Using Intentional and Attentional Structure for Anaphor Resolution

Monte Zweben*
MITRE Corporation
Burlington Road, Bedford, MA 01730
(617) 271-7026

12 February 1987

Abstract

This paper describes the *Scenes* knowledge representation that captures the intentional and attentional structure of discourse. Using this information a natural language interface can isolate context and resolve anaphors with focusing heuristics. Further, anaphor resolution can be coordinated with interruptions so that completed digressions are ignored.

1 Introduction

One of the goals of the KING-KONG Expert System Interface developed at the MITRE Corporation is to perform anaphoric resolution using a model of discourse. Grosz and Sidner [Grosz & Sidner 86] claim that any discourse has three main constituents: 1) the structure of the actual sequence of discourse utterances; 2) a structure of intentions; 3) an attentional state. This paper describes *Scenes* [Zweben & Chase 87] which are declarative knowledge representations of the intentional and attentional structure of discourse that facilitate anaphor resolution. Utilizing the attentional structure stored in scenes, anaphors can be resolved. Further, since scenes delineate interruptions, resolution strategies can correctly ignore antecedents that reside in interruptions. This paper describes the discourse model underlying *scenes*, the *scene* mechanism and finally, the anaphor resolution algorithm employed.

2 Intention and Attention

Grosz and Sidner distinguish between the intentional state of discourse and the attentional state. Intentional structure represents the underlying purposes that causally relate the utterances of a coherent discourse. Attentional state, on the other hand, captures the focus of attention in the discourse at any one moment, by recording the salient objects and relationships.

*This work was supported by MITRE Sponsored Research Project 90780.

2.1 Intentional Structure

Discourses can be partitioned into segments, each representing some purpose or intention. These discourse segment purposes (DSP's) can be related in special ways. Grosz and Sidner present two kinds of DSP relationships: dominance and satisfaction-precedence. The satisfaction-precedence relation represents one intended action being a pre-requisite of another. The dominance relation states that satisfying one intended action contributes to the satisfaction of another. This relation establishes a hierarchical structure of DSP's representing their dependencies. The intentions that dominate each other depend upon the type of discourse (e.g., general conversation vs. task-oriented dialogue). An expert system interface is primarily concerned with the computer accomplishing tasks. The DSP dominance relation, in an expert system interface, adopts the task-oriented dialogue rule presented here:

$\forall_{i=1,\ldots,n}[Intend(user, Intend(computer, Do(A))) \bigwedge$
$Intend(user, Intend(computer, Do(a_i))) \bigwedge$
$Believe(user, Generates(A, a_1, a_2, , \ldots, a_n))]$
$\longleftrightarrow$
$Dominates(Intend(user, Intend(computer, Do(A))),$
$Intend(user, Intend(computer, Do(a_i))))$

A general interpretation of the above is: If the user intends that the expert system executes tasks A and a_i , and the user believes that the performance of task a_i contributes to the performance of task A, then the intention concerning task A dominates the intention of task a_i.

2.2 Attentional Structure

The attentional structure captures the focus of attention in a discourse. It represents the prominent objects and relationships that are dynamically encountered in conversation. The attentional state is modeled by a set of focus spaces and rules for transitioning among them. Focus spaces are paired with their respective discourse purposes to associate intentional state with attentional state. One can view a focus space as the representation of *what* the discourse participant is talking about, while its association with intentional state explains *why*.

3 Knowledge Representation - Scenes

Scenes are knowledge representations of the intentional and attentional states of discourse. They are schema representations [Minsky 75], [Bobrow & Collins 75] , [Schank & Abelson 77] of plans of stereotypical interactions with the expert system. The hierarchical structure of scenes represents the dependencies of the user's intended actions according to the dominance relation defined for intentional structure. Thus, the root of a scene hierarchy represents the overall discourse purpose (DP) of the dialogue, and each remaining scene , in a hierarchy, supports its dominating scene.

In addition to intentional structure, scenes constrain the attentional structure of a discourse by defining the kinds of objects that would be prominent if a scene were active. These object descriptions, called the *roles* of a scene, represent the players participating in the action that the scene represents. Scene recognition is directed by the roles that are observed in the conversation; this process is described in detail later.

When a scene is appropriate (i.e., recognized as the current intentional state), it is instantiated to represent attentional state. Its roles are filled with the referents of the objects in the current clause and other objects already present in the discourse. The preceding scene is linked to the new one, maintaining a predecessor/successor network of scenes modeling the discourse.

Scenes

The Intentional Structure		The Attentional Structure	
Field	Description	Field	Description
Name	The type of scene.	Role-Fillers	The semantic representations of those objects filling the roles.
Roles	The prominent object classes.		
Inferiors	The scenes that this one dominates.	Predecessors	The scenes preceding this one in the actual discourse.
Superior	The scene dominating this one.		
Enables	The post-requisite scenes.	Sucessors	The scenes following this one.
Enables-by	The pre-requisite scenes.	Focus Cache	The objects available for anaphoric reference.
Triggers	The lexical items that are recognized for this scene and their filtering maps.	Expert System Goal	The abstract expert system actions to apply.

Figure 1: The slots of a scene.

An important distinction must be made between a plan and an intentional scene hierarchy. The scenes represent **stereotypical interactions** with an expert system. However, they **do not** represent the sequence of actions an expert system will take. This information is captured in the expert system's plans, goals and problem-solving strategies. Only data relevant to the user-machine interface is captured by the scene hierarchy.

To clarify the exposition of a scene hierarchy, the following figures present examples of scenes from the KRS mission planning application. The primary goal of this application is to plan a OCA mission task. In order to complete the plan, among other things, a target, an airbase, and a type of aircraft must be chosen. The KRS system is a mixed-initiative system which can fully plan missions or guide a user along using its constraint satisfaction mechanism. The following scenes represent the interactions with a user planning a mission. Figure 2 represents the intentional and attentional structure of the following discourse.

