Adding Explanation Capability to a Knowledge-Based System for Interpretation of Oceanographic Images

Susan Bridges
Mississippi State University
Starkville, MS 39762-5623

Abstract

An explanation component has been developed for an existing oceanographic expert system that predicts the movement of mesoscale features associated with the Gulf Stream. The information provided by the expert system is used by image processing analysts when the oceanographic features cannot be observed by satellite data due to interference such as cloud cover. The addition of an explanation capability gives users a basis for judging the quality of the system's decision making process.

The structure of the original system was not amenable to the incorporation of an explanation facility because the knowledge needed for explanation was not explicitly represented in the knowledge base. The system has been restructured with the knowledge represented declaratively rather than procedurally, thus, allowing the reasoning process to be recorded and used to produce explanations of decisions. The rules have been rewritten with the knowledge "chunks" in each rule at a finer level of granularity. Each rule corresponds to one decision and the results of each decision are explicitly asserted into the working memory of the system. The presence of the new information causes other rules to fire and other decisions to be made. An explanation is produced by capturing the chain of rules that have fired. In addition to a reduction in granularity, the rules have also been generalized, which allows the same rules to be used in many different situations with different instantiations of the variables.

The explanation component consists of an introspection module and a presentation module. The introspection module "watches" the reasoning process and records the data that caused each rule to fire and the new information produced as a result of each rule firing. The presentation module can use this information to present a detailed natural language trace of the rules that have fired or a shorter natural language summary of the reasoning used for the prediction. The trace will be most useful for those who are debugging the system, those who wish to modify the system, or those who need a detailed account of the system's reasoning. The summary will be more useful for the analysts who will use the system on a daily basis.

Acknowledgments

This document was prepared with support from the Office of Naval Technology, under Program Element 0602435N, T. Warfield, Program Manager. Susan Bridges was supported by the Naval Research Laboratory through the U.S. Navy/ASEE Summer Faculty Research Program.

Adding Explanation Capability to a Knowledge-Based System for Interpretation of Oceanographic Images

1.0 Introduction

The knowledge-based expert system for interpretation of oceanographic images that has been developed under the sponsorship of the Naval Research Laboratory (NRL) predicts the movement of mesoscale oceanographic features in the Gulf Stream region of the Atlantic Ocean. The knowledge-based approach was adopted because little progress has been made in developing numeric models of the movement of these features until quite recently. Although significant progress has been reported in modeling the movement of the Gulf Stream itself, the resolution of models is just beginning to approach the scale needed to model eddy movement (Hurlburt, et al., 1992). The knowledge-based system for predicting eddy movement encapsulates knowledge from NRL experts and the technical literature about the typical movement of warm and cold eddies in different areas of the Gulf Stream including the effects of interactions of different oceanographic features (Thomason, 1986).

Although the system has been demonstrated to have significant prediction capability (Lybanon, 1990), there is a reluctance to include it in an operational system (Lybanon, 1990, 1991a, 1991b). The addition of an explanation capability to the prediction system would facilitate use of the system in two ways. First, it would allow managers, who are responsible for making the decision, to move the expert system into operations to see how the system reaches its conclusions and evaluate the reliability of its reasoning process. Second, when the system is used to support decision making, it should be able to provide users with the basis for its predictions. This is especially important because the system is meant to be used as a decision aid for interpreting satellite data and not as a stand-alone system. Thus, it is important that users be able to identify situations where the system's predictions should be called into question.

In order for a knowledge-based system to produce a justification of its reasoning process, it must be able to record the facts and conclusions that are used in reaching a decision and then extract, organize, and present this information to the user. The most crucial factor that determines the extent to which an expert system can produce satisfactory justifications is the extent to which the knowledge used for decision making is explicitly represented in the knowledge base of the system. Unfortunately, in the ocean prediction system, most of the reasoning process is represented implicitly in the ordering of procedural code on the right-hand side (RHS) of each rule. In order to give the system the capability to justify its reasoning, it has proven necessary to reduce the granularity of the rules, so that each rule represents one decision. Although this would appear to cause an explosion in the size of the knowledge base, this was prevented by making the individual rules more general and using the instantiation process to apply these general rules to specific cases.

2.0 Original Structure of the Knowledge-Based System

The original oceanographic expert system was implemented using a combination of the OPS83 expert system development tool, the C programming language, and FORTRAN. OPS83 was used to implement the control structure and the knowledge-based portion of the system. Those portions of the system implemented in C include routines to perform calculations concerning the Gulf Stream and routines to graphically display the predicted movement of the features. Some FORTRAN is also used in the graphics routines.

The general organization of the knowledge-based system is illustrated in Figure 1. The knowledge base of oceanographic information is coded as if-then rules. The *if* part of each rule is called the left-hand side (LHS) and contains a set of patterns. These patterns are matched against data describing the state of the oceanographic features that are stored in working memory. The *then* portion of each rule, also called the RHS, consists of a set of actions to be performed when the rule fires. These actions may include calculations, if-then statements, procedure and function calls, and instructions to add elements to working memory or to modify existing working memory elements. A rule is eligible to fire whenever all the patterns on its LHS are matched against working memory. A particular matching of working memory elements (WME) with rule patterns is called a rule instantiation. At any one time, there may be several rule instantiations that are eligible to fire (including different instantiations of the same rule). A conflict resolution procedure selects one of the eligible rules to fire and the actions specified on the RHS of that rule are executed. The actions performed by a rule may cause changes in working memory that make new rule instantiations possible and make some previous rule instantiations invalid. This process of matching rule patterns against working memory, selecting a rule to fire, and performing the rules actions, is called the recognize-act cycle, and it is repeated until there are no more rules eligible to fire.

This type of rule-based system uses what is called forward chaining or data directed reasoning because it reasons from data to conclusions. It is the presence of data in working memory that determines which rules fire and drives the operation of the system. This "data directed control" is very different from the traditional "sequential control" typical of procedural programming.

In the original knowledge-based system, a set of WMEs representing the current status of each ring is asserted into working memory. At each time step the system cycles through all of the regions, processing all of the warm core rings (WCRs) first, and then all of the cold core rings (CCRs). Each of the major components of the system will be discussed.

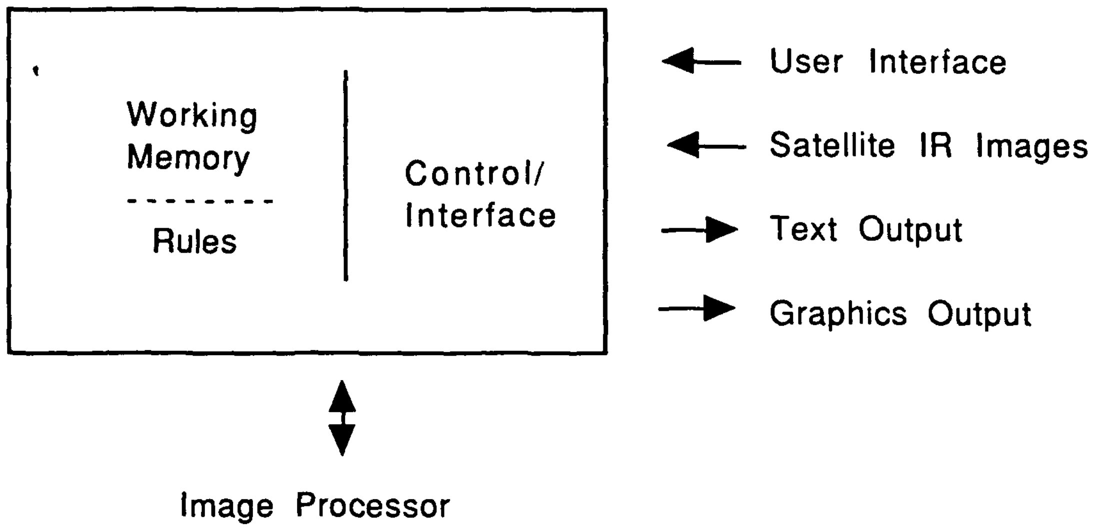

Figure 1. General Organization of the Knowledge-Based System (from Thomason, 1989).

2.1 Working Memory Elements and Other Data Structures

OPS83 requires type definitions for WMEs. The original system included definitions for three WME types: STATUS, CCR, and WCR. The STATUS type was used to instantiate an element that represented the current status of the computation. There was only one element of type STATUS in working memory at any one time. Fields in this WME represented the type of eddy currently under consideration (CCR or WCR), the region currently under consideration, and the current frame. The frame field could take a value of +1 or -1 and was used as a tag in all WMEs to indicate if a particular element had been updated to the current time step.

The other two WME types were used to represent WCRs and CCRs, respectively, and the fields in the two types are identical. Table 1 gives the name of each field, its type, and a description of the values it can take.

Note that when new estimates of the radius, latitude, and longitude of the ring are computed, these values are stored directly in the radius, latitude, and longitude fields for the ring. This means that no history is kept of the state of the ring at different time steps or during the process of the computation. Such a history, if present, could be used in a scheme that predicts ring movement based on their history, as well as facilitating explanation.

Table 1. CCR and WCR element type definitions.

Field name	Type	Value description
ring	symbol	value is WCR or CCR
refno	integer	unique identifier for ring
frame	integer	+1 or -1
region	integer	1-9
GSinteract	logical	initialized 0B, 1B after first rule fires for this ring, 0B after second rule fires
radius	real	radius of ring in km
lat	real	latitude of ring in degrees
long	real	longitude of ring in degrees
decay	real	% radius remaining after 1 wk decay
speed	real	speed in cm/sec (not used)
direction	integer	1-16 representing compass direction initialized to 13 for both WCR and CCR
status	symbol	not used

Directions are represented by positive integers from 1 to 16 that correspond to compass directions as follows: 1 N, 2 NNE, 3 NE, 4 ENE, 5 E, 6 ENE, 7 SE, 8 SSE, 9 S, 10 SSW, 11 SW, 12 WSW, 13 W, 14 WNW, 14 NW, 16 NNW.

In addition to the WME types, two record types are defined for region parameters, one type for CCRs and one type for WCRs. Variables defined to be of these types cannot be used for pattern matching on the LHS of rules, but can be accessed on the RHS of rules and used in computations. Table 2 gives descriptions of the fields for the cold core region record definition (type regc). The fields in the warm core region record type (type regw) are identical except that there are three break-point values instead of four. Break-points for the CCR region parameters are assumed to be strictly ascending in magnitude while those for the WCRs are strictly descending in magnitude. Nine variables of each region type are declared.

The upper and lower boundaries of the Gulf Stream are represented using array structures. This representation will not be discussed since it has not been revised.

Table 2. Type definition for regc record.

Field name	Type	Value description
name	symbol	value is WCR or CCR
speed	real	default speed of rings in region (cm/sec)
direction	integer	1-16 representing compass direction
dellat	real	change in latitude
dellong	real	change in longitude
decay	real	decay factor per week with no Gulf Stream interaction
minrad	real	minimum radius value in km
gsdec	real	additional decay factor per week with Gulf Stream interaction
b1	real	break point for ratio test in ccr looping
b2	real	break point for ratio test in ccr looping
b3	real	break point for ratio test in ccr looping
b4	real	break point for ratio test in ccr looping (not used)

2.2 The Rules

The knowledge base of the current system has a very simple structure that, at first glance, would appear to lend itself to the incorporation of an explanation facility. The system models the movement of the two types of eddies in each of nine different regions of the Gulf Stream. The knowledge base consists of two rules for each type of eddy in each region giving a total of 36 rules. Each rule has a very simple LHS that identifies the type of eddy and the region. The solution strategy is slightly different for CCRs and WCRs.

The first rule for CCRs in each region estimates the new radius for the eddy and asserts a fact into working memory that causes the second rule to fire for that eddy. The second rule also has a very simple set of patterns on the LHS, but the RHS is a very long set of decisions and calculations. This procedural code on the RHS determines the distance of the eddy from the Gulf Stream and calculates a ratio that specified the degree of interaction of the eddy and the Gulf Stream. The change in latitude and longitude of the eddy are then predicted based on the degree of interaction. If the interaction is negligible, then the calculation is very straight forward. If there is significant interaction, then the degree of interaction is used to select an interaction regime that is used for a more complex set of calculations. Depending on the interaction regime selected, changes may be made in the original value calculated for the radius, the direction of movement of the eddy, and the speed at which the eddy travels. The rules for each region are very similar, differing primarily in the parameter values used in the calculations.

The first rule for WCRs in each region not only estimates the new radius but also estimates the new longitude and latitude for the eddy. The RHS of the first rule may also include constraints that model the limitation of the movement of the eddies by physical barriers that occur in specific regions. The second rule estimates the degree of interaction with the Gulf Stream based on the revised radius, longitude, and latitude and then makes revisions in the estimated radius and position based on the degree of interaction. Again, most of the decisions are made on the RHS of the second rule in complex procedural code that is repeated in the rules for each region.

The difficulty in using rules with this type of structure to produce explanations can be seen from an example. If the system is trying to predict the movement of a warm-core eddy in region 4, then the classification of the degree of interaction with the Gulf Stream is never explicitly asserted in the knowledge of the system. Similarly, the selection of the interaction regime is done on the RHS using procedural code rather than the pattern-matching capability of the system. The influence of modifying features such as the Gulf Stream or Isobath is never specified in working memory, so there is no record of how the new values were obtained. Each single rule in the current system really represents many rules as is evidenced by the large number of "if-then" constructions on the RHS of each rule. Because the knowledge of the system has been divided into such large "chunks" for the reasoning system, it is not possible with the current system structure to provide explanation of how the system reached its conclusions.

