INTELLIGENT DECISION SUPPORT FOR COOPERATING EMERGENCY MANAGERS: THE TOGA BASED CONCEPTUALIZATION FRAMEWORK

Adam M. Gadomski*, Sandro Bologna, Giovanni Di Costanzo
ENEA - Italian National Agency for New Technology, Energy and the Environment
C.R. Casaccia, Via Anguillarese 301, 00060 ROMA (Italy)
* Independent Consultant, Via Dei Lecci 13 , 00062 Bracciano (Italy)
gadomski_a *or* bologna_s *or* dicostanzo_g@casaccia.enea.it

KEYWORDS: intelligent system, decision support system, emergency management, design.

ABSTRACT

The paper presents the results of a preliminary study related to the conceptual design of an Intelligent Decision Support System (IDSS) for cooperating emergency managers. Authors postulate to shift the designer efforts from passive DSSs, based on so called *menu-driven paradigm*, to the active DSSs based on the *goal-driven paradigm*. IDSS kernel should be the user-friendly interface between classical DSS and emergency managers. The general conceptualization methodology employed in this work is TOGA (Top-down Object-based Goal-oriented approach). The suggested architecture of IDSS is based on the model of Abstract Intelligent Agent, complemented with the results of the CEC funded Projects ISEM[1] and MUSTER[2].

INTRODUCTION AND MOTIVATIONS

In the stormy developing postindustrial society, environmental, technological and social decisions involve ever more risk and their consequences are often hardly predictable. This situation results from the continuously increasing decisional domains, their complexity, and many other factors of post-modern society. Very sophisticated, fast and precise modern technology requires qualitatively new tools for its control. Numerous disasters, caused by natural events and human-errors require an emergency and recovery management which need large volumes of data. For singular emergency managers, as well as, for the cooperating staffs, ever more frequently, these data are hardly evaluable and applicable. On the other hand, human mental capabilities are not yet in adequate way supported by new computer technologies which could significatively diminish probability of serious managerial errors.

So called passive Decision Support Systems (p-DSS) have been the first attempt to the computer aid for emergency managers. Unfortunately, their application requires from their users continuous learning and training to which typical emergency managers are not enough motivated. The use of large p-DSS involves so much technical decisions that these systems either require continuous effort of computer specialists or they are not used in practice.

Passive DSS is based on available "conventional" elements, methods and data-base management technologies. In these systems large part of the user decisions relies on the choice of the concrete button from a menubars or menutools being parts of a visualized hierarchical menu structures. In this way we can say that their input/output interfaces are funded on the *menu-driven paradigm*.

Let us mention that many decisions which are intuitively easy recognized by the managers as not "key decisions" or "less important " require only formal well structured knowledge (for example, searching in data bases, optimization), and should be recognize and allocated to DSSs. As the consequence, ever more frequently, *classical passive DSSs based on the menu-driven paradigm do not satisfy the requirements of their users.*

The contrary to menu-driven paradigm can be *goal-driven paradigm* of decision support. In this case, a computer system should be some kind of an active cooperator of its user, what we see, for example in (Hawgood et al. 1992) and (Muhanna 1993). Application of goal-driven paradigm eliminates redundancy of not actual in this moment alternatives and suggests choices determined by criteria defined on higher abstraction levels, for example related to importance, risk, duty or to a consequence of previous choices. In order to make autonomous decisions the Decision Support System needs a certain model of decisional processes (Yager 1992), (Watabe et al. 1992), (Michalski 1992) adaptable to particular choices resulting from a generic emergency scenario, and which are consequence of the current situation of emergency. In such context, IDSSs (Intelligent Decision Support System) can be viewed as computerized interfaces for "fitting" passive DSS functions to the requirements and preferences of man.

In general, IDSS are especially important when:

(a) the amount of information necessary for the management is so large, or its time density is so high, that the probability of human errors during emergency decision-making is not negligible,

(b) the coping with unexpected situations requires from the managers the remembering, mental elaboration and immediate application of complex professional knowledge, which if not properly used causes faulty decisions.