1. Build a mission.
2. Leave from Halfort
3. Send F-4cs.
4. Make Mermin the target.
5. What is the range of an F-4c?

By capturing both intentional and attentional state, our expert system interface demonstrates the ability to perform anaphoric reference. Intentional structure (i.e., scene hierarchies) enables the response handler to perform limited plan recognition [Allen & Perrault 80], [Cohen & Perrault 79], [Sidner & Israel 81], [Sidner 83b], [Litman 86], which isolates context, while attentional structure provides the dynamic information about objects in conversation. The next section describes scene recognition. Subsequently, an extensive demonstration of the system is presented followed by a description of the anaphoric reference algorithm.

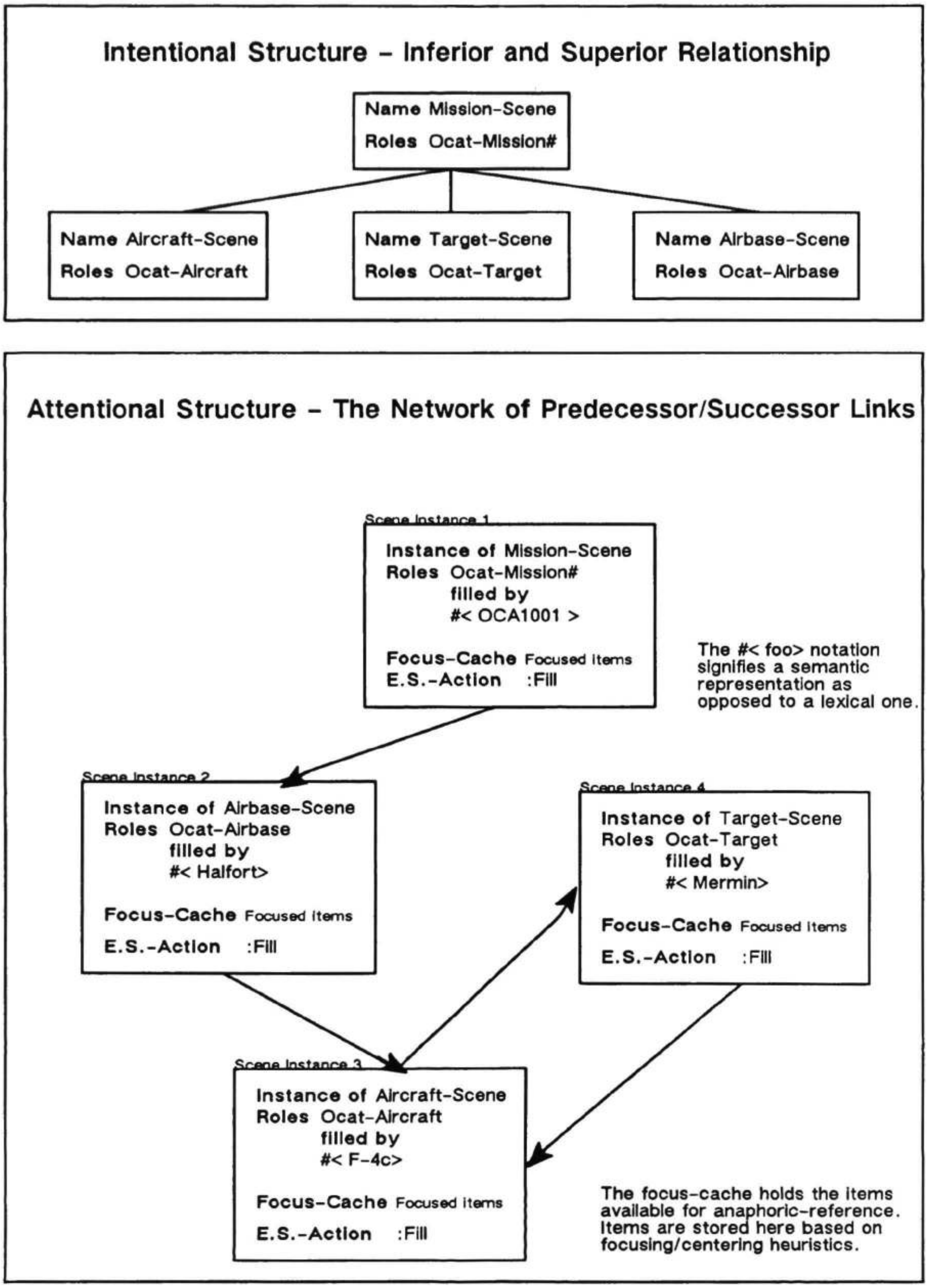


Figure 2: An Example of Intentional and Attentional Structure.

4 Scene Recognition

Scene recognition is the process of determining attentional state. This process is failure driven; a new scene is found if the response mechanism is unable to interpret the input in the current scene. The response handler can either tell the scene controller what scene to move to, or it can instruct the scene controller to use its discourse heuristics to find the new attentional state.

4.1 General Control Flow

Scene recognition is a generate and test process in which heuristics guide the generation of possibilities and roles filter them. When a new scene is required, intentional and attentional structure is used to provide new possibilities. For each scene proposed, the current input clause is tested against the lexical triggers of the scene, which maps the head verb, the arguments and the modifiers of the sentence to the roles of the scene. The goal of this test is to determine whether the referents of the semantic arguments and modifiers in the sentence match the role description specified in the lexical trigger mapping. If all the referents are consistent with the role description, the proposed scene becomes the current scene and is inserted into the predecessor/successor network. Here is an example of an inconsistent match:

Lexical Trigger: Hit [OBJ → Airbase, INSTR → Aircraft]

Input Clause: Hit *the tank* with *an ordnance*.
Both the object and instrument violate the lexical trigger map.

If the heuristics fail to provide a current scene, the user is asked what context his utterance pertains to (ie. which scene is appropriate) and is then requested to re-phrase his input. In the future, we hope to provide intelligent failure mechanisms with the ability to learn new scenes [Mooney & DeJong 85] and reason about misconceptions [Pollack 86].