2.3 Recognize-Act Cycle

The recognize-act component used in the current system calls the *select* routine to decide which of the eligible rules from the conflict set should fire on each cycle. The conflict resolution routine is very simple because the limited use of pattern matching means that there are never very many rules that can fire at any one time. When more than one instantiated rule is eligible to fire, the rule that has most recently become eligible to fire is always selected.

2.4 Revisions Necessary for Porting to Sun

When the original system was ported to a Sun workstation, several changes were necessary. The graphics routines used to display the predicted state of the eddies and Gulf Stream on the VAX are not available on the Sun. These routines are in a file called graphrtn.c. The contents of this file were copied to the file dumygr.c and all routines were replaced with dummy routines (routines that did nothing but return a meaningless value).

- The C compiler on the Sun requires that the names of header files be complete (contain the .h extension) and be enclosed in <> or "" as appropriate.

- In racycle.ops, select() was renamed newselect() to prevent name conflict messages.

- Changed all =+ to +=.

- Removed & from &code when this was an argument in a scanf statement, because code is already an address (an array).

- In setup.ops, the statement &MSG = &EGS = 0b does not seem to work as expected in the OPS83 on the Sun. It was split into two assignment statements.

- In io.c, two functions, *remove* and *fgetname*, unique to VAX C are used. The function call *remove(fname)* was replaced with *unlink(fname)*. The results of the statement containing the function call *fgetname(filespec)* were not used, so the statement was commented out.

- Hard-coded path names were removed. This forces the data files to be in the same directory as the executable program.

- On all scanf and fscanf statements, the Sun requires that doubles be read with an *lf* format rather than the *f* format that is allowed in VAX C. These changes are critical in the routines that read in the Gulf Stream data.

2.5 Errors Corrected in the Original Version

In the process of reconciling results obtained with the original and revised knowledge bases, a few errors were discovered in the original version. It appears that each of these errors either did not affect the performance of the system or was in the code for rules that never fire because they apply to eddy movement in regions where the eddies rarely, if ever, occur. The following corrections were made:

In the second rule for warm core eddies in the first four regions, the local variables, &LAT, &LONG, and &RAD were used but never given values. It appears that the correct values were picked up fortuitously because of the similarity of the first and second rule for each of these regions.

In the first rule for warm core rings in region 2 (WCR2), the local variable &REG was stored in the ring element without ever being assigned a value. This bogus value for the region prevented the second rule from firing. Comments in the code state that WCRs should never occur in region 2.

In the second rule for CCRs in region 3 (CCR3GS), the value of the variables &DIR and &DIST are used as the basis for making decisions about interaction of rings with the Gulf Stream, but the variables are never assigned values. Code was inserted to assign values to these variables in exactly the same way it is done for other CCRs. The value of &DIR was also stored in the ring working element at the end of the CCR3GS rule as was done for all other CCRs.

3.0 Revised Structure of the Knowledge-Based System

The knowledge-based system has been substantially revised in order to facilitate the addition of an explanation capability. These revisions include changes to the working memory element definitions and other data structures, changes to both the warm and cold core eddy rules, and replacement of the conflict resolution procedure. It must be emphasized that these revisions have not affected the functionality of the system. The revised system produces exactly the same results as the corrected original version described above. The following design principles have been used as the basis for the revisions that have been made:

- Each rule in the knowledge base should represent one decision. This means that if-then statements should not be used on the RHS of rules if avoidable.

- All input/output should be removed from the rules.

- Rules should be as general as possible. Pattern matching and instantiation should be used to make a single rule applicable in as many situations as possible.

- The reasoning of the system should be reflected in rule firings and changes to working memory.

- Rules for cold core eddies and warm core eddies should use the same type of WMEs and the same sort of logic to the extent possible with the goal of using many of the same rules for both cold and warm core eddies.

- The values of WME attributes should not be overloaded with multiple meanings. For example, in the original system, a negative value for direction is used as a flag to indicate an encounter with the Gulf Stream on the previous iteration. In the new design, these two concepts, direction and Gulf Stream interaction, are represented by two different attributes.

- Knowledge should be represented in a form that corresponds to that used by human experts. This makes the system easier to maintain because it is easier to understand. It also simplifies the explanation task because "decompilation" of the knowledge for explanation purposes is no longer necessary.

3.1 Revisions to Working Memory Elements and Other Data Structures

The STATUS WME type in the original system has been replaced by a type called GOAL that is used in a similar fashion. The name was changed in order to avoid confusion with the status attribute of the WME representing rings. Again, there is only one element of type GOAL present in the system at any one time. Table 3 describes the fields in the GOAL type.

Table 3. GOAL element type definitions.

Field name	Type	Value description
ringtype	symbol	WCR or CCR, current type of ring under consideration
time	integer	current time step—intialized to 0 and incremented by STEPSIZE
refno	integer	refno of the ring currently under consideration. Initialized to 0 and changed when a ring is selected for updating. Set back to 0 when updating is complete for that ring.
region	integer	1-9, current region under consideration

The two types used to represent CCRs and WCRs in the original system have been replaced by a single type called RING with a field that specifies the ring type. An instance of a RING WME represents a description of a ring at a certain point in time. A series of these elements representing a single ring exists in working memory showing the progression of the ring through time. This facilitates both explanation and the development of more sophisticated heuristics for predicting the movement of rings based on their history. The status field indicates the results of processing for a ring. A ring element with a status of *active* indicates that this is the most recent instance representing the specific ring and that the ring is still active at this point in time. If the rules detect that a ring dissipates or coalesces with the Gulf Stream during a time step, then its status for that time step is set to *dissipated* or *coalesced*. If the ring does not dissipate or coalesce with the Gulf Stream on a time step, then the status field of the element representing the ring at the last time step is set to *processed* and a new RING element is created with the new time stamp, updated values for its reg, radius, lat, long, and dir fields, and a status value of *active*. Note that in the original system, a dissipated or coalesced ring was indicated by setting the value of its region field to 0. This is an example of overloading a field value with multiple meanings.

Table 4. RING element type definition.

Field name	Type	Value description
ringtype	symbol	value is WCR or CCR
refno	integer	unique identifier for ring
time	integer	time stamp for this ring description
reg	integer	1-9 region in which ring is located
radius	real	radius of ring in km
lat	real	latitude of ring in degrees
long	real	longitude of ring in degrees
dir	integer	1-16 representing compass direction, initialized to 13 for both WCR and CCR
status	symbol	possible values: active, processed, dissipated, and coalesced

In the revised system, region parameters are stored in WMEs instead of records so they can be referred to on the LHS of rules (Table 5). As in the case for rings, one type is used for the parameters for both CCRs and WCRs by defining a field in the element that specifies the ring type. In addition, individual attributes for break-point values have been replaced by an array of break-point values. Note that the last break-point field will not be used when the type is used to define the region parameters for a WCR. In the original system, direction of movement was expressed by latitude and longitude factors (dlat and dlong; these factors were direction cosines derived from angle values (Thomason, 1992). In the process of "tuning" the system, some of the factors had been modified so they no longer represented a "pure" directional component. Some of this tuning was probably necessary because of the inaccurate unit conversions used in the original version. The longitude and latitude factors have been "decompiled" into a pure direction component that is represented as a heading value between 0 and 360° with 0 as N. This change has two advantages. First, the heading value is more meaningful for explanation purposes. Second, the direction component of predicted movement is now represented by one value rather than two. In cases where the longitude and latitude values no longer represented true direction cosines, the latitude factor was used to back calculate the angle because latitude calculations were more accurate in the original version than longitude calculations.

Table 5. REGION element type definition.

Field name	Type	Value description
ringtype	symbol	value is WCR or CCR
speed	real	default speed of rings in region (cm/sec)
direction	integer	1-16 representing compass direction
heading	real	0-360°; default heading; 0 is N
decay	real	decay factor per week with no Gulf Stream interaction
minrad	real	minimum radius value in km
gsdec	real	additional decay factor per week with Gulf Stream interaction
bkpt[4]	real	break points for ratio test

An additional WME type called movedRING has been defined. An element of this type is created every time a ring is updated during a time step and is used as a place to store important intermediate results as the rules fire. Once computation for the ring is complete for a time step and its values have been stored in a new RING element, the movedRING element is deleted. In general, changes in the values of fields in the movedRING element direct the firing of rules. Table 6 describes the fields in the movedRING element type. The lat, long, and radius fields are the same as those from the RING WME and are stored again in the movedRING WME for convenient retrieval by the explanation component. The newradius, newlat, and newlong are used for WCRs to store the initial estimates for new values based on default values for decay, heading, and speed. The dlat and dlong attributes are used to store the changes imposed by interaction of the Gulf Stream with WCRs. The modlat, modlong, and modradius attributes are used to store the values that take Gulf Stream interaction into account. The previousencounter attribute replaces the negative direction flag used in the original version.

Table 6. movedRING element type definition.

Field name	Type	Value description
ringtype	symbol	value is WCR or CCR
refno	integer	unique identifier for ring
reg	integer	1-9 region in which ring is located
lat	real	latitude of ring at start of time step
long	real	longitude of ring at start of time step
radius	real	radius of ring at start of time step
newradius	real	estimated radius of ring in km
newlat	real	estimated latitude of ring in degrees
newlong	real	estimated longitude of ring in degrees
newregion	integer	1-12 new region
newheading	real	0-360°; heading used for calculation of new position on this time step
newdir	integer	1-16; general direction for next time step
dlat	real	change in latitude for WCR (percent of radius per week)
dlong	real	change in longitude for WCR (percent of radius per week)
radiusfactor	real	change factor for ring radius
distance	real	distance moved by ring during this time step in degrees latitude
ratio	real	ratio of newradius to gsdist
gsdist	real	distance from eddy to Gulf Stream in degrees latitude
modlat	real	modified latitude value after interaction with Gulf Stream has been calculated
modlong	real	modified longitude value after interaction with Gulf Stream has been calculated
modradius	real	modified radius after interaction with Gulf Stream has been calculated
modifications	symbol	Indicates status of calculation of modifications due to Gulf Stream. Possible values: nil, calculated, applied
previousencounter	logical	0B if no encounter with Gulf Stream at previous time step, 1B otherwise

mdp	integer	minimum distance projection—indicates direction from eddy to nearest point on Gulf Stream
speed	real	velocity of ring in cm/sec
inter	symbol	indicates strength of interaction with Gulf Stream. Possible values: nil (has not been calculated), strong, weak, medium, none
regime	integer	index based on inter and previousencounter used to retrieve parameter for Gulf Stream interaction
status	symbol	possible values: active, processed, dissipated, and coalesced

The original rules contain many hard-coded parameters that describe the motion of the rings when there are different degrees of interaction with the Gulf Stream. These parameters have been removed from the rules and placed in look-up tables. The rules retrieve the appropriate parameter values based on the region where the ring is located, the interaction regime that has been specified, and either the general direction of movement of the eddy, or the direction from the eddy to the Gulf Stream. A description of the interaction regimes and the motion parameters for each can be found in Lybanon (1990). The motion parameters that described movement for CCRs in the original version consisted of a latitude factor, a longitude factor, and a final direction. The latitude and longitude factors have been replaced by a single heading value as described for the REGION WME. The heading value is used to compute the direction of movement for the ring for the current time step. The final direction parameter indicates the general direction of movement for the ring on the next time step. A record structure called MotionParameter has been defined with fields for these heading and direction values.

The latitude and longitude adjustment factors for WCRs are not direction cosines and so could not be decompiled into a heading value. A record structure called AdjustParameter has been defined for the WCR latitude and longitude adjustment factors.

The look-up tables for WCRs and CCRs are implemented as three-dimensional arrays where each element is an AdjustParameter or MotionParameter, respectively. The first index in each array is the region number, the second is the interaction regime, and the third is the direction. There are two possible interaction regimes for WCRs and four for CCRs. For WCRs, regime 1 is for strong Gulf Stream interaction and regime 2 is for medium Gulf Stream interaction. For CCRs, regime 1 is for weak Gulf Stream interaction, regime 2 medium Gulf Stream interaction, and regimes 3 and 4 strong Gulf Stream interaction. The third is used if there was no encounter with the Gulf Stream on the previous time step, and the fourth is used if there was an encounter on the previous time step. Appendix A contains a format description of the file containing all of the adjustment and motion parameters.

3.2 Revised Knowledge Base

The structure of the rules has undergone the most notable transformation. As in the original system, there are some rules in the revised system that apply only to cold core eddies and some that apply only to warm core eddies. In addition, there are some rules that are used for both types of eddies. The rules are activated in the same way they were in the original system with a GOAL WME controlling the activity instead of a STATUS WME. A GOAL WME is asserted into working memory with the time specified as the current time step, the current region, and the refno set to 0 (this indicates that no ring has been selected). Appendix B contains a listing of the WCR rules and Appendix C a listing of the CCR rules. The rules that apply to both types of eddies are listed with the CCR rules. The solution strategies for predicting the movement of the two types of eddies differs significantly, making it necessary to use two different sets of rules to determine the new position and radius. The general solution strategy implemented for each type of eddy is described below. For both types of eddies, there are sets of rules that are used at different stages of the prediction process. Figure 2 shows the rule sets for warm and cold core eddies and the order of their activation for a single ring.

