SPINNING AN EXPERT OIL SPILL INFORMATION MANAGEMENT SYSTEM ON THE WORLD WIDE WEB

John Collins, Christos Douligeris (corresponding author), Roger Riggs
Department of Electrical and Computer Engineering
Ocean Pollution Research Center
University of Miami
Coral Gables, Fl, 33124, USA
Voice: (305) 284-3597
Fax: (305) 284-4044
E-mail:
collins@ece.miami.edu
christos@ece.miami.edu
riggs@ece.miami.edu

SPINNING AN EXPERT INFORMATION SYSTEM ON THE WORLD WIDE WEB

KEYWORDS: Oil Spill Response, Intelligent Information Systems, CLIPS, World Wide Web, HTML

Abstract

The OSIMS (Oil Spill Information Management System) project funded by the US. Coast Guard is to build an information system applicable to oil spill planning, response and evaluation. In design phases it was decided to implement a distributed system with intelligent elements and coordination. In this article we describe our current prototype that uses HTML to provide world-wide access to information in a passive form.

We further describe our current efforts to introduce AI capability into the system by making information accessible to and through the CLIPS expert system shell. This begins with passive information access between HTML and expert system elements. In the expert system, we have added a simple mechanism for specific planning that can actively direct information acquisition for the crisis manager.

INTRODUCTION

Oil spill response requires the marshaling of masses of regulatory, descriptive and procedural detail. This information is contained in a diversity of systems including classical databases, GIS systems, programmed mathematical models, broadcast data and volumes of procedural information. Moreover oil spill response requires the invocation of disparate expertise, utilization of multiple independent actors and satisfaction of various authorities.

In an earlier paper [Douli], we described the design and initial efforts at constructing an oil spill information management system (OSIMS). The goals of this system were to integrate diverse data and program elements in a single system that would provide access to all actors involved in spill research, planning and response efforts. A distributed heterogeneous system was chosen.

In the past year we have adopted an implementation approach centered on making the information available through the World Wide Web (WWW). Gross has described NOAA efforts to provide disaster information on the web in

HazardNet [Gross]. We agree that the WWW "would be a reasonable framework on which to build ...[an] emergency management network."

While constructing the nexus of information in HTML, it became apparent that a more active, task oriented access to information would be useful. Our position in this is similar to Gadomski et al. [Gadom] that "many decisions require only formal well structured knowledge... and should be recognized and allocated to [the] DSS's." We have differed from them by choosing a frame work of specific 'plans' as developed by the USCG and other agencies.

MAJOR ELEMENTS IN OSIMS

In designing OSIMS, it was apparent that a diversity of data sources and decision support tools existed. Some examples include: USCG oil spill data, NOAA environmental sensitivity maps, RSMAS numerical model data, COTP response plans. Other data is being assimilated internally but we also want the capability to reach external sources as well.

A distributed, heterogeneous approach promised to avoid a difficult and lengthy duplication of effort. Such systems provide the necessary interface to isolate the user from details of the individual components while allowing free distribution of information between elements and users of the system [Douli].

A database, expert system and information shell were chosen to implement the total system. These elements form the backbone of the system used for information distribution and systems integration. We describe each element, its goals and general function in the following.

Database

To access traditional databases and to include such non-traditional data as graphical data we selected POSTGRES as the data base for the prototype. POSTGRES is an academic, object-oriented database management system built on an extended relational model [POST].

Object orientation eases the difficulty of re-constructing the hierarchical domain that exists other system elements like CLIPS and HTML. Graphical data sets can be stored and accessed as binary large objects (BLOB's). Functional extensions provide a mechanism for domain translation between the database and the other elements.

Lastly, it is assumed that as the amount of data in OSIMS grows and access becomes more general, data access will require the consistency, concurrency and efficiency guarantee which database systems provide.