4.2 Scene Heuristics

Some of the scene heuristics that currently generate possibilities for scene recognition are:

1. *Intentional Clues* - Choose a context that follows the plan.
   - preceded-by - Try all the post-requisites.
   - precedes - Try all the pre-requisites.
   - superior - Try the more general scene.
   - inferiors - Try the supporting scenes.
   - siblings - Try the scenes at the same intentional level.
   - all-relatives - Try all the scenes that are causally related.
2. *Attentional Clues* - Choose a context that was recently referred to.
   Backtrack through a scene's predecessor/successor network.
3. *Interrupt* - Find a new scene in a different dominance hierarchy.
   Sequence through the contextual lexicon to find a scene that recognizes the main verb of the sentence.
4. *Ask* - Query the user for the current context.
   For each known scene, ask the user whether it is the intended context.

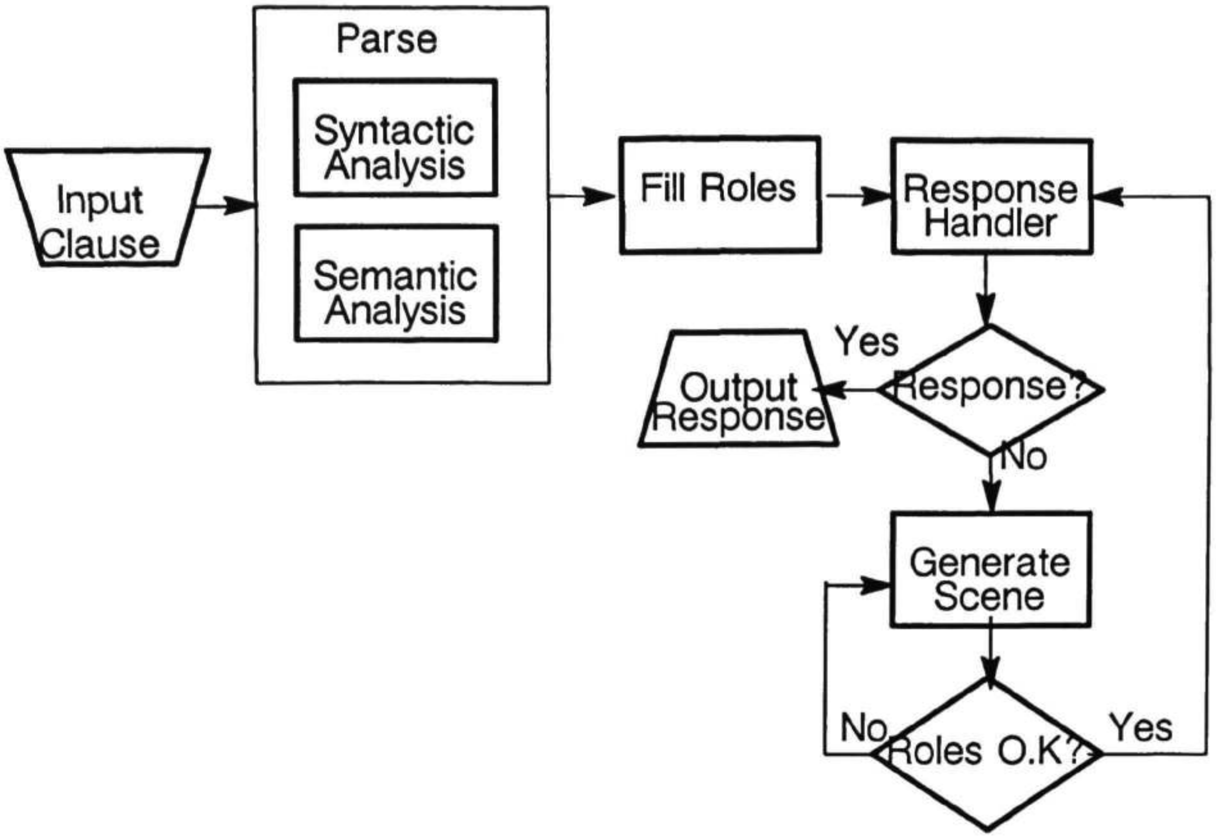


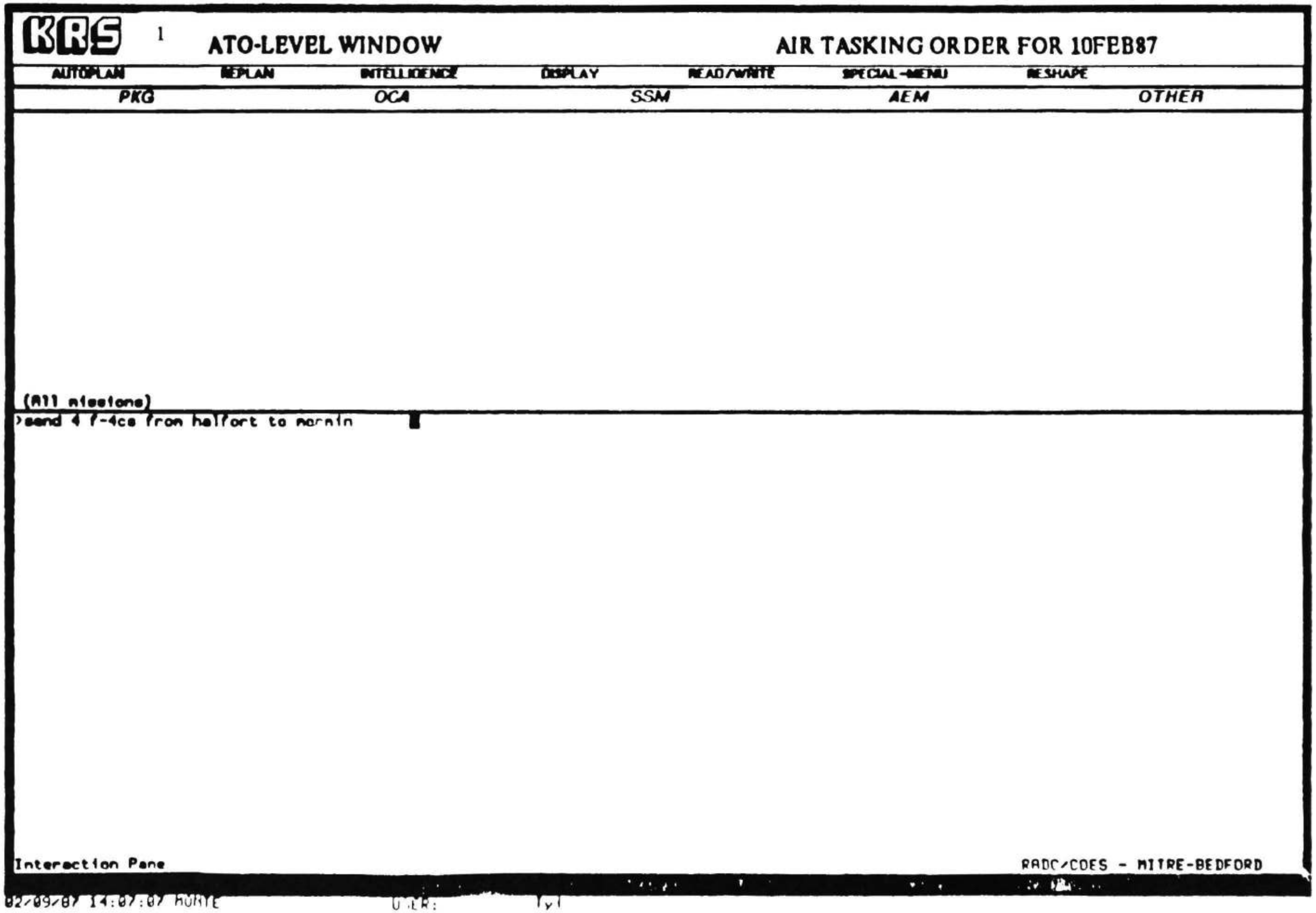
Figure 3: Overall Control Flow

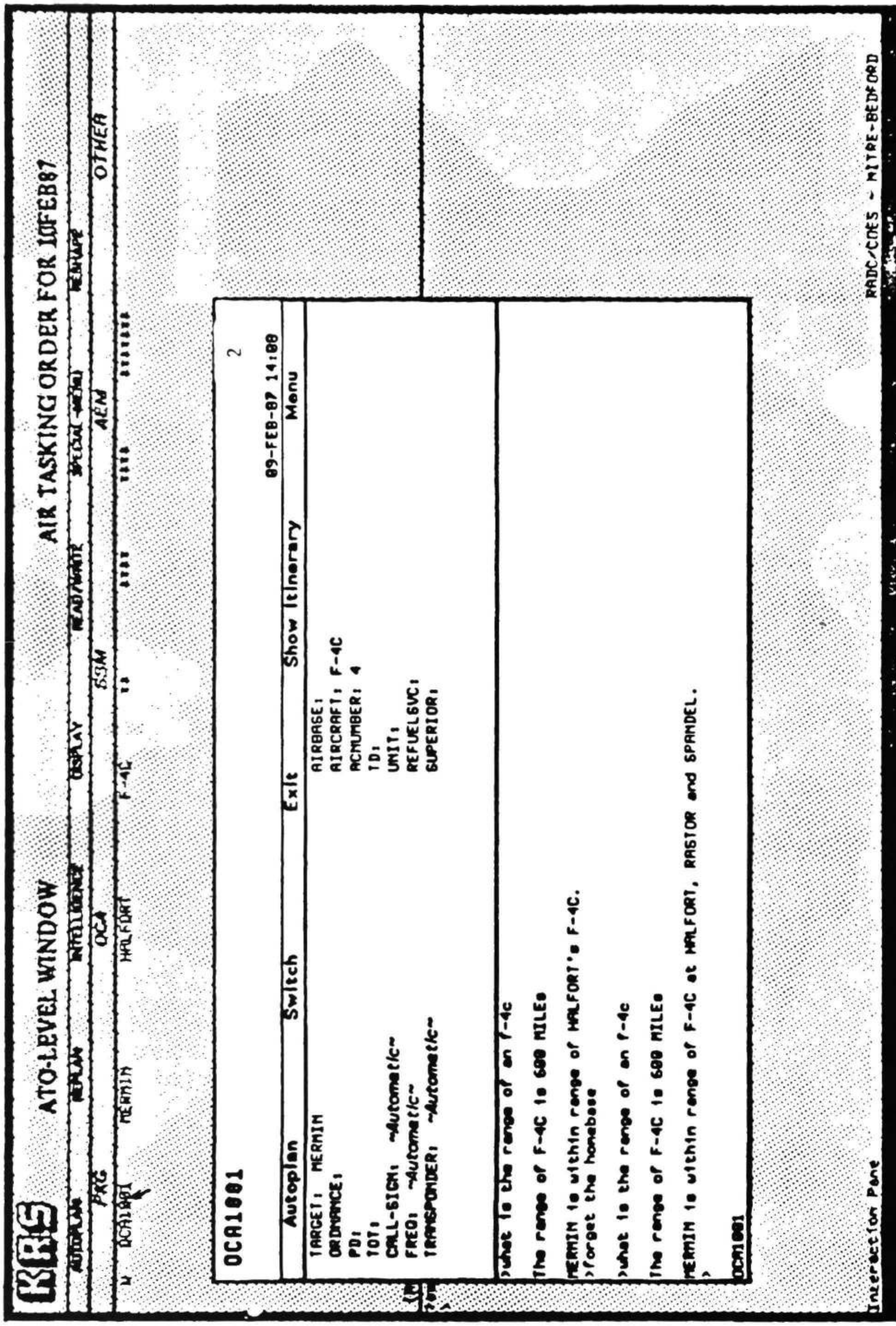