Warm Core Rule Sets

{WCREstimateMotion}
↓
{WCRGetInteraction}
↓
{WCRStrongInteraction,
WCRMediumInteraction,
WCRWeakInteraction,
WCRNoInteraction}
↓
{WCRModifyStrongMedium,
WCRModifyWeak,
WCRNoModification}
↓
{WCRApplyModifications}
↓
{WCRRegion2SouthernLimit,
WCRRegion2CoastalLimit,
WCRRegion3ShelfBreakLimit,
WCRRegion4Isobath,
WCRRegion7ShelfBreak,
WCRRegion8ShelfBreak,
WCRRegion1DoNotHappen}
↓

Cold Core Rule Sets

{CCREstimateMotion}
↓
{CCRStrongInterWEncounter,
CCRStrongInterNoEncounter,
CCRMediumInteraction,
CCRWeakInteraction,
CCRNoInteraction}
↓
{CCRModifyStrong,
CCRModifyMedium1,
CCRModifyMedium2,
CCRModifyMedium3,
CCRModifyMedium4,
CCRModifyWeak,
CCRNoModification}
↓
{CCRApplyModifications}
↓

↓
{NewRegion,
CheckDissipation,
CheckCoalescence}
↓
{StoreNewPosition,
StoreNonActive}

Figure 2. Rule Sets for WCRs and CCRs and their Order of Selection.

3.2.1 Warm Core Ring Rules

When a GOAL WME is asserted with ringtype WCR for a region that contains at least one active warm core eddy, the WCREstimateMotion rule will fire and select an active warm core eddy to update. The RHS of this rule uses the region defaults for speed, heading, and decay rate to estimate a new latitude, longitude, and radius for the ring. A movedRING element is created for the ring, and these values are stored in the newlat, newlong, and newradius fields. The GOAL refno is set to the refno of the current ring. This prevents the WCREstimateMotion rule from firing with any other active warm core eddies in the region until updating on the selected ring is complete. Next, the WCRGetInteraction rule will fire and determine the interaction ratio for the movedRING. This ratio is used to decide which of four InteractionRules will fire to classify the interaction as either strong, medium, weak, or none. If the interaction is classified as strong, then the regime is set to 1; if the interaction is classified as medium, then the regime is set to 2. Another set of rules (modification rules) now comes into play to determine how the estimated latitude, longitude, and radius should be modified based on the strength of the interaction. These rules store the modified values into the *modlat, modlong,* and *modrad* fields and set the *modification* field to *applied.* Since the calculations for strong and medium interaction are performed identically but using different parameters, one rule handles both of these interactions. Weak interaction only modifies the radius value. An interaction strength of none causes the unchanged values to be stored in the appropriate fields.

Once the modifications have been applied, a set of constraint rules is eligible to fire if the modified values violate constraints for specific regions. These rules implement the influence of the continental shelf, coastlines, isobath barriers, etc. In the original system, these constraint calculations were performed after the estimation of a new position, but before modification of position by the Gulf Stream was calculated. This occasionally resulted in the modification rules "undoing" the effect of the constraint rules.

After any relevant constraint rules have fired, a set of rules applicable to both WCRs and CCRs are eligible to fire. These rules determine the region for the new location (NewRegion), determine if the new radius value has dropped below the minimum for the region (CheckDissipation), and thus determine if the ring has coalesced with the Gulf Stream (CheckCoalescence). After all of these checks are complete, the new values are stored by one of two rules: StoreNewPosition for rings that are still active, or StoreNonActive for rings that have dissipated or coalesced with the Gulf Stream.

3.2.2 Cold Core Ring Rules

The CCR rules operate in a manner very similar to those for WCRs but differences in the way the new position is calculated make it necessary to have two separate sets of rules. The first rule activated for CCRs is the CCREstimateMotion rule. Where the rules for WCRs estimate the new radius and position and then modify these values, the first rule for CCRs only estimates the new radius. The old values for latitude and longitude are used to determine the distance to the Gulf Stream. This distance and the new radius value are used to calculate an interaction ratio. After the interaction ratio has been determined, one Interaction rule fires and asserts the strength of interaction with the Gulf Stream and the appropriate interaction regime. This decision activates one Modification rule that determines how the speed, direction, and radius decay rate are affected by the Gulf Stream. The CCRApplyModifications rule can then fire and assert the new position based on the direction and speed. The new radius value is modified to reflect any additional decay due to interaction with the Gulf Stream. The last two groups of rules for CCRs are identical to those for WCRs.

3.3 Recognize-Act Cycle

The revised rules make much greater use of pattern matching than the rules in the original version did and require a more sophisticated conflict resolution routine. The revised system uses the *select* routine supplied with OPS83 which performs conflict resolution based on the recency of the rule activation, the specificity of rule patterns, and rule priorities. Details of the conflict resolution algorithm are described in Forgy (1989).

4.0 Performance Comparison of Original and Revised Systems

The goal was to redesign the architecture of the rule base of the original system to facilitate explanation without changing its functionality. The two systems were run with two different data sets (see Appendix D for listings of the input files) and the predictions of the systems over a 28-day period were compared. The first data set represents real data and had three warm core eddies and five cold core eddies. This data set was tested one time using a nominal Gulf Stream and one time using the Gulf Stream files given in Appendix D. The second data set is artificial data designed to test the performance of the system in each of the 9 regions for each type of eddy. The locations and size of the warm core eddies cause each of the constraint rules for warm core eddies to fire. Eddies were also included that should dissipate during the prediction period, coalesce with the Gulf Stream, and move from one region to another. This data set was tested on both the original and revised versions with the nominal Gulf Stream and with the Gulf Stream files given in Appendix D. The revised system was developed in several phases. In the first phase, the rule structure was revised to that described in the previous section, but the latitude and longitude factors were not translated to headings, and the new geometry routines were not included. With this version, the predicted movement of the eddies was identical for both versions for all data sets tested. In the next phase, the latitude and longitude factors were translated to headings (except for Gulf Stream interaction with WCRs), and improved geometry routines were called for calculating the new position of the rings (see Section 5).

Although the original and revised rule architectures produce the same results, there are many differences in the rules of the two systems. The rule design in the old system was very uniform. There were 2 rules per eddy type per region for a total of 36 rules. In the new system, the rules are more general. Each rule makes only one decision but is applicable in many situations. There are a total of 34 rules in the new system (15 warm core rules, 14 cold core rules, and 5 rules used for both eddy types).

The size and complexity of the rules has been greatly reduced in the revised system as shown in Table 7. The revised rules show a modest increase in the number of patterns on the LHS, and a dramatic decrease in the number of actions performed on the RHS. The actions on the RHS of the revised rules consist almost entirely of numerical calculations and changes to working memory.

Table 7. Comparison of Original and Revised Rule Sets.

Complexity Measure	Original	Revised
Average number of patterns on LHS	5	6.5
Maximum number of patterns on LHS	5	9
Average number of actions on RHS	119	3.3
Maximum number of actions on RHS	180	11

The number of rules that fire in order to make a decision has also changed. In the original version, two rules would always fire to predict the movement of one eddy for one time step. In the revised system, the number of rule firings is not as predictable and depends on the conditions. On the first data set described above, the average number of rule firings with the revised system was 6.6 per ring per time step.

The Unix *gprof* profiling utility was used to compare the time requirements of the two systems using the second data set with the nominal Gulf Stream, the minimum output, and an elapsed time of 35 days. Unexpectedly, the revised version was slightly faster than the original version, although this is probably not significant. An analysis of the time spent in different routines revealed that for both versions, well over 90% of the total time was spent in math routines (sqrt and trigonometric functions) that are called to process the Gulf Stream. This indicates that the change in rule structure has had a minimal effect on efficiency.

5.0 Modified Geometry Routine

Matthew Lybanon has developed a C routine to calculate changes in latitude and longitude from the current latitude, latitude and longitude factors, the velocity of the ring, and the elapsed time. This routine addresses several problems in the original system. The routine has been adapted for use in the revised version of the expert system. Appendix E gives a listing of the geometry routines and the names of the specific rules where each is invoked. A comparison of the results of running the system with the data set described above (nominal Gulf Stream) with the new and old geometry routines is given in Figure 3. As expected, the differences are much more pronounced for longitude calculations, since the previous implementation neglected the difference in size of latitude and longitude degrees, as well as the variation of the latter with latitude.

Ring	New geometry		Old geometry		Difference	
Time 7	Latitude	Longitude	Latitude	Longitude	Latitude	Longitude
WCR1	39.5433	71.2168	39.5503	71.0814	-0.007	0.1354
WCR2	39.9746	67.9310	39.9816	67.7939	-0.007	0.1371
WCR3	40.4291	63.7969	40.4436	63.6649	-0.0145	0.1320
CCR1	35.2902	71.6026	35.2667	71.4887	0.0235	0.1139
CCR2	36.5902	65.6093	36.5667	71.4887	0.0235	0.1206
CCR3	37.7542	59.8893	37.797	59.9857	-0.0428	-0.0964
CCR4	38.1717	57.1437	38.1752	57.0367	-0.0035	0.107
CCR5	39.2109	54.5000	39.2714	54.5000	-0.0605	0.000

Figure 3. Comparison of Results with Old and New Geometry Routines.

6.0 Correction of Rate Calculations

Many calculations in the original expert system were not sensitive to the variable STEPDAYS, which specifies the length of time for which a prediction is being made. All calculations that deal with rate should include the step size as a factor. The appropriate changes were made to calculations of radius decay and the modification of position calculations for WCRs. The modifications were based on the assumption that rates in the original system were for a time step of 7 days (Thomason, 1992).

7.0 Explanation Component

When people use the term "explanation" they are referring to a wide variety of activities (Swartout, 1991). Explanations produced by expert systems attempt to model a subset of these capabilities. The types of explanations that can be produced by an expert system depend on the anticipated needs of the user, the types of knowledge that are represented in the system, how that knowledge is used by the system, and the goal of the system developer in providing explanation.

Typically, an explanation by an expert system describes *how* the system reached its conclusions based on its knowledge. If sufficient causal and strategic knowledge is incorporated in the system, then the explanation component can also explain *why* a particular reasoning process was used. In general, *how* explanations are much easier to provide than *why*. Note that this is not the same as the *why* capability included in many exper- systems that refers to the system's ability to explain why a question is being asked, not why the reasoning process is valid.

The goal of adding an explanation capability to the oceanographic expert system was to give the users access to the reasoning process of the system. For this purpose, a facility has been added to the system that can provide explanations of *how* the system reached its conclusions. The design and implementation of this facility will be discussed.

The explanation component that has been implemented consists of two modules (Figure 4). The Introspection module monitors the inference engine and records rule firings—both the patterns matched on the LHS and the changes that the rule makes to working memory when it fires. These rule instantiations are stored in sequential order of firing for each eddy for each time step. This type of information could not easily be collected with the original version of the expert system where most decisions were made in procedural code on the RHS of the rules and the process was not recorded in working memory.

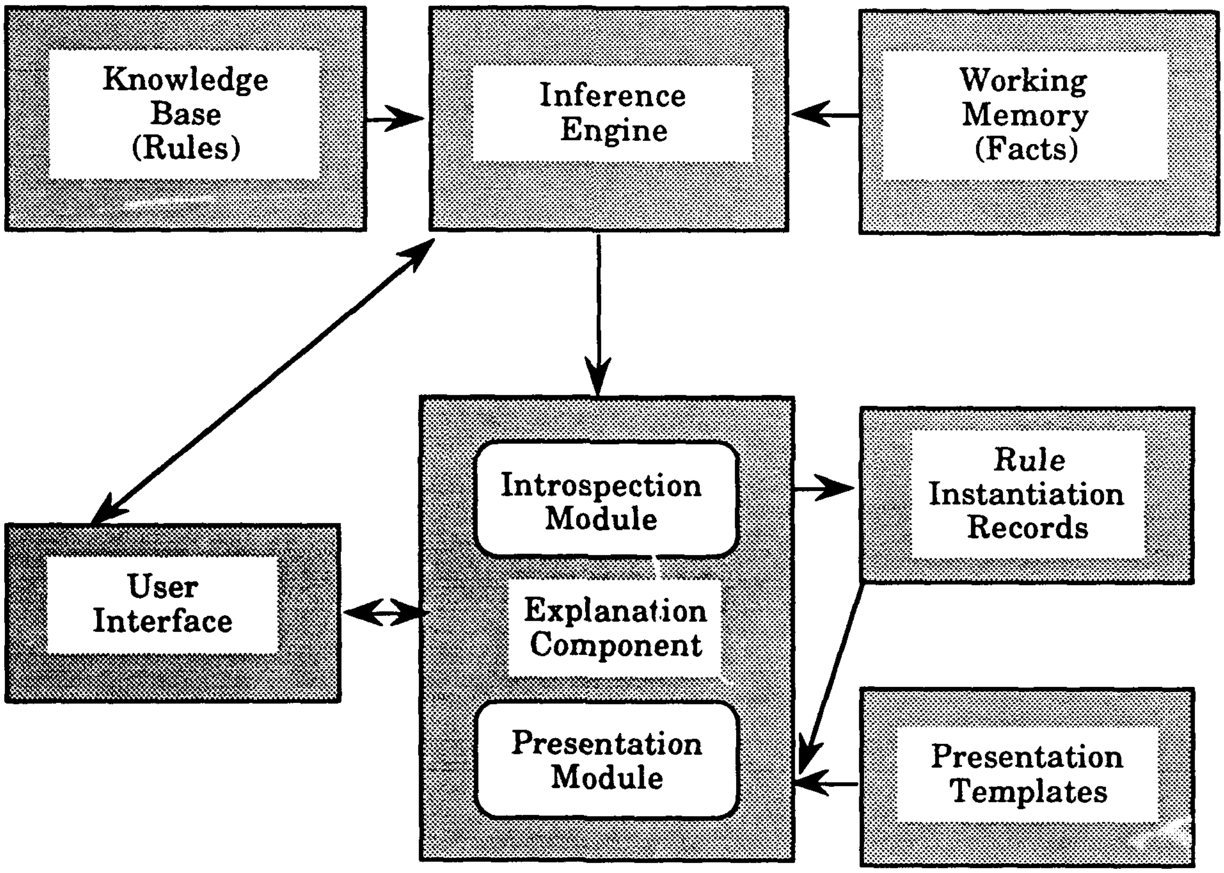

Figure 4. Explanation Component of Expert System.