The basic problems for design and development of IDSSs are the conceptualization and representation of the user knowledge. Recent research results (Kay 1991), (Beaumont 1994) lead to the conclusion, that modeling of the user roles is required for identification and specification of this knowledge.

---

[1] ISEM (Information Technology Support for Emergency Management) ESPRIT Project no. 2322.

[2] MUSTER (Multi-User System for Training and Evaluating Environmental Emergency Response) CEC ENVIRONMENT Project.

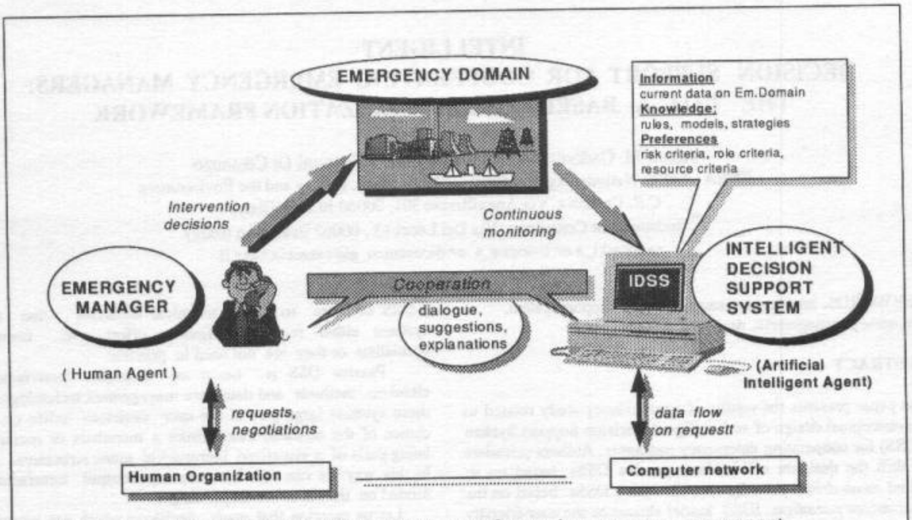

Fig. 1 General idea of an Intelligent Decision Support System for emergency management

We assume (Gadomski et al. 1993), (Balducelli et al. 1993) that user models should be based on a more general model of an intelligent agent which is able to realize autonomously goal-driven interventions in its environment.

Therefore the *goal-driven paradigm* must be based on a generic ideal model of decision-maker and on its decomposability on human and computer decision-makers. For such task *a general model of abstract intelligent agent as a role-dependent decision-maker is necessary.*

We say abstract, because such model of intelligent agent should be independent on its application-domain and independent on its software implementation environment.

The design of IDSS for EM (Emergency Management) requires an integrated application of the models and methodologies related to :

- large-scale emergency management
- abstract intelligent agent architecture
- decision-making under multilevel risk conditions and based on uncompleted and uncertain information or knowledge
- cooperation between intelligent agents
- computer aided training.

Formal specification frames of the behavior and knowledge of an ideal emergency manager (e-m) are the indispensable tools for IDSS design. This meta-knowledge is necessary for:

- acquisition of the domain-dependent knowledge (about EM)
- elaboration of the scenario of e-m activity
- allocation of the e-m activities and knowledge to IDSS
- definition of the new man-machine interface functions,
- standardization of the design documentation.

The aim of this paper is to indicate some criteria for the development and design of IDSS for EM. It should be a personal support for emergency managers working in frame of a large-scale emergency management system. Here, large-scale emergency-management concerns big industrial centers, nuclear or chemical plants, ports, railways networks, airports, as well as regional or national emergency organizations. A general view of the idea of IDSS is illustrated in Fig. 1.

