Technical Paper

O-Plan2: Tucson Demonstration

Brian Drabble, Austin Tate & Jeff Dalton

Artificial Intelligence Applications Institute
University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
United Kingdom

September 15, 1994

ARPA-RL/O-Plan/TP/9 Version 1

The O-Plan2 Project at the Artificial Intelligence Applications Institute of the University of Edinburgh is exploring a practical computer based environment to provide for specification, generation, interaction with, and execution of activity plans. O-Plan2 is intended to be a domain-independent general planning and control framework with the ability to embed detailed knowledge of the domain. The system combines a number of techniques:

- A hierarchical planning system which can produce plans as partial orders on actions.
- An agenda-based control architecture in which each control cycle can post pending tasks during plan generation. These pending tasks are then picked up from the agenda and processed by appropriate handlers (*Knowledge Sources*).
- The notion of a "plan state" which is the data structure containing the emerging plan, the "flaws" remaining in it, and the information used in building the plan.
- Constraint posting and least commitment on object variables.
- Temporal and resource constraint handling using incremental algorithms which are sensitively applied only when constraints alter.
- O-Plan2 is derived from the earlier Nonlin planner from which it takes and extends the ideas of Goal Structure, Question Answering (Modal Truth Criterion) and typed conditions.
- We have extended Nonlin's style of task description language Task Formalism (TF).

O-Plan2 is aimed to be relevant to the following types of problems:

- project management for product introduction, systems engineering, construction, process flow for assembly, integration and verification, etc.
- planning and control of supply and distribution logistics.
- mission sequencing and control of space probes and satellites such as VOYAGER and ERS-1.

O-Plan2 operates at 3-levels (see Figure 1). A user specifies a task that is to be performed through some suitable interface. We call this process *task assignment*. A *planner* plans to perform the task specified. The *execution system* seeks to carry out the detailed activities specified by the planner while working with a more detailed model of the execution environment. The current O-Plan2 system is able to operate both as a planner and a simple execution agent. The task assignment function is provided by a separate process which has a simple menu interface.

The O-Plan2 project has sought to identify modular components within an AI command, planning and control system and to provide clearly defined interfaces to these components and modules. O-Plan2 is implemented in Common Lisp on Unix Workstations with an X-Windows interface. It is designed to be able to exploit distributed and multi-processor delivery systems in future.

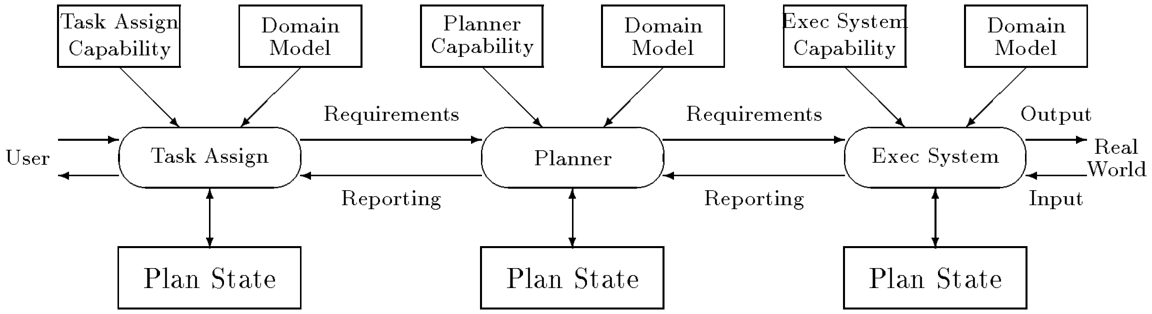

Figure 1: Communication between Strategic, Tactical and Operational Levels

The main components are:

1. Domain Information – the information which describes an application domain and tasks in that domain to the planner.
2. Plan State – the emerging plan to carry out identified tasks.
3. Knowledge Sources – the processing capabilities of the planner.
4. Constraint Managers and Support Modules – functions which support the processing capabilities of the planner and its components.
5. Controller – the decision maker on the *order* in which processing is done.

User Interface

An interface in AutoCAD has been built to show the type of User Interface we envisage (see Figure 2). The window in the top left corner shows the Task Assignment menu and supports the management of authority to plan and execute plans for a given task. The lower window shows a *Plan View* (such as showing the plan as a graph), and the upper right window shows a *World View* for simulations of the state of the world at points in the plan. The particular plan viewer and world viewer provided are declared to the system and the interfaces between these and the planner uses a defined interface to which various implementations can conform. Most of the developer aspects of the planner interface are not shown to the normal user. In figure 2, the developer windows are shown in iconic form along the top edge of the screen.