The Introspection module records the name of each rule before it fires. After the rule firing the Introspection module uses the name of the rule to invoke the appropriate procedure to extract critical attribute-value pairs. These values come from the patterns that were matched when the rule was instantiated and the changes that the rule makes to working memory. The procedures for extracting the values for each rule are written in OPS83.

At the completion of an expert system session (when "q" is selected), the Presentation module will be invoked if explanation is requested. The user can ask for an explanation of the predicted movement for a single eddy or for all eddies. Two explanation options are available. One is for a detailed natural language trace of the rules (the Rule Trace option), and the other is for a summary of the reasoning process (the Summary option).

7.1 Rule Trace Option

When the Rule Trace Option is requested, the Presentation module generates a Rule Trace in a straight forward way from the record of rule instantiations and a set of rule templates. For each rule in the system, the system designer must provide a natural language rule template. Figure 5 gives an example of a rule template and a complete listing of all templates is given in Appendix F. Each template consists of the name of the rule enclosed in vertical bars and then the body of the template enclosed in vertical bars. The words beginning with a question mark are variables that correspond to the name of a variable in a variable-value pair is recorded by the Introspection module for that rule. The rule templates are read from a file and stored in memory so they can be changed without recompiling the system.

```
|WCREstimateMotion|
|If the goal is to process a WCR in region ?reg at time ?time
  and WCR ?refno is active in region ?reg at latitude ?lat longitude ?long
  with radius ?radius km
 Then
   Using a default speed of ?speed cm/sec, WCR ?refno will travel
   a distance of ?distance degrees to a new position of latitude ?newlat
   longitude?newlong.
   Using the default decay rate of ?decay % per week, the new radius
   will be ?newradius km.|
```

Figure 5. Explanation of a Rule Template.

If the user requests a rule trace for a single eddy, then the Presentation module locates the list of instantiations for that eddy for each time step. It sequentially processes each list of rule instantiations, retrieving the appropriate template for each rule. A text string is produced from the template with all variables replaced by the corresponding values recorded in the instantiation record. The series of filled templates corresponding to each rule instantiation is passed to the user interface for presentation to the user. Figure 6 gives an example of a rule trace for one eddy for one time step.

****

Rules firing for WCR 1 at time 0

****

Rule: WCREstimateMotion
If the goal is to process a WCR in region 4 at time 0
and WCR 1 is active in region 4 at latitude 39.60 longitude 70.80
with radius 75.00 km
Then
Using a default speed of 6.00 cm/sec and heading of 260.00 degrees,
WCR 1 will travel a distance of 0.33 degrees to a new position
of latitude 39.54 longitude 71.22.
Using the default decay rate of 1.00% per week, the new radius
will be 74.25 km.

Rule: WCRGetInteraction
If the motion of WCR 1 has been estimated
Then
the interaction ratio with the Gulf Stream based on the
new location and new radius is calculated to be 0.38.

Rule: WCRNoInteraction

If the interaction ratio of WCR 1 is less than 0.50
Then
the interaction is classified as none.

Rule: WCRNoModification
If WCR 1 did not interact with the Gulf Stream
Then
the estimated latitude, longitude, and radius are not modified.

Rule: WCRApplyModifications
If the modification parameters for WCR 1 have been determined
Then
The adjustment factors for latitude, longitude, and radius were
applied to give modified values of latitude 39.54, longitude 71.22,
and radius 74.25 km

Rule: CheckCoalescence
If WCR 1 motion calculations are complete
Then
the distance to the Gulf Stream was calculated and the ring has
not coalesced with the Gulf Stream.

Rule: NewRegion
If the motion calculations for WCR 1 are complete
Then
the new region is determined to be 4.

Rule: StoreNewPosition
If all calculations for WCR 1 are complete
and the eddy is still active
Then
store the new position (latitude 39.54 longitude 71.22)
and radius (74.25 km) at time 7.

Figure 6. Rule Trace Generated by Explanation Component.

The rule trace provides a very detailed account of how the system has reached its conclusions. This type of trace is useful for debugging purposes and for communicating the system's reasoning to scientists working on the project. These "explanations" vividly illustrate the limitations of using simple rule traces for explanation. Although the rule trace provides an explanation of the system's reasoning that is much superior to that of typical procedural programs, it does not approach the sort of capability that comes to mind when people talk of explanation. Rule trace explanations are useful during system development to demonstrate how the heuristics have been implemented in the rule-based format. This type of explanation would, however, be of little value for users of the system on a daily basis. The explanations have little structure and include much extraneous information that is needed to direct the firing of the rules, but conveys little information about how the rules work. It is obvious that a "summarization" capability is needed to compress and organize the explanations.

7.2 Summary Option

The goal of producing concise summaries of the system's reasoning process coupled with an analysis of shortcomings of the rule traces led to the compilation of a set of summarization heuristics. These heuristics are used to produce text that is better organized, emphasizes important information, and omits extraneous information. The heuristics are as follows:

- *Highlight problem solving strategies*
  Rules traces have little inherent structure and tend to obscure the problem solving strategy. An organization needs to be imposed on the flat structure.

- *Use words parsimoniously*
  Natural language should emphasize critical information, not distract from it.

- *Do not repeat information*

Many rules use the same information to trigger firing. The information only needs to be presented one time.

- *Do not state the obvious*
In some cases, one rule will calculate a value and another rule will store the value. The reader will assume that a calculated value will be stored unless told otherwise.

- *Do not mention a lack of change*
A lack of change is usually not interesting. For example, a rule will always fire to determine which region an eddy is located in after it has moved. In cases where the region does not change, it should not be mentioned.

- *Emphasize special behavior*
When an eddy interacts with the Gulf Stream, its velocity will be higher than usual. This velocity will be more meaningful if it is contrasted to the default velocity.

- *Use understandable terms*
In some cases the representation used by the expert system is not easily understood by the user and should be translated into a more useful form. For example, compass directions are represented internally using integers.

Implementation of these heuristics has been accomplished by a variety of mechanisms. The summaries are generated using a template structure similar to that used for rule traces. The filled template for each rule is presented as a step in the problem solving process. If several actions are taken by one rule, these are presented as substeps. The result is a summary in outline form with short statements of the problem solving steps. Figure 7 shows the summary that corresponds to the Rule Trace given in Figure 6. The parsimonious use of words was accomplished by the wording chosen for the summary templates.

Rule filtering is used to control the inclusion of a rule summary as a step. Some rules always perform mundane tasks that never need to be expressed. These rules are filtered out by the use of an empty template. There are other rules that sometimes perform interesting tasks and sometimes do not. For example, the NewRegion rule only needs to be expressed if the eddy moves to a new region, and the CheckCoalescence rule only needs to be expressed if the eddy coalesces with the Gulf Stream. Conditional filtering by the Presentation module determines when these rules should be included as steps in the summary.

****
WCR 1 at time 0
****

1. Time 0 status: latitude 39.60 longitude 70.80 radius 75.00 km.
   Region 4 defaults were used to estimate new position and radius:
   -- Speed = 6.00 cm/sec.
   Heading = 260.00°
   Gives new position: latitude 39.54, longitude 71.22.

   -- Default decay rate = 1.00% per week
   Gives a new radius: 74.25 km.

2. The new position is 1.52° from the Gulf Stream.
   Gulf Stream interaction ratio calculated:
   ratio = (new radius)/(distance to Gulf Stream) = 0.38

3. There was no interaction with the Gulf Stream because the ratio was less than 0.50.

4. Status at time 7: latitude 39.54 longitude 71.22 radius 74.25 km.

Figure 7. Summary Generated by Explanation Component.

For rules that perform a series of actions, some of the actions may be interesting while others are not. For example, the WCRModifyStrongMedium rule sometimes changes the estimates of the radius, latitude, and longitude and in other cases only changes one or two of these values. An "if-interesting" mechanism has been added to the template structure to allow simple tests that can trigger the inclusion of some substeps. Figure 8 gives an example of a summary template that uses this mechanism for three substeps. The optional phrase is enclosed in curly brackets and the first three symbols after the open bracket are required to represent a variable, a test, and a value in that order. The variable and value must correspond to information collected by the Introspection Module. The test is for either equality (=) or inequality (!). If the test evaluates to true, then the remainder of the template inside the curly bracket is expressed, otherwise it is omitted.

|The GS is ?dir of the eddy and the ?inter interaction causes
{?dlat ! 0.00 --the ring to be pushed ?dlatupdown a distance that is ?absdlat % of radius per week
}{?dlong ! 0.00 --the ring to be pushed ?dlongupdown a distance that is ?absdlong % of radius per week
}{?x ! 1.00 --the radius to decay an additional ?x % per week. }|

Figure 8. A Summary Template with if-interesting Phrases.

****
WCR 3 at time 0
****

1. Time 0 status: latitude 40.40 longitude 63.40 radius 95.00 km.
   Region 6 defaults were used to estimate new position and radius:
   -- Speed = 6.00 cm/sec.
   Heading = 248.00°
   Gives new position: latitude 40.28, longitude 63.80.

   -- Default decay rate = 0.20% per week
   Gives a new radius: 94.81 km.

2. The new position is 0.74° from the Gulf Stream.
   Gulf Stream interaction ratio calculated:
   ratio = (new radius)/(distance to Gulf Stream) = 0.98

3. The Gulf Stream interaction is classified as strong if the ratio is greater than 0.90.

4. The Gulf Stream is S of the eddy and the strong interaction causes
   --the ring to be pushed N a distance that is 20.00% of radius per week
   --the radius to decay an additional 3.00% per week.

5. Modifications were applied to give a new position of latitude 40.45 longitude 63.80.
   The ring radius was adjusted to 91.97 km.

6. Status at time 7: latitude 40.45 longitude 63.80 radius 91.97 km.

Figure 9. Summary for Interacting Eddy.

Figure 9 is an example of a summary produced by the explanation component for an eddy that interacts with the Gulf Stream. Step 4 of Figure 9 shows how the if-interesting mechanism is used to filter out expression of a change in longitude of 0. Only the change in latitude and radius decay steps are expressed. The emphasis on special behavior and the use of understandable terms are accomplished by having the Introspection module retrieve relevant information and the Presentation module express it in an appropriate form.

Appendix F gives a complete listing of the templates used for both the rule trace and summaries. A description of each of the source files that implement the explanation component and the data files that it uses can be found in Appendix G.

8.0 Conclusions

Although explanation capability is often cited as one of the unique characteristics of expert systems, it may not be trivial to add an explanation component to an existing expert system. Explanation is possible only to the extent that the reasoning of the system is observable. In general, knowledge that is represented declaratively is more easily observed than that which is represented procedurally. In a forward chaining rule-based system, the if-then format is used to represent knowledge declaratively, and the reasoning process is observed by "watching" the pattern matching and rule-firing activities of the inference engine. The knowledge base of the oceanographic expert system was revised so that decisions previously made on the RHS of rules in procedural code are now accomplished by assertions into working memory that trigger a sequence of rule firings. The revisions have not only facilitated the construction of an explanation component, but have also resulted in rules that are more general and easier to understand and modify. This is illustrated by the ease with which new geometry calculations were added to the revised rules.

The explanation component consists of an Introspection module that captures the chain-of-rule firings and a Presentation module that can use this information to produce either a detailed rule trace or a summary of the reasoning process. The rule trace is useful for debugging purposes and for demonstrating how the oceanographic heuristics have been implemented in the rule base. The summaries provide critical information in a brief, coherent form and are useful for analysts who want to know the basis for the system's predictions.

There are plans to recode the oceanographic expert system in the CLIPS expert system shell to facilitate incorporation into the TESS system (Phegley et al., 1991; Lybanon, 1992). The Presentation module of the explanation component is written in standard C and should require no modification. The Introspection module, however, is dependent on the inference engine and will need to be rewritten to retrieve the necessary information from the CLIPS system.

The summaries produced by the current explanation system are quite compact for one eddy for one time step. If, however, the user wants a summary of the reasoning process for all eddies over several time steps, the summaries are very lengthy. This problem provides and interesting domain for studying temporal and spatial summarization.