CONCEPTUAL FRAMEWORK : ELEMENTS OF TOGA

In the recent years the models of intelligent agents are strongly investigated (see the subject matter literature, for example (Michalski 1992), (Hanks et al. 1993), (Gadomski 1994a), (Gadomski 1994b). Different new approaches to the software agents, like a general idea of agents-oriented programming (Shoham 1993), interactions of softbots (taskbots, userbots - the suffix "bot" is used for "software robot" (Com. ACM 1994) and others, have been analyzed. In this context we argue a domain-independent conceptualization of agent architecture which is based on Top-down Object-based Goal-oriented Approach (TOGA). The TOGA methodological part was inspired by Michalski, Stepp, Dontas, and Collins papers (Michalski 1983), (Stepp 1986), (Dontas 1988) and (Collins 1989), . They have proposed a main idea of the connection of an *object-based* conceptualization with *goal oriented* approach. In the plausible reasoning patterns they introduced too a reasoning generalization hierarchy. The conceptualization of artificial physical systems is an integration of Lind's (Lind 1982) MFM (Multi-level Flow Modeling) framework with, generally known in physics and engineering, processual representation of the physical phenomena. The *top-*

down approach is a generalization of the basic concepts of the structural design .

The TOGA approach being developed by Gadomski, was initially investigated in ENEA in the frame of the ISEM Project (Gadomski 1989) and, theoretically and practically analyzed in the fields of robotics and operator support systems (Gadomski 1993), (Gadomski 1994a.) Recently, the experiences acquired with the use of AIA to the modeling of emergency actors during collaborative training (Balducelli 1994) was also presented on the First and Second International Round-tables on AIA (1993, 1994).

According to the TOGA assumptions *we are able to define general architecture of AIA and basic functional schema of the domain-independent decision-making engine*.

TOGA is also a meta-tool for knowledge-based system development and can be partially confronted with the KADS methodology (Schreiber et al. 1993).,

Here only some selected information about TOGA are presented. The TOGA theory is composed of three elements:

1. Theory of Abstract Objects (TAO), which is a domain independent conceptualization system;
2. Knowledge Conceptualization System (KNOCS), which includes the axiomatic assumptions and definitions related to: conceptualization of the real world, physical realization of an abstract intelligent agent and the domains of its goal-oriented activity;
3. Methodological Rules System (MRUS) for specification of complex problems.

Fig. 2 presents the structure of the TOGA conceptualization layers.

TAO: Theory of Abstract Objects

Any theory can be considered as a frames system which enables structuring and operation over a certain class of sets. In the case of the Theory of Abstract Objects (TAO) its domain may be any numerable set Z, its elements are called *primitives* (or *vocabulary*). TAO is a frames system which enables structuring the primitives into the forms of:

- *objects;* specified by *object names*, *attributes' names*, *values*, and *value domains*.

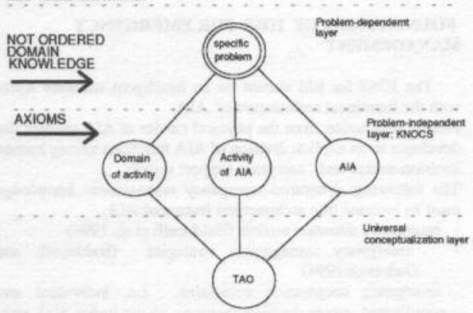

Fig.2 Three layers of the TOGA conceptualization

- *relations*; linking the objects attributes, and enabling the creation of isolated networks of objects
- *world-of-objects* (w-o-o); relational isolated networks of objects, and can be:
  * arbitrarily divided into systems and their environments,
  * aggregated in *universes of objects*.

TAO includes the definition of the class of singular objects called *agents*, and the formalization of the concept of the *point-of-view* referred to an object. The agents are particular objects which *are able* to create or to modify other objects inside their world-of-objects, i.e. this activity is not uniquely determined by the state of these w-o-o. They can be treated as "normal" objects in another word of objects. From designers perspective, TAO can be considered as a conceptual interface between KNOCS and programming languages.

KNOCS: Knowledge Conceptualization System

The Knowledge Conceptualization System (KNOCS) is a system of axioms and definitions for the description and conceptualization of the real world from the perspective of a real AIA.