HTML Interface

To make information available to multiple locations, on diverse platforms and with a simple, uniform interface we decided to use the world wide web and HTML as a system interface for OSIMS. The HTML interface is increasingly popular and provides hyper-text like interface that requires very little user training. WE constructed an autiomated HTTML interface that could access information paced in a hierarchy deteremined by an examination of user needs [Tebea].

Connectivity is immediate. HTML be accessed by any machine running Netscape, Mosaic, Lynx or any other HTML browser so long as it is connected directly or by modem to the net. Some restriction is imposed by available HTML viewers which cannot display all the sorts of graphical visualizations we can generate. In some cases we can provide only snapshots. On the other hand, we have been able to link to valuable text and graphics at foreign locations.

We have come to view the HTML nexus as a fundamental repository for information. As such it helps define the requirements of the system and to ensure that needed information is available. References to HTML objects such as maps and text can amplify the information created by CLIPS or retrieved from POSTGRES. We build explicit links from objects in these systems to HTML objects, but data objects are automatically included in the HTML hierarchy of access when stored in the system. Figure 5 shows a CLIPS referenced map object as viewed through HTML.

CLIPS Expert System Shell

CLIPS 6.0 is a forward chaining, rule based, expert system shell developed by NASA [CLIPS]. It supports objects, rules and functions. It can also embed, or be embedded in, C code. CLIPS is available on UNIX, DOS and Macintosh platforms. We detail the current use of CLIPS and some of it's proposed further uses in the remainder of this paper.

EXPERT INFORMATION SYSTEM

The need to integrate intelligent systems such as expert systems into traditional DSS and information systems has been argued elsewhere[Goul]. Expert systems can provide reasoning in domain specific terms to increase their perspicuity to both developers and users. Production rules are heuristic rules of an *if...then* nature that can be accumulated to increase the range of knowledge in the system.

We are able to represent much of the knowledge in OSIMS in the CLIPS object hierarchy itself. A domain such as PHYSICAL-OBJECT is a super-class for all particular concrete objects. Functions such as *distance(obj,obj)* are defined on the super-class PHYSICAL_OBJECT and are automatically available to any sub-class, such as BOOM, TRUCK, or UNIT. (These are, respectively, a class of physical boom placements, trucks and deployed tactical units.)

CLIPS contains a query mechanism that accesses instances of objects subject to specified attributes. The object oriented structure and CLIPS functions for accessing it made prototyping the CLIPS browser (discussed below) very straight-forward.

Bottom-up approach

Typically, expert systems begin with a specific problem, a well-defined domain, an expert(s) and a chosen tool. The tool enforces paradigms of knowledge representation and processing. In contrast, OSIMS began with a broad domain, many ill-defined problems and no single expert. Yet the problem of presenting information in a crisis situation is real and compelling.

Rich sources of information existed in the form of manuals, guidelines and regulations as well as dispersed expertise. Entraining that information in an easily accessible form is already a significant to an individual who must find information in a crisis. The OSIMS data available in HTML is partially duplicated in CLIPS (historically the CLIPS object hierarchy came first and we are adjusting it now).

CLIPS objects exist for types of boom, checklists, map references, etc. CLIPS objects are also created for all specific instances of units, resources, specific plans and major equipment from these class definitions. The problem then becomes to manipulate, navigate, focus and check information in this hierarchy.

CLIPS Browser

The HTML browser provides a direct view of CLIPS objects. CLIPS rules and functions are used to direct processing of information inside CLIPS. The CLIPS output is then formatted, sometimes internally to CLIPS, sometimes by independent C programs. Figures 2 and 3 show example output for the object hierarchy and the instances of a class, respectively.

The CLIPS browser shows sub- and super-classes of the current object. The definition the class can be displayed as well. The details of instances may also be examined. Focus is provided simply by user input, not context.

Planning

In OSIMS, plans are represented in a simple slot and filler approach based on USCG checklists for spill response planning. These forms are meant to collect and organize information on the various phases of spill response. We use them as a base planning by assembling and relating objects of concern, response tasks and response resources.