The explanations produced by the oceanographic system describe how the system makes a particular prediction, but not why the method for making the prediction is valid. In particular, the expert system is highly parameterized and there is no facility for recording or presenting a justification of the parameter values. When the expert system was first constructed, there was a scarcity of knowledge available for derivation of the parameters, and many of them are based on very few data sets or some vague statements made in the oceanographic literature (Thomason, 1992; Lybanon, 1991a). The Naval Oceanographic Office (NAVOCEANO) currently has data products that could be used to derive more accurate parameters for the heuristics. In addition, the analysts at NAVOCEANO who are responsible for compilation of this data would be excellent sources of expertise to help improve the heuristics of the system. The development of an approach for representing and presenting the knowledge needed to explain the source of the parameter values is a fertile research area.

9. References

Forgy, C. L. (1989). *The OPS83 Report, System Version 3.0.* Production Systems Technologies, Inc., p. 49.

Hurlburt, H. E., A. J. Wallcraft, Z. Sirkes, and E. J. Metzger (1992). Modeling of the global and pacific oceans: on the path to eddy-resolving ocean prediction. *Oceanography* v. 5, n. 1, pp. 9-18.

Lybanon, M. (1992). Oceanographic Expert System: Potential for TESS(3) Applications. Naval Research Laboratory, Stennis Space Center, MS, NOARL Technical Note 286.

Lybanon, M. (1991a). Personal communication.

Lybanon, M. (1991b). Recent results in Oceanographic Expert System Validation. In *Proceedings: Automated Interpretation of Oceanographic Satellite Images Workshop*, February 1991, Naval Oceanographic and Atmospheric Research Laboratory, Stennis Space Center, MS, pp. 49-56.

Lybanon, M. (1990). Oceanographic Expert System Validation Using GOAP Mesoscale Products and Gulfcast/Dart Validation Test Data. Naval Oceanographic and Atmospheric Research Laboratory, Stennis Space Center, MS, NOARL Report 5.

Phegley, L. and C. Crosiar (1991). The third phase of TESS. *Bulletin American Meteorological Society* v. 72, n. 7, pp. 954-960.

Swartout, W. R., C. L. Paris, and J. D. Moore (1991). Design for Explainable Expert Ststems. *IEEE Expert* v. 6, n. 3, pp. 58-64.

Thomason, M. G. (1992). Personal communication.

Thomason, M. G. (1989). Knowledge-Based Analysis of Satellite Oceanographic Images. *International Journal of Intelligent Systems* v. 4, pp. 143-154.

Thomason, M. G. and R. E. Blake (1986). *Development of an Expert System for Interpretation of Oceanographic Images.* Naval Ocean Research and Development Activity, Stennis Space Center, MS, NORDA Report 148.

Appendix A

Format for parms.dat

The parms.dat file contains many of the parameters that were formerly hard-coded in rules. The file contains a set of parameters for each region for each type of eddy. The parameters for warm core eddies for all regions are given first followed by parameters for the cold core regions for each region. For each region the warm core ring parameters have one line containing the region number followed by one line for each direction (1-16) containing the following values:

1. direction
2. regime1 latitude factor
3. regime 1 longitude factor
4. regime 2 latitude factor
5. regime 2 longitude factor

The cold core ring parameters have one line containing the region followed by one line for each direction (1-16) containing the following values:

1. general direction
2. regime 1 heading
3. regime 2 heading
4. regime 2 final direction
5. regime 3 heading
6. regime 3 final direction
7. regime 4 heading
8. regime 4 final direction

Appendix B
Warm Core Ring Rules

```
rule WCRModifyStrongMedium
--Calculate modified location and radius for strong or medium interaction
{
        &goal ( GOAL        ringtype = WCR;);

        &reg ( REGION       ringtype = WCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg   = &goal.reg;
                        modifications = nil;
                        ((&mring.inter = strong) \/
                         (&mring.inter = medium)););
-->
        local   &RADFAC:   real,
                &DLAT:     real,
                &DLONG:        real;


        &DLAT =
                &WCRparms[&goal.reg][&mring.regime][&mring.mdp].LatAdjust;
        &DLONG =
              &WCRparms[&goal.reg][&mring.regime][&mring.mdp].LongAdjust;
        &RADFAC = &reg.gsdec - 0.01; --fastest decay

        modify &mring (
                        dlat        = &DLAT;
                        dlong       = &DLONG;
                        radiusfactor  = &RADFAC;
                        modifications = calculated;
                        );

}; --end WCRModifyStrongMedium

rule WCRModifyWeak
{
        &goal ( GOAL        ringtype = WCR;);

        &reg ( REGION   ringtype = WCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg   = &goal.reg;
                        (&mring.inter = weak);
```

```
                    modifications = nil;);
-->
        local       &RADFAC:   real;

        &RADFAC = &reg.gsdec; --faster decay

        modify &mring (   radiusfactor = &RADFAC;
                    dlat       = 0.0;
                    dlong      = 0.0;
                    modifications = calculated;
                    );
}; --end WCRModifyWeak

rule WCRNoModification
{
        &goal ( GOAL        ringtype = WCR;);

        &reg ( REGION   ringtype = WCR;
                    reg = &goal.reg;);

        &mring ( movedRING
                    ringtype = WCR;
                    reg  =    &goal.reg;
                    inter =    none;
                    modifications = nil;);
-->
        modify &mring (   dlong = 0.0;
                          dlat  = 0.0;
                          radiusfactor = 1.0;
                          modifications = calculated;
                    );

}; --end WCRNoModification

rule WCRApplyModifications
--Calculate new location based on values for dlat, dlong, and speed
{
        &goal ( GOAL        ringtype = WCR;);

        &reg ( REGION   ringtype = WCR;
                    reg = &goal.reg;);

        &mring ( movedRING
                    ringtype = WCR;
                    reg  =    &goal.reg;
                    modifications = calculated;
                    );

        &ring ( RING
```

```
                ringtype = WCR;
                refno = &mring.refno;
                status = active);

-->
    local  &STEP:     real,
       &DLAT:     real,
           &DLONG:    real,
           &LAT:      real,
           &LONG:     real,
           &LATSTEP:  real,
           &LONGSTEP: real,
           &DIST:     real,
           &DIR:            integer,
           &RAD:      real;

    --Assume dlat and dlong %radius moved per 7 days S. Bridges 7/31/92
    --Make appropriate adjustments for &STEP
    --&DLAT and &DLONG are distances in km, where &DLAT is distance
    --  moved due north and &DLONG is distance moved due west.
    cast &STEP = &STEPDAYS;
    &DLAT =  (&mring.dlat * (&STEP / 7.0)) * &mring.newradius;
    &DLONG = (&mring.dlong * (&STEP / 7.0)) * &mring.newradius;
    &LAT   = &mring.newlat;

    --call adjustringstep to get change in lat and long in degrees.
    call adjustringstep(&LATSTEP, &DLAT,&LONGSTEP, &DLONG, &LAT);
    &LAT =  &mring.newlat + &LATSTEP;
    &LONG = &mring.newlong + &LONGSTEP;

    --decay rates are weekly, apply additional decay for gs interaction
    &RAD =  &mring.newradius *(1.0 - ((1.0 - &mring.radiusfactor) *
                       (&STEP / 7.0)));

    modify &mring (   modlat = &LAT;
                modlong = &LONG;
                modradius = &RAD;
                modifications = applied;
                distance = &DIST;
                );

}; --end WCRApplyModifications


--
*********************************************************************************
--Check any cons   ints on the movement of the eddy in that may occur in
-- some regions.
--Note that many of the eddies will still violate the constraints after
```

```
-- the indicated modifications.

rule WCRRegion2SouthernLimit(50)
--Prevent rings in region 2 from going too far south
{
        &goal ( GOAL      ringtype      = WCR;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg  =   2;
                        status = unknown;
                        modifications = applied;
                        (@.modlat < 36.0););
-->
 modify &mring (modlat = 36.0001);

}; --end WCRRegion2SouthernLimit

rule WCRRegion2CoastalLimit(50)
--Prevent rings in region 2 from going too close to coast
{
        &goal ( GOAL      ringtype      = WCR;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg  =   2;
                        status = unknown;
                        modifications = applied;
                        (@.modlong > 74.8););
-->
 modify &mring (modlong = 74.7999);

}; --end WCRRegion2CoastalLimit

rule WCRRegion3ShelfBreakLimit(50)
--N & W shelf break constraints for region 3--ring stays below sloped line
{
        &goal ( GOAL      ringtype      = WCR;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg  =   3;
                        status = unknown;
                        modifications = applied;
                        (@.modlong > ((101.0 - @.modlat)/0.86)););
-->
        local &DIST:  real,
            &LONG:  real,
            &MODLONG: real,
```

```
        &MODLAT: real;

    &MODLONG = &mring.modlong;
    &MODLAT = &mring.modlat;
    &DIST = (101.0 - &MODLAT)/ 0.86;
    &LONG = &DIST;

    modify &mring (modlong = (&LONG - 0.0001); status = constrained);

}; --end WCRRegion3ShelfBreakLimit

rule WCRRegion4Isobath(50)
--Insure no crossing of 200 m isobath
{
    &goal ( GOAL      ringtype      = WCR;);

    &mring ( movedRING
                ringtype = WCR;
                reg  =   4;
                status = unknown;
                modifications = applied;
                (@.modlat > 40.0););
-->
    local &LAT:  real;

    &LAT = 40.0;
    modify &mring (modlat = &LAT - 0.0001; status = constrained);

}; --end WCRRegion4Isobath

rule WCRRegion7ShelfBreak(50)
--Shelf break constraints
{
    &goal ( GOAL      ringtype      = WCR;);

    &mring ( movedRING
                ringtype = WCR;
                reg  =   7;
                modifications = applied;
                status = unknown;
                (@.modlat > 42.0););
-->
    local &LAT:  real;

    &LAT = 42.0;

    modify &mring (modlat = &LAT - 0.0001; status = constrained);

}; --end WCRRegion7ShelfBreak
```

```
rule WCRRegion8ShelfBreak(50)
--Shelf break constraints
{
        &goal ( GOAL        ringtype        = WCR;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg  =   8;
                        status = unknown;
                        modifications = applied;
                        (@.modlat > 43.2););
-->
        local &LAT:  real;

        &LAT = 43.2;

        modify &mring (modlat = &LAT - 0.0001; status = constrained);

}; --end WCRRegion8ShelfBreak

rule WCRRegion1DoNotHappen
--WCR rings should not appear in region 1--mark any that occur as dissipated
{
        &goal ( GOAL        ringtype        = WCR;);

        &mring ( movedRING
                        ringtype = WCR;
                        reg  =    &goal.reg;
                        modifications = applied;
                        status = unknown;
                        newregion = 1;);
-->
        modify &mring (status = dissipated);

}; --end WCRRegion1DoNotHappen


}; --end of module
```