KNOCS assumes that every product of human reasoning activity can be conceptualized in the framework of TAO. After TAO, KNOCS is a second conceptualization layer. It contains three fundamental frameworks of :

- *goal-oriented activity* of AIA , - *domain of activity* (d-o-a) of AIA, - AIA model.

All of them consist one terminological system. In this way, KNOCS enables conceptualization of different real-world systems such as industrial plants, robots, human operators or organizations. It also has the role of a conceptual interface between the designers and domain experts.

MRUS: Methodological Rules System

In the perspective of designers, the Methodological RUles System (MRUS) is a methodological approach to the top-down knowledge ordering for the specification of complex problems. It assumes that, at the beginning of a problem specification, the knowledge of the problem solving agent is incomplete and not goal-ordered. The problem specification activity is based on two fundamental mechanisms:

- top-down mechanism: it indicates the specification direction: from very general statements on the top abstraction levels to the details which are the elements of the particular problem solution;
- goal-driven mechanism; it always controls the links between the specified/identified object and the problem goal.

This mechanism creates bottom-up rules (synthesis rules). The rules of the MRUS system concern the meta-preferences criteria of AIA, independent on abstraction levels of problem specifications, they also enable "navigation" into the KNOCS frameworks.

Framework of Abstract Intelligent Agent

In frame of KNOCS, a general architecture and basic reasoning mechanism of simple and intelligent abstract agents, are defined. The construction of these agents if funded on the following basic concepts:

information: how situation looks (before, now, in the future)?
knowledge: how situation may be classified, and what is possible to do ?
preferences: what is more important ?
goal: what should be achieved ?

More precisely, these relative concepts always refer to a predefined d-o-a which is a source of information *(inf)*. They are defined together by the following two generic reasoning processes executed by the *preferences system (PR)* and *knowledge systems (KN)*:

$$goal = PR\ (inf1)$$
$$inf3 = KN[goal]\ (inf2)$$

where *inf-i* may be any element of the current abstract domain of activity. Abstract d-o-a can be a representation of the real d-o-a of a physical agent. In this way, *PR* produces a goal, the goal activates an adequate *KN* which produces a new information. The carrier of these processes is called *abstract simple agent* (ASA). Its architecture is defined as a triangle composed of : abstract d-o-a, preferences system, and knowledge system.

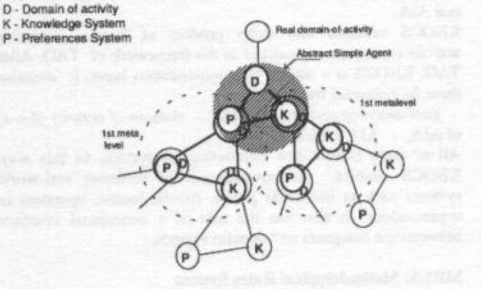

Fig.3 TOGA framework of AIA (Abstract Intelligent Agent) functional. architecture.

Let us now introduce a definition of Abstract Intelligent Agent: *AIA is an abstract agent which is able to reason about, and modify its own knowledge and preferences.*

In TOGA, AIA consists of the hierarchical pyramidal structure of ASAs. For the modifications of KN and *PR* of the basic ASA, these two systems must became abstract domains of activity for two ASAs located on the higher *meta-level*. Every next meta-level includes more ASAs. This structure is illustrated in Fig. 3.

In this way , for example, the basic conceptualization of a learning process is represented as:

$$inf\,3(KN_1) = KN_2\ [goal_2\ ]\ \ inf2(\ KN_1)$$

where KN_n is a n-th meta-knowledge,
$inf\text{-}i\,(Da_n)$ is a i-th information from the abstract d-o-a Da_n .
$goal_2$ is a goal created on the 2nd meta-level of AIA.