One example is the general spill response checksheet. It contains a general form and a list of further checksheets. Along with other details, it specifies that a subsidiary form and a series of subordinate checksheets be created. The Spill Report Form is used to input and collect information. Subordinate checksheets detail planning elements for individual stages of the response effort such as initialization, assessment, action strategy and disposal strategy. We discuss the form below.

We consider the sub-checksheets as 'sub-plans' in the CLIPS representations of the spill response checksheet. Instances of these are created on user or system demand and enrolled as tasks in the CLIPS object. The user may add or delete these. This 'fixed' plan approach is sometimes called special purpose planning as opposed to general planning.

Lower planning levels may have further sub-plans and forms, but specific tasks are at the bottom level. Tasks are of a number of specific types such as containment booming, in situ burning or skimming. Equipment, personnel and regulatory pre-conditions for the task are explicitly listed in the class objects for these tasks. The system can suggest specific tasks and units to perform them if the user requests. In any case the final selection of specfic tasks is left to the user, operating through the HTML interface.

The object orientation allows us to generate instances of the generic tasks with specific features bound to the user choices. The hierarchy of instantiated checksheets and tasks is regarded as the total response plan. Figure 1 shows an example of the CONTAIN_RESPONSE object as CLIPS code.

In our initial studies, we found a lack of any complete, detailed, accessible analysis of a spill response. Storing the details of spill response in this form should allow us eventually to recall spills as information to the user. Off-line analysis should allow us to criticize and perhaps even learn from recorded spills.

Information acquisition

While the object (HTML or CLIPS) hierarchy allows us to access information by category, even simple plans can drive information selection and acquisition by the demands of the task. This begins to provide a rudimentary active information access, presenting only useful and timely information to the user.[1]

Detailed information might be gleaned from the discursive accident description by automatic means. This however seems at the current experimental state of the art in automated text analysis. [Gomez] Rather we have focused on eliciting details and generating further information. Figure 4 shows the initial input screen for spill information.

For example, the choice of spill organization is conditioned by such definite features as the size of spill, nature of pollutant, impacted resources and available response materials. Less definite features, such as public interest, are also material. Rules are triggered by such features to derive new information such as a 'best' choice of spill organization. The final choice is left to the user.

Similarly, the need for containment booming of a vessel can be deduced from the USCG type classifications of the spill incident. (Some are: vessel-to-facility transfer, ship grounding, etc.) The exact length of boom can be deduced from entered or stored details of the vessel.

Task objects have preconditions on equipment, regulations, etc. Rules check requirements such as boom type and length against the equipment recorded in a UNIT's equipment-list. Initial UNIT selection is restricted to suitable objects by combining defined features of TASK objects such as quantities and object type into a query for

---

[1] All HTML information is always available through links.

acceptable objects of the required type and amount. Demons associated with equipment and units perform related tasks such as availability and suitability checks and inventory or scheduling updates.

Response evaluation

Response evaluation should occur in two ways. The system can compute changes in requirements, plans and outcomes as information, such as tasks and assigned units, change. The system can also present comparative information from other spill incidents.[2] Warnings, indicators and measures should be introduced for critical elements and values. Ideally changes in response tactics would interact with predictors such as shoreline recovery or ocean current models.

Secondly, comparative information could be used to gauge reasonability of choices. At present, the CLIPS system returns pointers to comparative data recorded in CLIPS objects. Ideally, comparative scenarios would be referenced based on situation variables.

GENERAL EXPERIENCES

System specification has been a significant problem. HTML is probably the quickest way to structure a corpus of information while providing simple, complete access. We have received some favorable comments on the HTML interface, but it will be critical to test the ease of planning for with real world personnel.

We have hand coded (laboriously and with many ellipses) knowledge into CLIPS, most of which also appears in HTML. Perhaps KQML, or Edinburgh's PLINTH or even HARDY, would provide some help in this direction.