Appendix C
Cold Core Ring Rules

```
module ccrules()
{
use eddytypes;
use regions;

-- The start procedure in module eddies cycles through all regions
-- asserting GOAL with refno = 0 and ringtype = CCR.
-- CCREstimateMotion selects an eddy in the region to update if there is one
-- and does initial estimates of radius and gets GS interaction info

rule CCREstimateMotion
-- locate a ring in this region that needs to be moved and calculate
--    distance to GS and new radius. Interaction ratio with GS is
--    calculated as (ring radius)/(distance to GS). Higher values
--    correspond to stronger interaction.
{
        &goal ( GOAL        ringtype = CCR;
                            refno = 0);

        &ring ( RING        ringtype = CCR;
                            time   = &goal.time;
                            reg    = &goal.reg;
                            status  = active;);

        &reg ( REGION       ringtype = CCR;
                            reg = &goal.reg;);
-->
        local  &LAT:        real,
               &LONG:       real,
               &RAD:        real,
               &DIR:        integer,
               &ENC:        logical,
               &STEP:       real,
               &RATIO:      real;

        cast &STEP = &STEPDAYS;
        &LAT =  &ring.lat;
        &LONG = &ring.long;

        --decay rates are weekly
        &RAD =  &ring.radius * (1.0 - ((1.0 - &reg.decay)*(&STEP / 7.0)));

        --find CCR-GSpoints distance
        call RtoGS2(&LGSRAY, &GSDIST, &lcntgs, &LAT, &LONG);
        cast &DIR = &GSDIST[3];
        &RATIO = &RAD / (FACTOR2 * &GSDIST[2]);
```

```
        if (&ring.dir < 0)
              &ENC = 1b
        else
              &ENC = 0b;

        make (movedRING
              ringtype =   CCR;
              refno    =   &ring.refno;
              reg      =   &ring.reg;
              lat      =   &ring.lat;
              long     =   &ring.long;
              radius   =   &ring.radius;
              newradius=   &RAD;
              inter    =   nil;
              regime =     0;
              modifications = nil;
              ratio    =   &RATIO;
              dir =        abs(&ring.dir);
              previousencounter = &ENC;
              mdp =        &DIR;
              gsdist =     &GSDIST[2];
              speed =      -1.0;
              radiusfactor = -1.0;
              newregion = -1;
              status =     unknown;
           );

        modify  &goal (refno = &ring.refno);

}; -- end of CCREstimateMotion

--*************************************************************************
--The next set of rules determine the strength of the interaction with the GS
--   based on the ratio and and appropriate region breakpoint.

rule CCRStrongInterWEncounter
--Is the CCR-GS interaction strong?
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
              ringtype = CCR;
              reg   = &goal.reg;
              inter = nil;
              (&mring.ratio >= &reg.bkpt[3]);
```

```
            (&mring.previousencounter = 1B););
-->

        modify &mring (inter = strong; regime = 4);

}; --end CCRStrongInteractionWEncounter

rule CCRStrongInterNoEncounter
--Is the CCR-GS interaction strong with no previous encounter?
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                ringtype = CCR;
                reg   = &goal.reg;
                inter = nil;
                (&mring.ratio >= &reg.bkpt[3]);
                (&mring.previousencounter = 0B););
-->
        modify &mring (inter = strong; regime = 3);

}; --end CCRStrongInteractionNoEncounter

rule CCRMediumInteraction
--Is the CCR-GS interaction medium?
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION
                        ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = CCR;
                        reg   = &goal.reg;
                        inter = nil;
                        ((&mring.ratio <= &reg.bkpt[3]) /\
                         (&mring.ratio > &reg.bkpt[2])));
-->
        modify &mring (inter = medium; regime = 2);

}; --end CCRMediumInteraction


rule CCRWeakInteraction
--Is the CCR-GS interaction weak?
```

```
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = CCR;
                        reg   = &goal.reg;
                        inter = nil;
                        ((&mring.ratio <=  &reg.bkpt[2]) /\
                         (&mring.ratio > &reg.bkpt[1])));
-->
        modify &mring (inter = weak; regime = 1);

}; --end CCRWeakInteraction

rule CCRNoInteraction
--Is the CCR-GS interaction nonexistant?
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION     ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = CCR;
                        reg   = &goal.reg;
                        inter = nil;
                        (&mring.ratio <= &reg.bkpt[1]););
-->
        modify &mring (inter = none );

}; --end CCRNoInteraction

--*************************************************************************
--The next set of rules modifies the speed, direction and radius factor
--  based on the strength of interaction with the GS, region, and
--- previous encounters with GS.

rule CCRModifyStrong
--Calculate modified location and radius for strong or medium interaction
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION     ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
```

```
                ringtype = CCR;
                reg   = &goal.reg;
                modifications = nil;
                inter = strong;);
-->
    local  &RADFAC:   real,
           &SPEED:  real,
           &NHEAD:     real,
           &NEWDIR:  integer;

    &SPEED = &reg.speed + 4.0;
    &NHEAD=
           &CCRparms[&goal.reg][&mring.regime][&mring.mdp].Heading;

    &NEWDIR =
           &CCRparms[&goal.reg][&mring.regime][&mring.mdp].FinalDir;

    &RADFAC =  &reg.gsdec; --decrease radius

    modify &mring (   newheading = &NHEAD;
                speed = &SPEED;
                modifications = calculated;
                radiusfactor = &RADFAC;
                newdir = &NEWDIR);

}; --end CCRModifyStrong

rule CCRModifyMedium1
--medium interaction, no previous encounter with GS (regions 1-4)
{
    &goal ( GOAL       ringtype = CCR;);

    &reg ( REGION   ringtype = CCR;
                reg = &goal.reg;);

    &mring ( movedRING
                ringtype = CCR;
                reg   = &goal.reg;
                (&mring.reg <= 4);
                previousencounter = 0B;
                (&mring.inter = medium);
                modifications = nil;);

-->
    local  &RADFAC:   real,
           &NHEAD:    real,
           &SPEED:  real,
           &NEWDIR:  integer,
           &SPEEDUP:  real;
```

```
        &SPEEDUP = 1.0;
        &SPEED = &reg.speed + &SPEEDUP;

        &NHEAD =
              &CCRparms[&goal.reg][&mring.regime][&mring.dir].Heading;
        &RADFAC =  1.0;

        &NEWDIR =
              &CCRparms[&goal.reg][&mring.regime][&mring.dir].FinalDir;


        modify &mring (   newheading   = &NHEAD;
                          radiusfactor =  &RADFAC;
                          newdir       = &NEWDIR;
                          speed            = &SPEED;
                          modifications = calculated;
              );

}; --end CCRModifyMedium1

rule CCRModifyMedium2
--medium interaction, no previous encounter with GS (regions 5-9)

{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                          reg = &goal.reg;);

        &mring ( movedRING
                          ringtype = CCR;
                          reg   = &goal.reg;
                          (&mring.reg > 4);
                          previousencounter = 0B;
                          (&mring.inter = medium);
                          modifications = nil;);
-->
        local   &RADFAC:    real,
                &NHEAD:     real,
                &SPEED:   real,
                &NEWDIR:   integer,
                &SPEEDUP:  real;

        &SPEEDUP = 2.0;
        &SPEED = &reg.speed + &SPEEDUP;

        &NHEAD =
              &CCRparms[&goal.reg][&mring.regime][&mring.dir].Heading;
```

```
        &RADFAC =  1.0;

        &NEWDIR =
              &CCRparms[&goal.reg][&mring.regime][&mring.dir].FinalDir;

        modify &mring (   newheading   =  &NHEAD;
                          radiusfactor =  &RADFAC;
                          newdir       =  &NEWDIR;
                          speed        =  &SPEED;
                          modifications = calculated;
              );

}; --end CCRModifyMedium2

rule CCRModifyMedium3
--medium interaction, with previous encounter with GS (regions 1-4)
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                          reg = &goal.reg;);

        &mring ( movedRING
                          ringtype = CCR;
                          reg    = &goal.reg;
                          (&mring.reg <= 4);
                          previousencounter = 1B;
                          (&mring.inter = medium);
                          modifications = nil;);

-->
        local   &RADFAC:    real,
                &NHEAD:    real,
                &SPEED:    real,
                &NEWDIR:    integer,
                &SPEEDUP:   real;

        &SPEEDUP = 1.0;
        &SPEED = &reg.speed + &SPEEDUP;

        &NHEAD =
                &CCRparms[&goal.reg][&mring.regime][&mring.dir].Heading;
        &RADFAC =  1.1;

        &NEWDIR =
              &CCRparms[&goal.reg][&mring.regime][&mring.dir].FinalDir;
```

```
        modify &mring (  newheading   =  &NHEAD;
                         radiusfactor =  &RADFAC;
                         newdir       =  &NEWDIR;
                         speed =         &SPEED;
                         modifications = calculated;
                );

}; --end CCRModifyMedium3

rule CCRModifyMedium4
--medium interaction, with previous encounter with GS (regions 5-9)
{
        &goal ( GOAL      ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = CCR;
                        reg   = &goal.reg;
                        (&mring.reg > 4);
                        previousencounter = 1B;
                        (&mring.inter = medium);
                        modifications = nil;);
-->
        local   &RADFAC:   real,
                &NHEAD:    real,
                &SPEED:   real,
                &NEWDIR:   integer,
                &SPEEDUP:  real;

        &SPEEDUP = 2.0;
        &SPEED = &reg.speed + &SPEEDUP;

        &NHEAD =
                &CCRparms[&goal.reg][&mring.regime][&mring.dir].Heading;
        &RADFAC =  1.1;

        &NEWDIR =
                &CCRparms[&goal.reg][&mring.regime][&mring.dir].FinalDir;


        modify &mring (  newheading   =  &NHEAD;
                         radiusfactor =  &RADFAC;
                         newdir       =  &NEWDIR;
                         speed =         &SPEED;
                         modifications = calculated;
                );
}; --end CCRModifyMedium4
```

```
rule CCRModifyWeak
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = CCR;
                        reg   = &goal.reg;
                        (&mring.inter = weak);
                        modifications = nil;);

-->
        local   &RADFAC:   real,
                &SPEED:   real,
                &NHEAD:    real;

        &SPEED = &reg.speed;
        &NHEAD =
                &CCRparms[&goal.reg][&mring.regime][&mring.dir].Heading;

        modify &mring (   newheading    = &NHEAD;
                        radiusfactor  = 1.0;
                        speed      = &SPEED;
                        newdir = (&mring.dir \ 16 ) + 1;
                        modifications = calculated;
                );
}; --end CCRModifyWeak

rule CCRNoModification
{
        &goal ( GOAL        ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = CCR;
                        reg   =    &goal.reg;
                        inter =    none;
                        modifications = nil;);

-->
        local  &SPEED:     real,
               &NHEAD:      real;

        &SPEED = &reg.speed;
```

```
        &NHEAD =  &reg.heading;

        modify &mring (   newheading   = &NHEAD;
                          radiusfactor = 1.0;
                          speed = &SPEED;
                          modifications = calculated;
                          newdir = @.dir
                          );

}; --end CCRNoModification

rule CCRApplyModifications
--Calculate new location based on values for dlat, dlong, and speed
{
        &goal ( GOAL       ringtype = CCR;);

        &reg ( REGION   ringtype = CCR;
                          reg = &goal.reg;);

        &mring ( movedRING
                          ringtype = CCR;
                          reg  =   &goal.reg;
                          modifications = calculated;
                          );

        &ring ( RING
                          ringtype = CCR;
                          refno = &mring.refno;
                          status = active);

-->
        local  &SPEED:     real,
               &NHEAD:        real,
               &STEP:     real,
               &LAT:       real,
               &LONG:      real,
               &LATSTEP:   real,
               &LONGSTEP:  real,
               &DIST:      real,
               &DIR:             integer,
               &RAD:       real;

        cast &STEP = &STEPDAYS;
        &SPEED = &mring.speed;
        &NHEAD  = &mring.newheading;
        &LAT =   &ring.lat;

        call nringstep(&DIST, &STEP, &SPEED, &LATSTEP,
               &LONGSTEP, &LAT, &NHEAD);
```

```
        &LAT =  &ring.lat + &LATSTEP;
        &LONG = &ring.long + &LONGSTEP;

        --decay rates are weekly
        &RAD =  &mring.newradius *
                        (1.0 - (( 1.0 - &mring.radiusfactor) * (&STEP / 7.0)));

        modify &mring (  modlat =      &LAT;
                    modlong =          &LONG;
                    modradius =        &RAD;
                    modifications =    applied;
                    distance =         &DIST;
                    );

}; --end CCRApplyModifications


--**************************************************************************
-- Following rules are used for both wcr and ccr .
-- Determine new region, test for dissipation or coalescence, store new
--   position.

rule NewRegion(-50)
--What region did the eddy move to
{
        &goal ( GOAL );

        &mring ( movedRING
                    ringtype = &goal.ringtype;
                    reg  =  &goal.reg;
                    status =  active;
                    modifications = applied;
                    newregion < 0;);
-->
        local   &REG:          integer;

        &REG = testreg(&goal.ringtype, &mring.modlat,
                        &mring.modlong,  &mring.modradius);

        if (&REG = 0) write()'\n', |Error calculating new region, 0 returned|;
        modify &mring ( newregion = &REG);

        }; --end NewRegion

rule CheckDissipation
--Has ring radius has reached minimum value?
--Note that this is based on the minimum radius for the old region
```

```
{
        &goal ( GOAL        );

        &reg ( REGION   ringtype = &goal.ringtype;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = &goal.ringtype;
                        reg  =   &goal.reg;
                        modifications = applied;
                        ((@.status = unknown) \/ (@.status = constrained));
                        (&mring.modradius < &reg.minrad););
-->
        modify &mring (    status = dissipated);

}; --end CheckDissipation


rule CheckCoalescence
--Has ring coalesced with the GS?
--Note that this rule should fire for every ring. The function call for
--coalescence must appear on rhs because it is a C function (not simple)
{
        &goal ( GOAL );

        &reg ( REGION   ringtype = &goal.ringtype;
                        reg = &goal.reg;);

        &mring ( movedRING
                        ringtype = &goal.ringtype;
                        reg  =   &goal.reg;
                        modifications = applied;
                        ((@.status = unknown) \/ (@.status = constrained));
                        );
-->
        local  &REG:  integer,
               &LONG:  real,
               &LAT:       real;

        &LAT = &mring.modlat;
        &LONG = &mring.modlong;
        &REG = &mring.newregion;
        &REG = in_gs(&REG, &LAT, &LONG);

        if(&REG = 0)
               modify &mring (status = coalesced )
        else
               modify &mring (status = active);
}; --end CheckCoalescence
```

```
rule StoreNewPosition
--After all modifications have been made, store the new position if
-- the ring is still active
{
        &goal ( GOAL);

        &mring ( movedRING
                        ringtype = &goal.ringtype;
                        reg  =   &goal.reg;
                        (&mring.newregion > 0);
                        status = active;);

        &ring (RING     ringtype = &goal.ringtype;
                        reg   = &goal.reg;
                        refno = &mring.refno;
                        time = &goal.time);
-->

        modify &ring (status = processed);
        modify &goal (refno = 0);
        remove  &mring;

        make  (RING     ringtype =  &goal.ringtype;
                        refno          =  &ring.refno;
                        time   =  &ring.time + &STEPDAYS;
                        lat      =  &mring.modlat;
                        long     =  &mring.modlong;
                        radius   =  &mring.modradius;
                        reg       =  &mring.newregion;
                        dir      =  &mring.newdir;
                        status   =  active );

}; --end CCRStoreNewRing


rule StoreNonActive (-100)
--After all modifications have been made, store info about nonactive rings
--   coalesced or dissipated
{
        &goal ( GOAL);

        &mring ( movedRING
                        ringtype = &goal.ringtype;
                        reg  =    &goal.reg;
                        ((@.status = coalesced)\/(@.status = dissipated));
                        );

        &ring (RING     ringtype = &goal.ringtype;
```

```
                    reg   = &goal.reg;
                    refno = &mring.refno;
                    time = &goal.time);
-->
        modify &ring (status = processed);
        modify &goal (refno = 0);
        remove &mring;
        make (RING    ringtype = &goal.ringtype;
                    refno          = &ring.refno;
                    time    =  &ring.time + &STEPDAYS;
                    lat     = &mring.modlat;
                    long     =  &mring.modlong;
                    radius   =  &mring.modradius;
                    reg      =  &mring.newregion;
                    dir      =  &mring.newdir;
                    status   =  &mring.status );

}; --end NonActive

}; --end of module
```

Appendix D
Input Files for Comparison of Original and Revised Versions

Warm Core Rings--Actual Data

```
3
2 39.9 67.6 115.
1 39.6 70.8 75.
3 40.4 63.4 95.
```

Cold Core Rings--Actual Data

```
5
5 39.7 54.5 75.
1 35.1 71.2 85.
2 36.4 65.2 115.
3 38.1 60.2 115.
4 38.2 56.8 95.
```

Warm Core Rings--Artificial Data

```
9
1 34.0  72.1  100.
2 35.99 74.71 100.
3 38.22 73.0  100.
4 41.0  70.0  100.
5 41.0  67.0  100.
6 41.0  64.0  75.
7 38.0  62.0  30.
8 49.4  59.9  50.
9 40.0  57.0  100.
```

Cold Core Rings--Artificial Data

```
9
1 34.9 72.1 85.
2 36.0 73.0 115.
3 38.0 72.1 50.
4 35.1 69.0 75.
5 35.0 67.0 75.
6 39.0 65.9 75.
7 38.0 62.0 35.
8 39.0 59.9 50.
9 40.0 50.0 90.
```

Upper Gulf Stream Data

```
26
35.8 75.
```

36.5 73.9
37.3 73.
37.8 72.
38.3 71.
38.7 70.
38.8 69.4
39.1 68.1
39. 67.8
38.6 66.4
38.8 66.
39. 65.
39.1 64.
39.3 63.
39.6 62.
40.3 61.
40.5 60.2
40.8 59.
41. 58.2
41.2 57.6
41.7 56.6
42.3 55.4
41.8 53.7
42.4 52.6
42. 51.
41.7 50.