Decision-Making Engine, DME

Every decision-making relating to a selected Da_n requires knowledge and preferences on a higher meta-level of the AIA architecture. According to TOGA this process is defined as follows:

Decision-making (d-m) is a mental activity implied by the possibility of choice, started when either choice criteria are unknown, or alternatives are unknown, and finished when the choice is performed.

Decision is a result of this choice and refers to the state of currently analyzed domain-of-activity: This domain can be a knowledge or preferences on the AIA different meta-levels.

We should mention that choice criteria are elements of the preferences system - PR_{n+1} , and the alternatives are included in the knowledge system - KN_{n+1}. As consequence, DME must work on the *(n+2)* meta-level.

For the reason of universal, i.e. level-independent , role of DME, it has to have an ability to the activation of various problem-dependent searching, learning or optimization procedures. Methods applied used into these procedures depend on the particular properties of decisional problems.

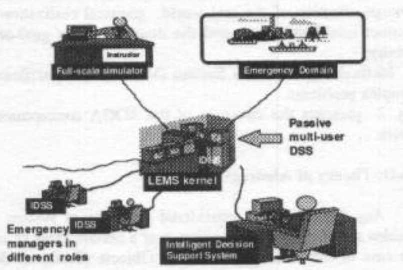

Fig. 4 Large-scale Emergency-Management System, LEMS

FOUNDATIONS OF IDSS FOR EMERGENCY MANAGEMENT

The IDSS for EM should be an intelligent software agent with the functional architecture of AIA.

Formal abstraction from the physical carrier of AIA enables the developer to an explicit division of AIA functions among human decision-maker and computer support system.

The following structured emergency management knowledge must be inserted into architectural frames of AIA:

- emergency situation models (Balducelli et al. 1994)
- emergency management strategies (Balducelli and Gadomski1994)
- emergency cooperation constrains, i.e. individual and coordinated group decision-making (d-m) under high risk, stress and time constrains (Muhanna 1993), (Watabe et al. 1992).

The above knowledge has been partially elaborated in frames of the ISEM, MUSTER and CAMS (Fantoni et al. 1994) projects.

As it is shown in Fig. 4, the main design assumption is to separate passive components of LEMS from the active personal DSS , i.e. Data Bases from Knowledge Bases.

According to TOGA, the problem of IDSS development should be divided on the following conceptual tasks:

1. Choice of a real application domain.
2. Definitions of the basic terminology and criteria for a description of the generic EM in frame of an Emergency Management Conceptualization System.
3. Top-down identification of the Generic Scenario of Emergency Management (GSEM); in terms of previously defined terminology, such as agents, actions, decisions, and domains of activity.
4. Decomposition of GSEM on model-based functions of emergency manager, and models building (every agent activity is an activation of its functions).
5. Definition of general architecture of LEMS reusing the elements of the available p-DSSs.
6. Specification of IDSS kernel architecture and its conceptual design by the choice and application of adequate software tools and methods
7. Validation of the received conceptual products.
8. Implementation of the prototype.
9. Verification and testing of the kernel with an external emergency domain simulator.

In this way, the IDSS can be realized in the open-system architecture using flexible incremental prototyping approach. Hypothetical functional structure of the IDSS is illustrated in Fig.5.

Emergency Management Conceptualization System

The experiences from the ISEM and MUSTER projects indicates that one of the basic problems of the conceptual design of decision support systems for large scale emergency management is the lack of uniform and complete conceptual web, especially on the level of generic emergency-management scenario.

Reconceptualization and unification and of the multi-disciplinary terminology and elimination of terminological redundancy in the domain of EM is indispensable task in the IDSS development. It is necessary:

- for reusing of the elements of p-DSSs ,
- for standardization of the methodology for the user-tailored domain-dependent knowledge acquisition,
- for formal verification of the completeness and congruency of the used models.

Therefore the elaboration of an Emergency Management COnceptualization System (EMCOS) with a Language (EMCOL) are here urgent tasks for knowledge engineers .