5 Implementation

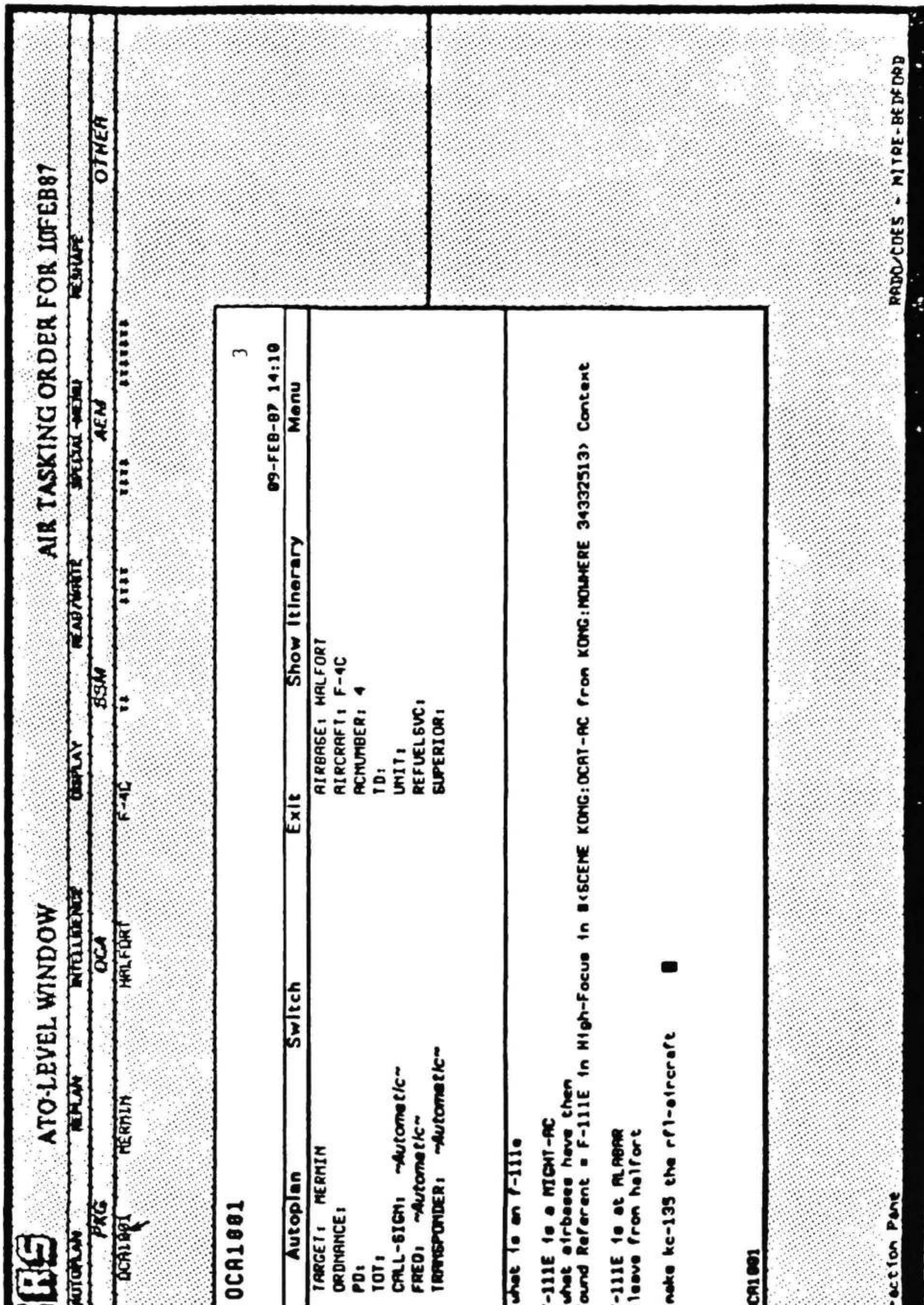
The entire interface is implemented in Zetalisp on Symbolics lisp machines. Scene instances are represented as flavor instances with instance variables for the fields presented and methods for executing the heuristics. Scene recognition is handled by the scene controller, which is also implemented as a flavor instance to retain dynamic information and to facilitate integration with the expert system's i/o loop. The overall control structure is shown in Figure 3.