Lower Gulf Stream Data

26
34.8 75.
35.5 73.9
36.3 73.
36.8 72.
37.3 71.
37.7 70.
37.8 69.4
38.1 68.1
38. 67.8
37.6 66.4
37.8 66.
38. 65.
38.1 64.
38.3 63.
38.6 62.
39.3 61.
39.5 60.2
39.8 59.
40. 58.2
40.2 57.6

40.7	56.6
41.3	55.4
40.8	53.7
41.4	52.6
41.	51.
40.7	50.

Appendix E
Modified Geometry Routines

The procedure adjustringstep is called from the rule WCRApplyModifications. It takes as input the change in latitude in km (lat_dist), the change in longitude in km (long_dist), and current latitude (lat); it returns the change in latitude in degrees lat_step and the change in longitude in degrees(long_step). All parameters are passed by address.

The procedure nringstep is called from the rules CCRApplyModifications and WCREstimateMotion. It uses the stepsize, speed, heading, and current latitude to calculate the distance moved in degrees of latitude (dist), the change in latitude, and the change in longitude.

```
void adjustringstep ( lat_step, lat_dist, long_step, long_dist, lat)
        double *lat_step, *lat_dist,
              *long_step, *long_dist, *lat;

{
        double deg_per_rad = 57.29577951;    /*     180 / pi        */
        double dist;

/*      2 * pi * Rearth / 360 = 111.3 km, the size of a latitude degree.
                Rearth = 6378 km.                                */

/*     *lat_dist is a pointer to the change in latitude in km.
        Convert to degrees lat  */
        *lat_step = *lat_dist / 111.3;


/*      long_dist is a pointer to the change in longitude in km.
      Convert to degrees long  */

/*      lat is a pointer to "previous" ring center latitude.        */

        *long_dist = *long_dist / 111.3;
        *long_step = *long_dist / cos ((*lat) / deg_per_rad);

}
```

```
void nringstep ( dist, step, speed, lat_step, long_step, lat, heading)
        double *dist, *step, *speed, *lat_step,
                *long_step,  *lat;
        double  *heading;

{
        double deg_per_rad = 57.29577951;     /*       180 / pi         */
        double theta, dlat, dlong;

/*      0.864 converts from cm/sec to km/day (86,400 seconds/day).        */
/*      2 * pi * Rearth / 360 = 111.3 km, the size of a latitude degree.
                Rearth = 6378 km.                                 */
/*      step is a pointer to the time increment in days.          */
/*      speed is a pointer to the ring speed &Wn.speed for region n.      */

        *dist = *step * (*speed) *0.864 / 111.3;

/*       *heading is a compass heading. This is converted into an angle
       in the where 0 degrees is due west and then the change in
       latitude and longitude are calculated.                       */

       theta = *heading + 90.0;

        dlat = sin (theta/deg_per_rad);
        dlong = cos (theta/deg_per_rad);

        *lat_step = *dist * (dlat);

/*      long_factor is a pointer to the longitude adjust factor
                &Cn.dellong for region n.                         */
/*      lat is a pointer to "previous" ring center latitude.      */

        *long_step = (*dist / cos ((*lat) / deg_per_rad))* (dlong);

}
```

Appendix F
Templates for Explanation Component

These templates are in the data file templates.dat. For each rule there will be three entries enclosed in vertical bars. The first entry is the rule name, the second is the template used for a rule trace, and the third is the template for a summary.

```
|WCREstimateMotion|
|If the goal is to process a WCR in region ?reg at time ?time
  and WCR ?refno is active in region ?reg at latitude ?lat longitude ?long
  with radius ?radius km
 Then
   Using a default speed of ?speed cm/sec and heading of ?newheading degrees,
   WCR ?refno will travel a distance of ?distance degrees to a new position
   of latitude ?newlat longitude ?newlong.
   Using the default decay rate of ?decay % per week, the new radius
   will be ?newradius km.|

|Time ?time status:  latitude ?lat longitude ?long radius ?radius km.
       Region ?reg defaults were used to estimate new position and radius:
       -- Speed   = ?speed cm/sec.
          Heading = ?newheading degrees
        Gives new position: latitude ?newlat, longitude ?newlong.

       -- Default decay rate = ?decay % per week
          Gives a new radius: ?newradius km.|

|WCRGetInteraction|
|If the motion of WCR ?refno has been estimated
 Then
  the interaction ratio with the Gulf Stream based on the
  new location and new radius is calculated to be ?ratio.|

|The new position is ?gsdist degrees from the GS.
       Gulf Stream interaction ratio calculated:
              ratio = (new radius)/(distance to GS) = ?ratio |

|WCRStrongInteraction|
|If the interaction ratio of WCR ?refno
  is greater than ?bkpt1
 Then
  the interaction is classified as strong and
  interaction regime 1 is selected.|

|The Gulf Stream interaction is classified as strong if the ratio
       is greater than ?bkpt1.|

|WCRMediumInteraction|
```

| If the interaction ratio of WCR ?refno
is greater than ?bkpt2 and less than ?bkpt1
Then
the interaction is classified as medium and
interaction regime 2 is selected. |

| The Gulf Stream interaction is classified as medium if the ratio
is greater than ?bkpt2 and less than ?bkpt1. |

| WCRWeakInteraction |
| If the interaction ratio of WCR ?refno
is greater than ?bkpt3 and less than ?bkpt2
Then
the interaction is classified as weak. |

| The Gulf Stream interaction is classifed as weak if the ratio
is greater than ?bkpt3 and less than ?bkpt2. |

| WCRNoInteraction |
| If the interaction ratio of WCR ?refno is less than ?bkpt3
Then
the interaction is classified as none. |

| There was no interaction with the Gulf Stream because the ratio
was less than ?bkpt3. |

| WCRModifyStrongMedium |
| If the interaction of WCR ?refno is classified as ?inter
Then
The adjustment factor for region ?reg, regime ?regime, and
GS direction ?dir for latitude is ?dlat % of radius per week,
and for longitude is ?dlong % of radius per week.
The radius decay factor is ?x % per week. |

| The GS is ?dir of the eddy and the ?inter interaction causes
{?dlat ! 0.00 --the ring to be pushed ?dlatupdown a distance that is ?absdlat % of radius per week
}{?dlong ! 0.00 --the ring to be pushed ?dlongupdown a distance that is ?absdlong % of radius per week
}{?x ! 1.00 --the radius to decay an additional ?x % per week. } |

| WCRModifyWeak |
| If the interaction of WCR ?refno is classifed as weak
Then
the estimated latitude and longitude are not modified and
the radius decay factor is ?x % per week. |

| Weak Gulf Stream interaction causes an additional decay of ?x % per week. |

| WCRNoModification |
| If WCR ?refno did not interact with the Gulf Stream
Then
the estimated latitude, longitude, and radius are not modified. |
| |

| WCRApplyModifications |
| If the modification parameters for WCR ?refno have been determined
Then
The adjustment factors for latitude, longitude, and radius were
applied to give modified values of latitude ?modlat, longitude ?modlong,
and radius ?modradius km |

| Modifications were applied to give a new position
of latitude ?modlat longitude ?modlong.
The ring radius was adjusted to ?modradius km. |

| WCRRegion2SouthernLimit |
| If WCR ?refno is in region 2 and
the estimated latitude is less than 36.0 N
Then
set the estimated latitude to the southern limit of ?modlat N |

| Warm core rings in region 2 are not allowed to move south of 36.0 N.
Estimated latitude set to ?modlat N |

| WCRRegion2CoastalLimit |
| If WCR ?refno is in region 2 and
the estimated longitude is greater than 74.8 W
Then
set the estimated longitude to the coastal limit of ?modlong. |

| Estimated position exceeds coastal limit of 74.8 W.
Estimated longitude set to ?modlong W. |

| WCRRegion3ShelfBreakLimit |
| If WCR ?refno is in region 3 and
the estimated longitude is greater than
(101.0 - ?modlat) / 0.86
Then
set the estimated longitude to ?modlong. |

| Estimated longitude exceeds the shelf break limit defined by the line
(101.0 - latitude) / 0.86. Estimated longitude set to ?modlong. |

| WCRRegion4Isobath |
| If WCR ?refno is in region 4 and
the estimated latitude is greater than 40.0
Then

set the estimated latitude to ?modlat. I

I Estimated latitude, exceeds the isobath limit at latitude 40.0.
Estimated latitude set to ?modlat. I

I WCRRegion7ShelfBreak I
I If WCR ?refno is in region 7 and
the estimated latitude is greater then 42.0
Then
set the estimated latitude to ?modlat I

I In region 7 the shelf break limit is at latitude 42.0.
Estimated latitude set to ?modlat. I

I WCRRegion8ShelfBreak I
I If WCR ?refno is in region 8 and
the estimated latitude is greater than 43.2
Then
set the estimated latitude to ?modlat I

I Estimated latitude exceeds shelf break limit of 43.2.
Estimated latitude set to ?modlat. I

I WCRRegion1DoNotHappen I
I If WCR ?refno is in region 1
Then
mark the ring as dissipated because warm core rings do not
occur in this region. I

I Warm core rings in region 1 dissipate rapidly. WCR ?refno was marked
as dissipated. I

I CCREstimateMotion I
I If the goal is to process a CCR in region ?reg at time ?time
and CCR ?refno is active in region ?reg at latitude ?lat longitude ?long
with radius ?radius km
Then
Using the default decay rate of ?decay % per week, the new radius
will be ?newradius km.
Using the current latitude and longitude, the distance to the
GS is calculated as ?gsdist degrees, direction ?dir.
An interaction ratio of radius/distance is calculated as ?ratio.
The ring ?yesno have a strong encounter with the GS at the
last time step.
I

I Time ?time status: latitude ?lat longitude ?long radius ?radius km.
--Region ?reg default decay rate = ?decay per week.
Gives a new radius: ?newradius km.