Generic Scenario of Emergency Management (GSEM)

GSEM is a network of the domain-independent activities of the manager. These activities should be modeled and linked with the implemented modules of LEMS and with specified activities of the user. GSEM must be abstracted from specific emergency domains but under constrains of human organization activities such as: cooperation, coordination and , in general, multi-agent decision-making.

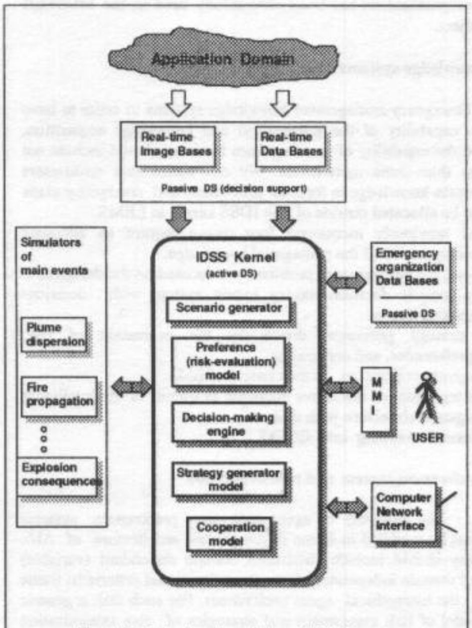

Fig. 5 Main functional modules of LEMS system.

Four basic domains of activity of emergency-agents are distinguished:

D1 - *intervention domain*; it is an abstract emergency domain and the end-domain of the manager decisions,

D2 - *data-bases domain;* it is a domain of periodically updated information, knowledge and duties. It includes logistic data, such as DBs on emergency organizations, on intervention resources, and on emergency plan and emergency procedures.

D3 - *event simulation domain*; it is a source of "what if" data about possible futures consequences of the current state of emergency and about planned interventions

D4 - *cooperation domain*; it is a dynamic domain consists of cooperating agents. It is a sources of messages necessary for the emergency manager reasoning and his decision-making activity. In general, a consensus on the communication subjects between cooperation agents must be achieved.

Real and Abstract Domains of Activity

The physical emergency domain is a domain of the goal of emergency management activities. A mental image of this domain is the direct domain of activity of the manager, there he expects to achieve particular intervention-goals. The suggested LRS conceptualization framework is described in the paper (Balducelli 1994). It is composed of three layers :

Layout layer (LL), Resources Layer (RL), Scenario Layer (SL). The SL and RL layers are mapped into LL. All of them are represented by abstract objects-relations networks. This conceptualization has been successfully used in the MUSTER Project.

Knowledge systems

Emergency-management knowledge systems in order to have the capability of the information and knowledge acquisition, and the capability of learning from the user, should include not less than three meta-levels. We can notice that multi-users domain-knowledge in form of procedures and emergency plans can be allocated outside of the IDSS kernel in LEMS.

The previously mentioned four ds-o-a require an adequate decomposition of the managerial knowledge.

Here, the key research problems are focused on the design of :

- a generic decision-making expert system with decision-making engine
- strategy generator, driven by the evaluation of risk, preferences, and constraints
- agents cooperation, in the context of EM.
- integration of the above elements in frames of the intelligent agent architecture with their
- formal mapping into GSEM.

Preferences system and risk evaluation

The hierarchy of agent emergency preferences systems must be modeled in frame of the generic architecture of AIA. They should include structured domain dependent (variable) and domain independent (invariant) decisional criteria in frame of the hierarchical agent preferences. For such task a generic model of risk assessment and strategies of risk minimization are necessary. The preferences related to risk evaluation play fundamental role in AIA reasoning processes, therefore a generic model of hazard and risk is under elaboration in frame of the TOGA frameworks.

The preferences of a particular emergency manager are role dependent but, in general, can be represented as duties, situation-dependent instructions, and moral rules/constrains. They are decomposed relatively to the basic D1-D4 activity domains, and depend on the range of managers responsibility, competencies, and assigned tasks.