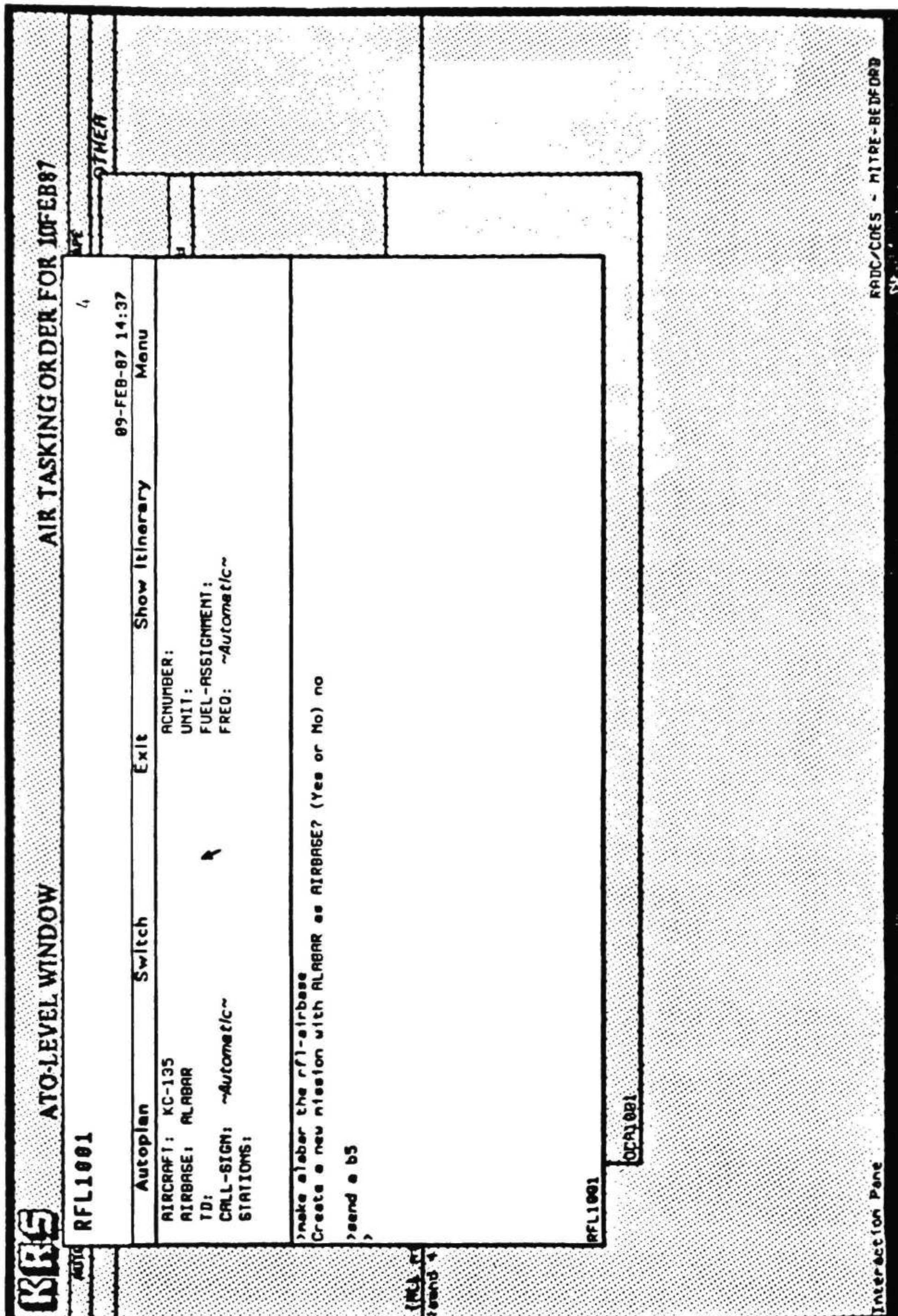
6 Demonstration

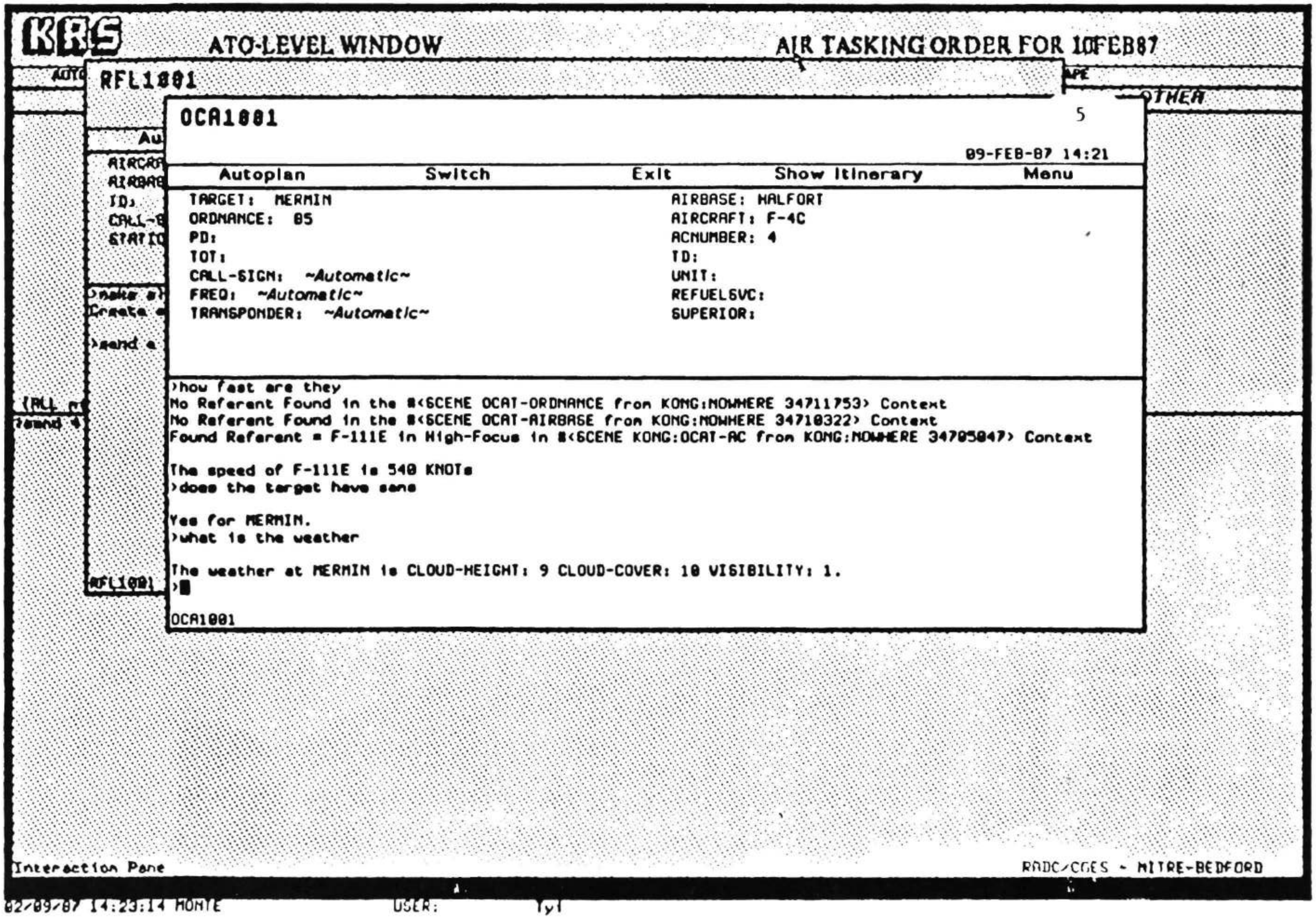
- Pane 1 is the KRS main window with a command entered to plan a mission.
- In response to the first sentence, KRS creates the mission frame shown in Pane 2 with the appropriate slots filled in. The sentences in Pane 2 demonstrate the ability to respond intelligently and the ability to respond differently to the same utterance in different contexts.
- In pane 3, f-111es are made prominent and then they are referred to with a pronoun.
- Pane 4 is a digression into a refueling context.
- Pane 5 demonstrates that the scene maintained its attentional state with an anphoric reference to the salient F-111e. The conversation ends with a contextual inference that used the context of targets to constrain the weather query.