Demonstration Content

The Edinburgh O-Plan2 project demonstration shows the generation of COA plans in a cut down Tunisian IFD-2 scenario. The plans generated demonstrate O-Plan2's ability to:

- select from a number of missions in order to deter an enumerated set of immediate threats,
- to develop the transportation plans required to move the selected forces from their initial positions to their point of employment.

For the purposes of the demonstration the number of available army, navy and airforce units have been reduced from those available in the original IFD-2 domain description. However, the number of initial facts (associated with the reduced forces), operators and tasks is the same as the original IFD-2 domain description.

Work is now underway for mid 1994 to show the O-Plan2 planner working with an enriched model of resources related to NEO evacuee transportation in the PRECiS environment. A later demonstration in 1995 is intended to show how plans can be generated, their execution monitored with simple problem fixes applied in the PRECiS environment.

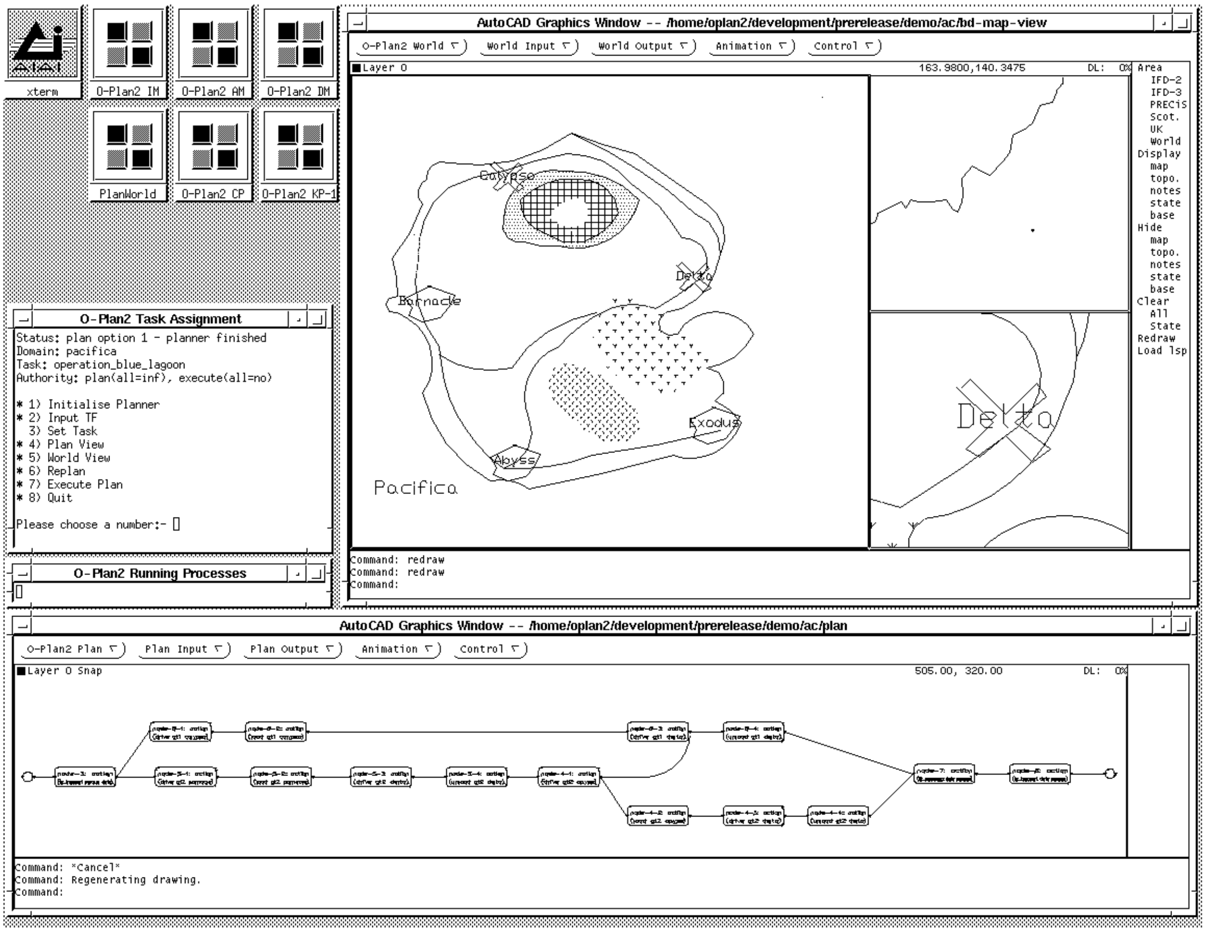

Figure 2: Example Screen for the AutoCAD-based PlanView User Interface