--Current position is ?gsdist degrees from the GS.
--GS is ?dir of the eddy.
Gulf Stream interaction ratio was calculated:
ratio = (new radius)/(distance to GS) = ?ratio. |

| CCRStrongInterWEncounter |
| If the interaction ratio of CCR ?refno
is greater than or equal ?bkpt1 and
there was an encounter with the GS at the previous time step
Then
the interaction is classified as strong and
interaction regime 4 is selected. |

| The Gulf Stream interaction is classifed as strong if the ratio is greater than ?bkpt3. The encounter with the Gulf Stream on the previous time step influences the direction of movement. |

| CCRStrongInterNoEncounter |
| If the interaction ratio of CCR ?refno
is greater than or equal ?bkpt3 and
there was no encounter with the GS at the previous time step
Then
the interaction is classified as strong and
interaction regime 3 is selected. |

| The Gulf Stream interaction is classifed as strong if the ratio is greater than ?bkpt3. |

| CCRMediumInteraction |
| If the interaction ratio of CCR ?refno
is greater than or equal ?bkpt2 and less than ?bkpt3
Then
the interaction is classified as medium and
interaction regime 2 is selected. |

| The Gulf Stream interaction is classified as medium if the ratio is greater than or equal ?bkpt2 and less than ?bkpt3. |

| CCRWeakInteraction |
| If the interaction ratio of CCR ?refno
is greater than or equal ?bkpt1 and less than ?bkpt2
Then
the interaction is classified as weak and
interaction regime 1 is selected. |

| The Gulf Stream interaction is classified as weak if the ratio is greater than or equal ?bkpt1 and less than ?bkpt2. |

| CCRNoInteraction |

I If the interaction ratio of CCR ?refno
is less than ?bkpt1
Then
the interaction is classified as none. I

I There was no interaction with the Gulf Stream because the ratio was
less than ?bkpt1. I

I CCRModifyStrong I
I If the interaction of CCR ?refno is classified as ?inter
Then
The speed of the ring will be ?speedup cm/sec faster than
the default speed of ?speed cm/sec.
The heading for region ?reg, regime ?regime, and GS direction ?dir
is ?newheading degrees.
The general direction of eddy movement for the next time step is ?newdir.
The radius modification factor is ?x. I

I The strong interaction with the Gulf Stream causes
-- a ?speedup cm/sec speedup over the default speed of ?speed cm/sec.
-- a heading of ?newheading degrees.
-- a general direction for the next time step of ?newdir.{?x ! 0.00
-- an additional ?x % decay of the radius. } I

I CCRModifyMedium1 I
I If the interaction of CCR ?refno is classified as ?inter
and the region is <= 4
and there was no encounter on the last time step
Then
The speed of the ring will be ?speedup cm/sec faster than
the default speed of ?speed cm/sec.
The heading for region ?reg, regime ?regime, and general direction ?dir
is ?newheading degrees.
The general direction of eddy movement for the next time step is ?newdir.
The radius modification factor is ?x I

I The medium interaction with the Gulf Stream caused
-- a ?speedup cm/sec speedup over the default speed of ?speed cm/sec.
-- a heading of ?newheading degrees.
-- a general direction for the next time step of ?newdir. I

I CCRModifyMedium2 I
I If the interaction of CCR ?refno is classified as ?inter
and the region is > 4
and there was no encounter on the last time step
Then
The speed of the ring will be ?speedup cm/sec faster than
the default speed of ?speed cm/sec.
The heading for region ?reg, regime ?regime, and general direction ?dir

is ?newheading degrees.
The general direction of eddy movement for the next time step is ?newdir.
The radius modification factor is ?x |

| The medium interaction with the Gulf Stream caused
-- a ?speedup cm/sec speedup over the default speed of ?speed cm/sec.
-- a heading of ?newheading degrees.
-- a general direction for the next time step of ?newdir. |

| CCRModifyMedium3 |
| If the interaction of CCR ?refno is classified as ?inter
and the region is <= 4
and there was an encounter on the last time step
Then
The speed of the ring will be ?speedup cm/sec faster than
the default speed of ?speed cm/sec.
The heading for region ?reg, regime ?regime, and general direction ?dir
is ?newheading degrees.
The general direction of eddy movement for the next time step is ?newdir.
The radius modification factor is ?x |

| The medium interaction with the Gulf Stream caused
-- a ?speedup cm/sec speedup over the default speed of ?speed cm/sec.
-- a heading of ?newheading degrees.
-- a new direction of ?newdir.
-- a 10% increase in radius size is caused by the previous encounter. |

| CCRModifyMedium4 |
| If the interaction of CCR ?refno is classified as ?inter
and the region is > 4
and there was an encounter on the last time step
Then
The speed of the ring will be ?speedup cm/sec faster than
the default speed of ?speed cm/sec.
The heading for region ?reg, regime ?regime, and direction ?dir
is ?newheading degrees.
The general direction of eddy movement for the next time step is ?newdir.
The radius modification factor is ?x |

| The medium interaction with the Gulf Stream caused
-- a ?speedup cm/sec speedup over the default speed of ?speed cm/sec.
-- a heading of ?newheading degrees.
-- a new direction of ?newdir.
-- a 10% increase in radius size is caused by the previous encounter. |

| CCRModifyWeak |
| If the interaction of CCR ?refno is classified as ?inter
and the region is > 4
and there was an encounter on the last time step

Then
The speed of the ring will be ?speedup cm/sec faster than
the default speed of ?speed cm/sec.
The heading for region ?reg, regime ?regime, and direction ?dir
is ?newheading degrees.
The general direction of eddy movement for the next time step is ?newdir.
The radius modification factor is ?x |

|The weak interaction with the Gulf Stream caused
-- a heading of ?newheading degrees.
-- general direction for next time step of ?newdir.
-- speed of ?speed cm/sec |

|CCRNoModification|
|If the interaction of CCR ?refno is classified as ?inter
Then
The default speed of ?speed cm/sec is used.
The default heading for region ?reg is ?newheading degrees.
The general direction of eddy movement for the next time step remains ?newdir.
The radius modification factor is ?x |

|Default parameters used
-- speed of ?speed cm/sec
-- a heading of ?newheading
-- general direction for next time step remains ?dir. |

|CCRApplyModifications|
|If the modification parameters for CCR ?refno have been determined
Then
The ring will move a distance of ?distance km to a new
position of latitude ?modlat longitude ?modlong with
a radius of ?modradius km. |

|The ring was moved a distance of ?distance to latitude ?modlat
and longitude ?modlong. The new radius is ?modradius km. |

|NewRegion|
|If the motion calculations for ?ringtype ?refno are complete
Then
the new region is determined to be ?newregion. |

|Ring moved from region ?reg to region ?newregion. |

|CheckDissipation|
|If ?ringtype ?refno has a new radius (?modradius km) less than
the minimum radius (?minrad km)for region ?reg
Then
the ring is marked as dissipated for the next time step. |

|The new radius (?modradius km) was less than the minimum
radius of ?minrad km for region ?reg.
Ring marked as dissipated.|

|CheckCoalescence|
|If ?ringtype ?refno motion calculations are complete
Then
the distance to the Gulf Stream was calculated and the ring has
?yesno coalesced with the Gulf Stream.|

|The distance to the Gulf Stream was calculated and the ring
has coalesced with the Gulf Stream.|

|StoreNewPosition|
|If all calculations for ?ringtype ?refno are complete
and the eddy is still active
Then
store the new position (latitude ?lat longtitude ?long)
and radius (?radius km) at time ?time.|

|Status at time ?time : latitude ?lat longitude ?long radius ?radius km.|

|StoreNonActive|
|If all calculations for ?ringtype ?refno are complete
and the eddy is not active
Then
mark the eddy as ?status at time ?time.|

||

Appendix G
Explanation Component Source and Data Files

1. Changes to existing files
   eddytypes.ops
   The variable &EXP is declared. This variable is set to 1b if information needed for explanation should be collected.

   eddies.ops
   The variable &EXP is initialized.
   If &EXP is 1b, the procedure InitTemplates is called to initialize the template structures.
   If &EXP is 1b, the procedure Explain is called when the use enters "q" to terminate a session.

   racycle.ops
   After a rule has been selected for firing, if &EXP is 1b, the irule procedure is called to get the rule name, and the instance procedure is called to record information about the instantiation. After the rule has fired, the procedure StoreRuleInstance will be called to record the rule instantiation information.

2. The Introspection Module is implemented in OPS83. The code for this module is found in the file explain.ops. The data structures used to store the instantiation records are defined in explan.h. Utility routines used to access the routines are written in C. The source for the C routines is in explan.c.

3. The Presentation Module is implemented in C. The source code for this module is in the files explan.h and explan.c.

4. Templates used by the Presentation Module are found in the file templates.dat. This data file must reside in the same directory as the executable.

5. Traces and summaries of all eddies are written to the data file explain.dat by the Presentation Module.

DISTRIBUTION LIST

Fleet Numerical Oceanography Center
Monterey, CA 93943-5105
Attn: Commanding Officer, Capt. Plant
Code 42, Mike Clancy
Code 43, Mr. Jim Cornelius
Code 70, Ms. P. Chavasant

Naval Research Laboratory
Washington, DC 20375
Attn: Dr. Ken Johnston, Code 4200
Dr. Vince Noble, Code 4200
Library (2)

Space and Naval Warfare Systems Command
2511 Jeff Davis Hwy.
Washington, DC 20362-5100
Attn: Code PMW 165, CDR Ranelli
Code PMW 165, Ed Harrison
Code PMW 165, CDR Trumbower
Code PMW 165, Mr. Don Montgomery

Institute for Naval Oceanography
Building 1103
Stennis Space Center, MS 39429
Attn: Dr. John Apel, Director

Naval Eastern Oceanography Center
McAdle Bldg (U-117)
Naval Air Station
Norfolk VA 23511-5399
Attn: Mr. Chuck Weigand

Naval Oceanographic Office
Stennis Space Center, MS 39522-5001
Attn: Commanding Officer, Capt. Felt
Code TD, Mr. Landry Bernard
Code M, Mr. Doug Gordon
Code MSF, Mr. Mark Boston
Code DO, Dr. Jerry Carroll
Code O, Dr. Bill Jobst
Code OP, Dr. Vance Sprague
Code OPT, Mr. A. Johnson
Code OPTA, Mr. T. Bennett, Dr. C. Horton
Code OC, CDR Denny Whitford, Mr. Phil Bylsma
Code OPTR, Mr. J. Rigney
Library (2)

Naval Postgraduate School
Monterey, CA 93943
Attn: Code 68, Dr. C. Collins, Dr. C. Wash

Office of Naval Research
800 N. Quincy St.
Arlington, VA 22217-5000
Attn: Code 120M, Mr. R. Peloquin

U.S. Naval Academy
Department of Oceanography
Annapolis MD 21402-5026
Attn: Chairman
Dr. Al Strong

Commander, Naval Oceanography Command
Stennis Space Center, MS 39529-5000
Attn: Dr. Paul Moersdorf
LCDR D. Poffenberger
Mr. Chet Wilcox
Mr. Ray Partridge

Naval Research Laboratory
Stennis Space Center, MS 39529-5004
Attn: Code 125L (10)
Code 125P
Code 200
Code 222, Mr. G. Kerr
Code 300
Code 320
Code 321
Code 321, Mr. D. May, Mr. R. Holyer, Dr. S. Gallegos, Ms. J. Chase, Mr. J. Hawkins, Ms. S. Peckinpaugh
Code 322, Dr. J. Harding, Dr. M. Carnes, Dr. G. Heburn
Code 323, Dr. D. Thompson, Dr. H. Hurlburt, Dr. J. Kindle, Mr. D. Fox, Mr. J. Metzger, Dr. J. Mitchell
Code 350, Mr. M. Harris

Naval Research Laboratory
Atmospheric Directorate
Monterey, CA 93943
Attn: Code 401, Dr. P. Tag
Code 431, Dr. T. Tsui, Dr. C. Crosiar, Mr. K. Richardson
Code 441, Ms. Rosemary Lande
Code 442, Lt. D. Miller

Planning Systems, Inc.
115 Christian Lane
Slidell, LA 70458
Attn: Dr. R. Crout

Sverdrup Technology, Inc.
Stennis Space Center, MS 39529
Attn: Dr. G. Ransford
Mr. W. Osterman
Ms. S. Seal

Louisiana State University
Department of Computer Science
Baton Rouge, LA 70803
Attn: Prof. S.S. Iyengar

Mississippi State University
Department of Computer Science
Drawer CS
Starkville, MS 39762-5623
Attn: Dr. S. Bridges

Office of Naval Technology
800 N. Quincy St.
Arlington, VA 22217-5000
Attn: Code ONT 234, Dr. Tom Warfield

Tulane University
Department of Computer Science
301 Stanley Thomas Hall
New Orleans, LA 70118
Attn: Dr. B. Buckles
Dr. F. Petry

University of Tennessee
Computer Science Department
Room 107 Ayres Hall
Knoxville, TN 37996
Attn: Dr. M. Thomason

REPORT DOCUMENTATION PAGE

Form Approved
OBM No. 0704-0188

1. Agency Use Only *(Leave blank).*

2. Report Date.
September 1992

3. Report Type and Dates Covered.
Final

4. Title and Subtitle.
Adding Explanation Capability to a Knowledge-Based System for Interpretation of Oceanographic Images

5. Funding Numbers.

Program Element No.	0602435N
Project No.	3582
Task No.	MOG
Accession No.	DN256010
Work Unit No.	13212R

6. Author(s).
Susan Bridges*

7. Performing Organization Name(s) and Address(es).
Naval Oceanographic and Atmospheric Research Laboratory
Ocean Science Directorate
Stennis Space Center, MS 39529-5004

8. Performing Organization Report Number.
NOARL Technical Note 296

9. Sponsoring/Monitoring Agency Name(s) and Address(es).
Naval Oceanographic and Atmospheric Research Laboratory
Basic Research Management Office
Stennis Space Center, MS 39529-5004

10. Sponsoring/Monitoring Agency Report Number.
NOARL Technical Note 296

11. Supplementary Notes.
*Mississippi State University
Department of Computer Science
Starkville, MS 39762-5623

12a. Distribution/Availability Statement.
Approved for public release; distribution is unlimited.

12b. Distribution Code.

13. Abstract *(Maximum 200 words).*

An explanation component has been developed for an existing oceanographic expert system that predicts the movement of mesoscale features associated with the Gulf Stream. The information provided by the expert system is used by image processing analysts when the oceanographic features cannot be observed by satellite data due to interference such as cloud cover. The addition of an explanation capability gives users a basis for judging the quality of the system's decision making process.

14. Subject Terms.
Remote Sensing, Artificial Intelligence, Data Assimilation, Satellite Date

15. Number of Pages.
65

16. Price Code.

17. Security Classification of Report.	18. Security Classification of This Page.	19. Security Classification of Abstract.	20. Limitation of Abstract.
Unclassified	Unclassified	Unclassified	SAR