KRS
1
ATO-LEVEL WINDOW
AIR TASKING ORDER FOR 10FEB87
AUTOPLAN
REPLAN
INTELLIGENCE
DISPLAY
READ/WRITE
SPECIAL-MENU
RESHAPE
PKG
OCA
SSM
AEM
OTHER
(All missions)
>send 4 f-4cs from halfort to mornin
Interaction Pane
RADC/COES - MITRE-BEDFORD


KRS
ATO-LEVEL WINDOW
AIR TASKING ORDER FOR 10FEB87
AUTOPLAN
REPLAN
DISPLAY
PKG
OCA
SSM
AEW
OTHER
OCA1001
MERMIN
HALFORT
F-4C
OCA1001
2
09-FEB-87 14:00
Autoplan
Switch
Exit
Show Itinerary
Menu
TARGET: MERMIN
ORDNANCE:
PD:
TOT:
CALL-SIGN: ~Automatic~
FREQ: ~Automatic~
TRANSPONDER: ~Automatic~
AIRBASE:
AIRCRAFT: F-4C
ACNUMBER: 4
TD:
UNIT:
REFUELSVC:
SUPERIOR:
>what is the range of an f-4c
The range of F-4C is 600 MILEs
MERMIN is within range of HALFORT's F-4C.
>forget the homebase
>what is the range of an f-4c
The range of F-4C is 600 MILEs
MERMIN is within range of F-4C at HALFORT, RASTOR and SPANDEL.
>
OCA1001
Interaction Pane
RADC/COES - MITRE-BEDFORD


KRS
ATO-LEVEL WINDOW
AIR TASKING ORDER FOR 10FEB87
AUTOPLAN
REPLAN
INTELLIGENCE
DISPLAY
READ/WRITE
SPECIAL-MENU
RESHAPE
PKG
OCA
BSM
AEW
OTHER
OCA1001
HERMIN
HALFORT
F-4C
OCA1001
3
09-FEB-87 14:10
Autoplan
Switch
Exit
Show Itinerary
Menu
TARGET: HERMIN
ORDNANCE:
PD:
TOT:
CALL-SIGN: ~Automatic~
FREQ: ~Automatic~
TRANSPONDER: ~Automatic~
AIRBASE: HALFORT
AIRCRAFT: F-4C
ACNUMBER: 4
TD:
UNIT:
REFUELSVC:
SUPERIOR:
>what is an F-111e
F-111E is a NIGHT-AC
>what airbases have them
Found Referent = F-111E in High-Focus in #<SCENE KOMG:OCAT-AC from KOMG:NOWHERE 34332513> Context
F-111E is at ALABAR
>leave from halfort
>make kc-135 the rfl-aircraft
OCA1001
Interaction Pane
02/09/87 14:11:24 MONTE
USER:
RADO/COES - MITRE-BEDFORD


ATO-LEVEL WINDOW
AIR TASKING ORDER FOR 10FEB87
RFL1001
09-FEB-87 14:37
Autoplan
Switch
Exit
Show Itinerary
Menu
AIRCRAFT: KC-135
AIRBASE: ALABAR
TD:
CALL-SIGN: ~Automatic~
STATIONS:
ACNUMBER:
UNIT:
FUEL-ASSIGNMENT:
FREQ: ~Automatic~
>make alabar the rfl-airbase
Create a new mission with ALABAR as AIRBASE? (Yes or No) no
>send a b5
OCA1001
RFL1001
Interaction Pane
RADC/COES - MITRE-BEDFORD


KRS
ATO-LEVEL WINDOW
AIR TASKING ORDER FOR 10FEB87
RFL1001
OTHER
OCA1001
5
09-FEB-87 14:21
Autoplan
Switch
Exit
Show Itinerary
Menu
TARGET: MERMIN
ORDNANCE: 05
PD:
TOT:
CALL-SIGN: ~Automatic~
FREQ: ~Automatic~
TRANSPONDER: ~Automatic~
AIRBASE: HALFORT
AIRCRAFT: F-4C
ACNUMBER: 4
TD:
UNIT:
REFUELSVC:
SUPERIOR:
>how fast are they
No Referent Found in the #<SCENE OCAT-ORDNANCE from KONG:NOWHERE 34711753> Context
No Referent Found in the #<SCENE OCAT-AIRBASE from KONG:NOWHERE 34710322> Context
Found Referent = F-111E in High-Focus in #<SCENE KONG:OCAT-AC from KONG:NOWHERE 34705047> Context
The speed of F-111E is 540 KNOTs
>does the target have sams
Yes for MERMIN.
>what is the weather
The weather at MERMIN is CLOUD-HEIGHT: 9 CLOUD-COVER: 10 VISIBILITY: 1.
>
OCA1001
RFL1001
Interaction Pane
RADC/CGES - MITRE-BEDFORD
02/09/87 14:23:14 MONTE
USER:
Ty1

7 Anaphors

Pronouns are processed using simple focus heuristics [Sidner 83a], [Reichman 85]. Each scene has an ordered focus cache of semantic representations that are salient. After each sentence is processed, its semantic arguments are pushed onto the current scene's cache. When a pronoun is encountered, the focus cache is searched for a referent. This search requires a semantic analysis that checks whether the cached object makes sense as a substitute for the pronoun. If successful, the object is moved to the front of the list and a message is supplied to the user to inform him of the pronoun resolution. Otherwise, a message that no referent was found in the current scene is provided followed by recursive search in the previous scene's focus cache. Currently, the system does not forget focussed objects and backtracks exhaustively through the scenes until successful. Anaphor resolution is integrated with an interruption component of the scene controller so that completed interruptions do not provide possible antecedents, which is the topic of the next section.