In general the forward chaining paradigm seems adequate to meet our soft real time requirements. However, backward chaining can be implemented by explicit programming in CLIPS. We will have to see what happens as the volume of information, tasks and sub-systems grows.

We have tried to obey a general caveat against absolutely restricting the users choices. Because all information is ultimately stored in objects, the user always retains the ability to remove or add specific items though

---

[2] At present there are only 3 artificial, paradigm spills

the HTML interface. Update of dependent objects proceeds by demons from that point.

Future Directions

We have many ideas for extensions to the current system. The TASK's objects will be extended to include start and finish times. This will allow us to develop a time line for the evolving plan. From that, and effects models, we hope to be able to predict relative costs of changing plans.

Heuristic estimators of response characteristics can be added to the system. Data sources, such as Exxon's spill response manual, already provide estimators of response requirements. Dependent quantities such asnumber of personnel needed are parameterized by such features as spill volume, days into the spill, etc. [Exxon]

Comparative information is presently based on only the three paradigm spills. As data accrues, we intend to examine the features that are most critical to similarity and assessment. Such information would allow case based reasoning to be employed.

One original goal, to employ distributed processing, has gone begging in our effort to get a system in place. In general development has been a three step process: 1) make information available, 2) make information access pertinent, 3) make information automatically drive system components such as effects simulators, other numerical programs or communications.

REFERENCES

[Goul] Goul M., Hendersen J.C., Tonge F. M., The Emergence of Artificial Intelligence as a Reference Discipline for Decision Support Systems research, in Decision Sciences, V23, 1992

[CLIPS] A top-level reference to CLIPS info at the source: http://www.jsc.nasa.gov/~clips/CLIPS.html

[Douli] Douligeris, Collins, Iakovou, Sun and Riggs, "Designing an Oil Spill Information Management System", TIEMEC'95, Nice, France, May 9-12, 1995, PP 386-391

[Exxon] Oil Spill Response Field Manual, EXXON Corp.

[Gadom] Gadomski, Adam M., Bologna, Di Costanzo, Intelligent Decision Support for Cooperating Emergency Managers: The TOGA Based Conceptualization Framework, TIEMEC'95, Nice, France, May 9-12, 1995, PP 379-385

[Gomez] Fernando Gomez, U. of Central Florida, in a personal communication regarding the state of automated text analysis, November 95

[Gross] Gross, Edward M., The Design and Management of an International Disaster Information Resource Network [Building and Emergency Lane on the Information Superhighway], TIEMEC'95, Nice, France, May 9-12, 1996, PP 250-252

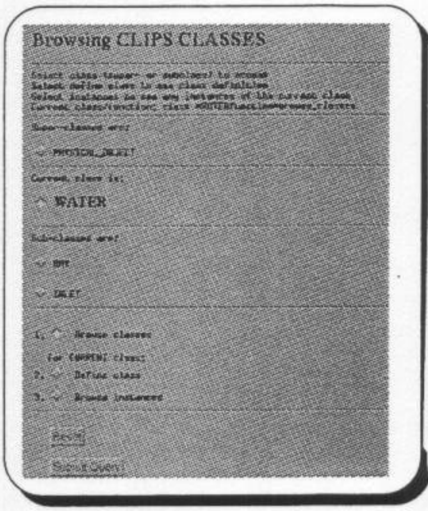
[POST] A top-level entry to information on POSTGRES, from the source: http://info.fuw.edu.pl/market/postgres.html also Stonebreaker and Kemnitz, "The POSTGRES Next Generation DBMS", Communications of the ACM, v34, N10 Oct. 1991.