Summarizing, the decision-making processes require risks and benefits evaluation in two basic contexts: direct intervention domain (emergency-domain) and communication domain. The last one concerns the same and higher level managers. For example, one of the top preferences is the request of autonomous intervention in situation of the loosing of communication link between cooperating managers.

A time-scale preferences should handle the real agent actions according to the time-constraints of the user possibilities and the dynamics of emergency domain .

The Fig. 6. illustrates a localization of main elements of the EM functional modules into the AIA structural framework.

CONCLUSIONS

This paper, according to the goal-driven paradigm, indicates only some main problems and criteria of the development of IDSS for emergency managers. Today, IDSS seems to be realistic and attractive task for interdisciplinary teams of specialists.

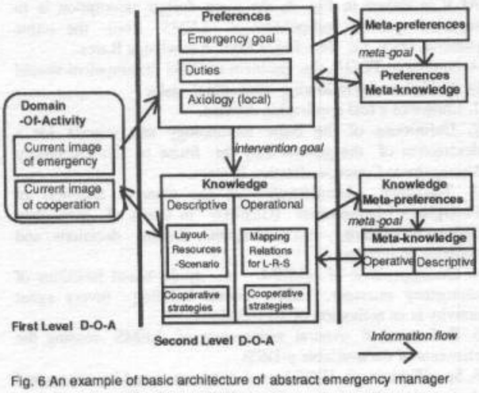

Fig. 6 An example of basic architecture of abstract emergency manager

We can notice that the application of the TOGA methodology and the suggested IDSS architecture are not only independent of specific emergency domains but they also are independent of the concrete size of the implemented data-bases, knowledge and preferences. This property should facilitate the development of personal IDSSs in the incremental prototyping manner. The personal IDSS can be expanded according to the local specific possibilities and needs. It can qualitatively support different Large-scale Emergency-Management Systems, especially where computerized information systems are just in use.

IDSS should have the capability of an explanation of its own support decisions, suggested intervention decisions, as well as a capability of evaluation of its user decisions.

From the technical point of view, the IDSS kernel should be interfaced with :

- data bases and knowledge bases containing object-based representation of the emergency physical domain (models of resources, risk objects and events), plans and procedures, all of them may be a part of conventional passive information system.
- a GIS systems for the presentation of pictorial visualizations of dynamic emergency scenarios (using maps).
- man-computer interface equipped with a set of multimedia tools allowing visualization of the real events and resources, using pictures and video-sequences.
- external simulators to simulate propagation of emergency events, such as: plume, fire, heat, etc.

In general, IDSS can be also seen as a set of expert systems hierarchically structured according to the TOGA frameworks. Partial models of the reality, everyone representing a certain single-user role-dependent viewpoint should be included.

As a final remark, we can notice that the decomposition of the role of ideal emergency manager on the roles of human user and IDSS, and continuous development of ever more detailed models of emergency management will shift human decisions on higher preferences levels, which include evaluation of the credibility of data, the application of moral and cultural criteria, as well as new consensus criteria for cooperative work.

REFERENCES

Anderson, V.; S. Bologna; G. Di Costanzo. 1993. " ISEM Information Technology Support for Emergency Management" In *Proceedings of the SRA-Europe Fourth Conference* (Rome, Italy, Oct., 1993).

Balducelli, C; S. Bologna; G. Vicoli. 1994. "Computer Simulation of Emergency Scenarios for Multi-Agents Cooperative Work" In *Proceedings of The SCS Simulation Multi-conference* (La Jolla, California, April).

Balducelli, C. and A.M. Gadomski. 1994 "Intelligent Agents in Computer Supported Training for Emergency Management". In *Proceedings of the First International Round-Table on Abstract Intelligent Agent* (Rome,Italy, Jan. 1993). Printed by ENEA.

Beaumont, I.H. 1994. "User Modelling in the Interactive Anatomy Tutoring System ANATOM-TUTOR". *User Modeling and User-Adapted Interaction no. 4: 21-45.*

Collins, A. and R.S. Michalski, R.S. 1989. "The Logic of Plausible Reasoning: A Core Theory". *Cognitive Science J.*, Vol.13, pp.1-49.