7.1 Interruptions

When the scene controller chooses a scene in a different scene hierarchy, it is changing its expectation of the overall intention of the user. This represents an interruption which is flagged accordingly in the new scene. When the digression is complete and the user returns to the old context, the old context is marked. When searching for antecedents, this region, which represents a full interruption, is skipped. This is shown in the demonstration when the system is asked, "How fast is it ?". The most recent antecedent is the *kc-135* referred to in the interruption. Instead, the interruption is skipped and the *F-111e* is chosen. This demonstrates the utility of scenes and their ability to retain and manage attentional state. Even though the interruption causes a context change, the salient objects (e.g. *F-111e*) are maintained and the interruption is marked, thus enabling the correct anaphor resolution.

7.2 Other Approaches

This approach to anaphoric resolution differs from previous attempts in the following manner: the algorithm is based upon a discourse model that distinguishes intentional and attentional structure. This integrates anaphor resolution with context processing, resulting in the ability to "skip" over interruptions as possible placeholders for antecedents. Most commercial systems simply search for the most recent NP, which often leads to strange resolutions. Further, the use of focus heuristics is not sufficient to provide intelligent resolutions. The most desirable approach is to use focus heuristics on top of a discourse model, which was originally proposed by Grosz and Sidner [Grosz & Sidner 86].

However, the scenes mechanism differs from Grosz and Sidner's model in the representation of attentional state. We maintain a predecessor/successor graph of focus spaces, while Grosz and Sidner use a focus stack. In their model, focus spaces are pushed onto the stack until completed. Anaphors can be resolved with antecedents found by iterating through the focus spaces on the stack until one of two conditions: an interruption is found or the bottom of the stack is encountered. When a focus space is is completed, it is popped off the stack, thereby making its focused objects inaccessible. Hence, completed interruptions are no longer available to provide possible antecedents. If the same intentional state is returned to later, a new focus space is instantiated, which ignores the objects that were prominent in the old space. In the scenes system, completed focus spaces are flagged as such, but they maintain their focus cache. If a discourse participant returns to a closed scene, the cached items in the focus list are still available for reference. While this capability exists in the scene mechanism, the linguistic evidence for this behavior is not conclusive. Nevertheless, users of the natural language system seem to refer to objects that are prominent in completed focus spaces. Grosz and Sidner claim that all these items must be re-introduced to be available for anaphoric reference. Here is an interaction which seems contradictory to this assumption.

User: Are F-4cs and F-111es night aircraft ? *(Aircraft Scene)*

Computer: *Yes*

User: Send an F-4c. *(In a stack model, this focus space would now be popped)*

Computer: *O.K.*

User: Send a B10. *(Ordnance Scene)*

Computer: *F-4cs do not carry B10's.*

User: Send the other one. *(In a stack model, the referent would have been popped)*

The scenes mechanism is able to skip over interruptions without enforcing the stack model. The discourse is represented as a predecessor/successor graph of the scenes used, possibly marked with interruption boundaries. Anaphors are resolved by backtracking through this graph, skipping interruptions. Currently, this graph is never pruned creating a very large structure in lengthy interactions. We recognize the problems of maintaining this graph but the use of the stack model seems too drastic because of examples like the one above. More research is necessary to determine the correct model of attentional state.

8 Conclusion

We have designed a natural language interface that makes extensive use of the **Scene** knowledge representation. Scenes are based upon Grosz and Sidner's model of discourse that distinguishes attentional and intentional structure. This knowledge representation facilitates limited plan recognition which establishes the context of an utterance. Further, it captures the salient objects of a discourse as well as maintaining a model of a user's interaction. Utilizing this information, the interface resolves anaphoric references.

References

[Allen & Perrault 80] Allen, J.F. & Perrault, C.R. (1980). Analyzing Intention in Utterances. *Artificial Intelligence*, 15:143–178.

[Bobrow & Collins 75] Bobrow, D. & Collins, A.M. (1975), editor. *Representation and Understanding*. Academic Press, New York.

[Cohen & Perrault 79] Cohen, P. & Perrault, C.R. (1979). Elements of a Plan-Based Theory of Speech Acts. *Cognitive Science*, 3:177–212.

[Grosz & Sidner 86] Grosz, B.J. & C. L. Sidner, C.L. (1986). Attentions, Intentions, and the Structure of Discourse. *Computational Linguistics*, 12:175–204.

[Litman 86] Litman, D.J. (1986). Linguistic Coherence: A Plan-Based Alternative. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, pages 215–223, New York.

[Minsky 75] Minsky, M. (1975). A Framework for Representing Knowledge. In P. H. Winston, editor, *The Psychology of Computer Vision*, McGraw-Hill, New York.

[Mooney & DeJong 85] Mooney, R. & DeJong, G. (1985). Learning Schemata for Natural Language Processing. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 681–687, Los Angeles.

[Pollack 86] Pollack, M.E. (1986). A Model of Plan Inference that Distinguishes Between the Beliefs of Actors and Observers. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, pages 207–214, New York.

[Reichman 85] Reichman, R. (1985). *Getting Computers to Talk Like You and Me*. The MIT Press, Cambridge, MA.

[Schank & Abelson 77] Schank, R.C. & Abelson, R.P. (1977). *Scripts, Plans, Goals and Understanding*. Lawrence Erlbaum Associates, Hillsdale, N. J.

[Sidner & Israel 81] Sidner, C.L. & Israel, D.J. (1981). Recognizing Intended Meaning and Speakers' Plans. In *Proceedings of Seventh International Joint Conference on Artificial Intelligence*, pages 203–208, Vancouver, B. C.

[Sidner 83a] Sidner, C.L. (1983a). Focusing in the Comprehension of Definite Anaphora. In M. Brady and R.C. Berwick, editors, *Computational Models of Discourse*, The MIT Press, Cambridge, MA.

[Sidner 83b] Sidner, C.L. (1983b). What the Speaker Means: The Recognition of Speakers' Plans in Discourse. *International Journal, of Computers and Mathematics*, 9:71–92.

[Zweben & Chase 87] Zweben, M. & Chase, M.P. (1987). Scenes: A Representation of Intentional and Attentional Structure. Submitted to AAAI-87, Seattle, WA.