[Tebea] Tebeau, Peter, Survey of User Needs and System Components for the South Florida Oil Spill Research Management System (OSIMS), Tech Report, Dept. of Electrical and Computer Engineering, U. of Miami, 1994

FIGURES

```
;; the elements of containment and cleanup checklist
(defclass CONTAIN_RESPONSE
  (is-a PLAN)
  (slot off_shore_strategy (default TDA)...)
  (slot nearshore_strategy (default TDA)...)
  (slot shoreline_strategy (default TDA)...)
  (slot inland_strategy    (default TDA)...)
  (slot sensitive_area__strategy (default TDA)...)
  (multislot staging_areas
       (default TDA staging_areas)...)
  (multislot monitor-oil    (default schedule)...)
  (slot chemical_measures  (default TDA)...)
  (multislot disposal
       (default select disposal_sites)...)
```

figure 1 - A CLIPS object THE CONTAIN_RESPONSE class. TDA indicates a plan must be developed for that slot, other slots hold values. PLAN's have slots for individual tasks which are the physical actual response actions. These in turn have required slots filled by unit, target, etc.



Browsing CLIPS CLASSES

Select class (super- or subclass) to access
Select define class to see class definition
Select instances to see any instances of the current class
Current class/function: [illegible]

Super-classes are:

PHYSICAL_OBJECT

Current class is:

^ WATER

Sub-classes are:

BAY

INLET

1. ^ Browse classes

for CURRENT class:

2. Define class

3. Browse instances

Clips class info in HTML
Figure 2.

^ RESOURCE

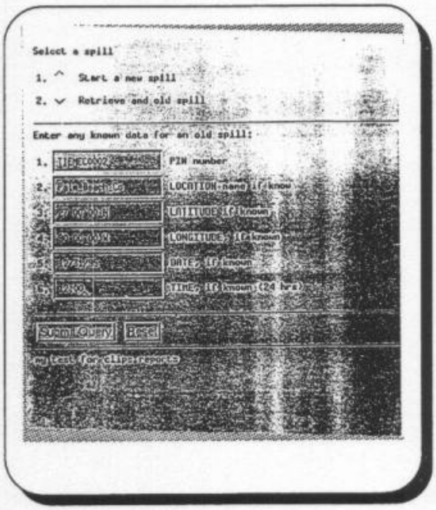
Current class is RESOURCE which has the instances:

1. PB-County-A1
2. PB-County-A2
3. PB-County-A5
4. PB-County-A9
5. PB-County-A10
6. PB-County-A11
7. PB-County-C11
8. PB-County-A13
9. w-turtles
10. wo-turtles
11. power-plant-cooling-intake
12. con2
13. con3
14. con4
15. con5
16. con6
17. con7
18. con8
19. con9

[PB_County_A1]
Palm Beach county, map2, location A1

ident is PB A1
resource_name is Ocean Beach - North of Lake Worth
priority is A
map is PB_county_map2
responseTasks has the values:
[Boom and skim at sea]

CLIPS instance info in HTML
Figure 3.



Select a spill

1. ^ Start a new spill
2. ∨ Retrieve and old spill

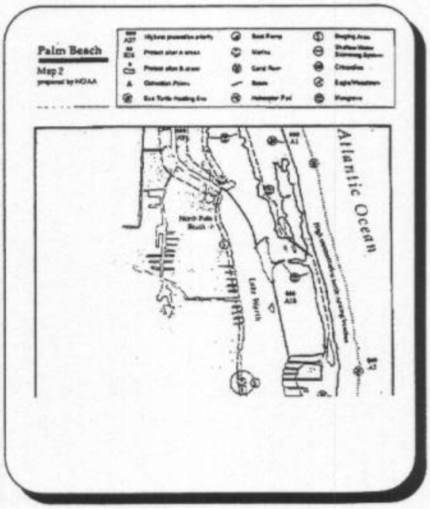
Enter any known data for an old spill:

1. IIEMEC0002 PIN number
2. Palm Beach Co LOCATION name if know
3. 27 00'00N LATITUDE if known
4. 80 00'00W LONGITUDE, if known
5. 07/31/95 DATE, if known
6. 1230 TIME, if known (24 hrs)

my test for clips reports

The spill initiation form

Figure 4.



An (HTML) map referenced
from CLIPS
Figure 5.