Communication of the ACM J.1994. *Special Issue: Intelligent Agents, vol.37, n.7* .

Dontas, K. and M.Zemankova. 1988."APPLAUSE: An Experimental Plausible Reasoning System". In *Methodology for Intelligent Systems, 3*. Elsevier Science Publishing Co., Inc., 1988.

Gadomski, A.M. 1989. "Specification of the problem: Information Technology support for Emergency Management using Top-down Object-based Goal-oriented Approach". *The materials of ISEM/ESPRIT Project 2322, Feb. 1989.*

Gadomski, A.M. (editor).1993 *"The Proceedings of the First International Round-Table on Abstract Intelligent Agent"*. Jan.1993, Rome. Printed by ENEA, Feb.1994.

Gadomski,A.M.; S. Taglienti; V. Nanni.1993 "Theoretical and Practical Aspects of the Modeling of Abstract Intelligent Agent : ENEA Experiences". In *Proceedings of the International Symposium on Intelligent Information Systems*, Augustow, Poland, Jul.,1993.

Gadomski, A.M. (editor).1994a *"Preprints of the Second International Round-Table on Abstract Intelligent Agent: Situation Assessment"* . Feb.1994, Rome.Printed by ENEA.

Gadomski, A.M.1994b "TOGA: A Methodological and Conceptual Pattern for Modeling Abstract Intelligent Agent." In *Proceedings of the First International Round-Table on Abstract Intelligent Agent*. Rome. Printed by ENEA, Feb.1994.

Hanks S. et al.1993. "Benchmarks, Test Beds, Controlled Experimentation, and the Design of Agent Architectures". *In AI Magazine, vol.14, no.4.*

Hawgood J. et al. 1992. "Analysis of future development opportunities for OIS methods and tools". *Decision Support System J., Vol.8,no.3.*

Kay J.1991. "Generalised User Modelling Shells - a Taxonomy". In *Proceedings of IJCAI-91 Workshop W.4 : Agent Modelling for Intelligent Interaction,pp.169-185.*

Fantoni, P. et al. 1994 "Description of the computerized accident management support system (CAMS) prototype and design" OECD Halden Reactor Project Report HWR-390.

Lind M.1982 "Multilevel Flow modeling of Process Plant for Diagnostic and Control". RISO National Laboratories Report, RISO-M-2357.

Michalski R.S. and R.E Stepp 1983 "Learning from Observation: Conceptual Clustering". In *Machine Learning: An Artificial Intelligence Approach*, Michalski, R.S., Carbonell, J.G., Mitchel, T.M.(Eds.). Mogan Kaufmann Pub. Inc., pp.331-363, 1983 .

Michalski, R.S.; L. Kerschberg; K.A. Kaufman. 1992 "Mining for Knowledge in Databases: The INLEN Architecture, Initial Implementation and First Results". *Journal of Intelligent Information Systems, Vol.1, pp. 85-113.*.

Muhanna, W.A.1993 "An Object-oriented framework for model management and DSS development" *Decision Support System J., vol.9,no.2.*

Shoham, Y. 1993 "Agent-oriented programming." *Artificial Intelligence 60 51-92.*

Schreiber, G.; B. Wielinga, J. Breuker. 1993 *KADS. A principled approach to knowledge-based system development.* ACADEMIC PRESS LIMITED

Stepp, R.E.;R.E.Michalski. 1986 "Conceptual Clustering of structured Objects: A Goal-Oriented Approach" *Artificial Intelligence*, Vol.28, pp.43-69.

Yager, R.R. 1992 "Second order structures in multi-criteria decision making." *International Journal of Man-Machine Studies 36, 553-570.*

Watabe, K. et al. 1992. "Coordination support in nemawashi decision process" *Decision Support System J., Vol.8,no.2.*