisp or CL-HTTP in Macintosh Common Lisp). This facilitates the transmission of either (1) dynamically generated HTML/XML responses or (2) files from disk (e.g., HTML/XML documents) to multiple Internet browser clients using standard CGI (Common Gateway Interface), and is used to support both visualization and interactive development of plans and schedules. The coupling of the core planner/scheduler with a web server provides immediate interoperability with most any web browser (and by extension, nearly any hardware platform), and even more importantly, the broad range of existing interactive visualization tools and applications designed to work with them. The COMIREM system is therefore able to take advantage of the wealth of web-based visualization tools that already exist and allow the user to interact with it using whatever tool is best suited for the present circumstances (e.g., hardware considerations, user preference).

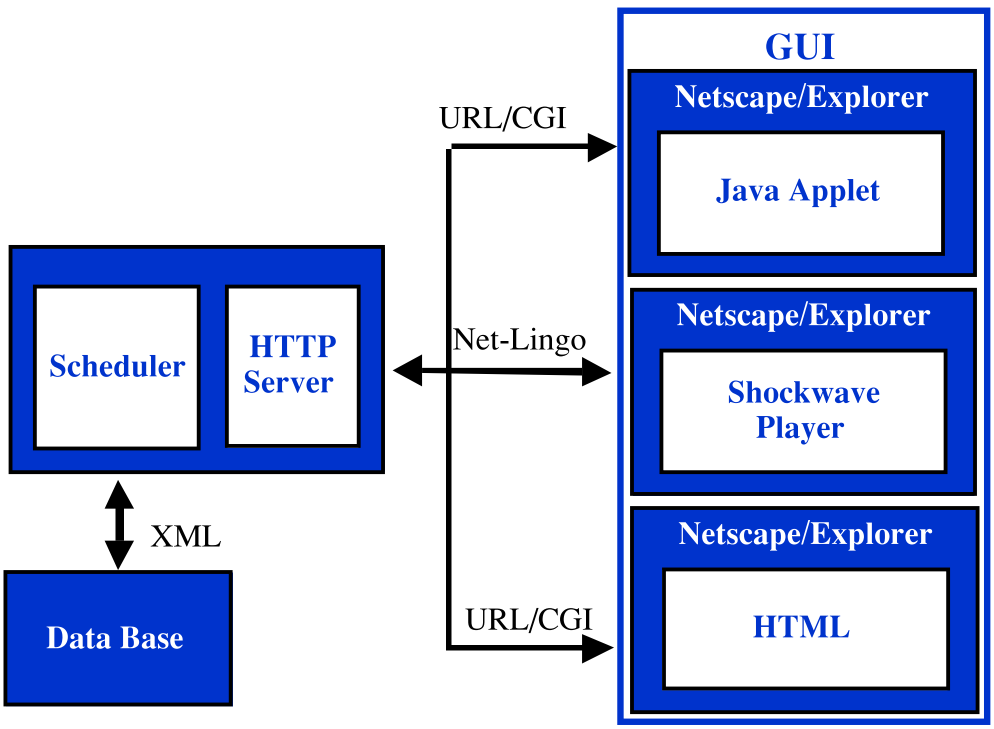

Figure 2.4 COMIREM Architecture

COMIREM is presently capable of interacting with three different types of visualization and interaction tools.

- A Shockwave application serves as the primary user interface component (see Figure 2.1). This choice reflects an underlying objective to explore the potential of more modern web-based design tools for interactive visualization and manipulation of plans and schedules. This application has been built using Macromedia's Director program and is executable in any common web browser using a freely available plug-in application.
- The architecture also supports the use of standard HTML forms as an information display and input medium. We envision such interface components as being particularly necessary in some mobile circumstances, where screen real estate and computational considerations may prohibit extensive graphical interface capabilities. Within the current COMIREM interface, forms are provided for adding, retracting and editing various plan constraints and resource assignments.
- Finally, provision is made to interact and inter-operate with externally developed Java applets. The current COMIREM interface exploits one such component: an encapsulated interactive display for conveying and manipulating resource capacity information developed in a previous scheduling application [Becker and Smith 2000]. In general, our current user interface design perspective is to de-emphasize Java components in favor of an expanded functionality Shockwave application. While inter-operability with Java applets is important given their widespread use in web application environments, our experience to date nonetheless indicates that Shockwave offers the potential for much more lightweight software components (i.e., smaller footprint, no run-time library support required, etc.)

Each of these interfaces communicates with the COMIREM server via standard HTTP protocol (currently HTTP/1.1). Requests from browser clients are encoded as URLs and transmitted to the server, and HTML/XML responses are then generated (or uploaded from disk) and returned to the client web browser using CGI. (Direct peer-to-peer socket communication with the COMIREM server is also supported, as in the case of the Java-based resource inspector described above.) For the most part, interaction with the COMIREM planner/scheduler is simply achieved by launching a web browser and requesting a specific web page.

3. Status and Directions

We have demonstrated the use of the current COMIREM prototype in the context of an embassy rescue scenario. Our short term goals are aimed in three directions: (1) at expanding the system's search capabilities to support a range of automated and semi-automated (i.e., user focused) planning and scheduling modes, (2) at deepening the system's domain model to incorporate all relevant resource and planning constraints, and (3) at providing components for flexibly generating and comparing alternative decision options. In the longer term, our interests lie in investigating issues of collaborative plan/schedule development by multiple users, and in dynamic configuration of planning and scheduling services to fit particular mobile decision-making circumstances.

References

[Becker and Smith 2000] Becker M. and S.F. Smith, "Mixed-Initiative Resource Management: The AMC Barrel Allocator", *Proceedings 5th International Conference on AI Planning and Scheduling,* Beckenridge CO., April 2000.

[Smith et al. 1996] Smith, S.F., O. Lassila and M. Becker, "Configurable Systems for Mixed-Initiative Planning and Scheduling", in (ed. A. Tate) *Advanced Planning Technology*, AAAI Press, 1996.