

Personoids Organizations: An Approach to Highly Autonomous Software Architectures, */a preliminary study/*

Adam Maria Gadomski

ENEA - Italian National Agency for New Technology, Energy and the Environment
C.R. Casaccia, Via Anguillarese 301, 00060 Rome (Italy),
E-mail: gadomski_a@casaccia.enea.it

ENEA, Casaccia Research Center

An extended version, April 1998

Abstract

In the work, an analysis of applicability of intelligent autonomous software units as structural elements of complex highly autonomous systems such as intelligent computer advisors and robot “minds”, is presented. Suggested web architecture of such units is based on theoretical frameworks of cognitive perspective and human organizations. Autonomous intelligent units can be seen as specific intelligent agents. They are defined according to the TOGA theory paradigms and conceptualizations, and are called personoids. A specificity of the personoid introduced relies on its, so-called, structural intelligence. Structural intelligence results from the assumption of functional architecture of every intelligent unit and it does not depend on personoid current knowledge which may be modified from outside by the user, and from inside by learning processes. Personoids do not need to be emotional, therefore simplified, and not realistic for people, and models of human organization, such as the bureaucracy model of Max Weber, can serve as a useful example of a personoids' organization oriented on the system design goal. At present, a personoid organization is considered as a possible architecture for the reasoning kernel of Intelligent Decision Support Systems employed in the industrial emergency management, as well as a MAS system of autonomous task-bots supporting management of large, complex and distributed infrastructure networks (energy, gas, services, public administration).

Keywords: *agent, intelligent agent modeling, intelligent software, software architecture, decision-making, Artificial, Intelligence, Intelligent Agent Technology, personoid, TOGA theory, preferences, knowledge, learning, Decision Support System, MAS.*

personoid: *humanlike intelligent software-being living only in computer systems and forming social structures, observable and manipulable from the exterior but without any contact with computer output/input devices.*

*[invented by Stanislaw Lem (a famous science-fiction writer):
"NON SERVIAM" from book "Insomnia", Cracow, 1971]*

1. Introduction

The actual trend in the emerging software technology is to produce systems that are always more **active** in the interactions with their users. Many approaches and strategies are currently used and tested in this research field, but probably the most fruitful technology worldwide experimented is Intelligent Agent Technology to be based on a multi-agent approach (MAS, multi agent systems). Its main issue is to distribute heterogeneous problem knowledge and its management strategies among certain number of more or less "intelligent" software agents assigned to the tasks in different activity domains, knowledge contexts, and defined on various abstraction levels.

The presented approach is based on the application of a uniform model of highly-autonomous functional units called **personoids**, and on the framework of an abstract **human organization** for the developing intelligent software systems.

The theoretical background adopted in this work is the TOGA (Top-down Object-based Goal-oriented Approach) knowledge ordering framework [Gadomski, 1990,1993,1995]. It includes ontological and epistemological requirements to *abstract intelligent agents* (AIA). Personoids are AIAs with properties and intelligent architectures specified in TOGA. TOGA also includes some general vision of the real world represented from the perspective of AIA.

In this paper we discuss some practical aspects of personoids which can be considered as structural elements of modern software systems. Personoids' Organization Framework (POF) should enable modeling of human managerial activities in frame of one conceptualization framework. The roles of a human and computer advisor are obtained by functional decomposition of an ideal manager, for example, an emergency-manager what is discussed in [Gadomski et al., 1995].

The key elements of the POF approach is an uniform architecture of the cooperating, role-dependent, autonomous personoids in frame of a functional centralized or distributed schema of organization.

This paper is an extension of the article accepted on the *11th International Conference on Mathematical and Computer Modeling and Scientific Computing* [Gadomski,1997].

Some initial intuitive definitions

personoid: *personlike; being that is person in form only, having the form of person; is formed as person.* [extrapolated from The Concise Oxford Dictionary]

person: *human being recognized as an individual in social and psychological contexts* [from The Concise Oxford Dictionary].

***form**: a structured meta-property of a thing, being, individual, behavior, change, which includes its essential features.*

***society**: an enduring set of beings which cooperate for common interests.*

***software architecture**: a structure of functional software units, defined as whole for the achieving the system design-goals.*

***autonomy** (in this context): a capability of a functional software unit to execute tasks according to its own information, preferences and knowledge / beliefs not limited by any external requests how these tasks have to be executed.*

***high autonomy**: a faculty of autonomous task execution according to its own modification strategies of the proper information, preferences and knowledge/beliefs, and adequate only to available means/resources.*

2. The concepts of Agent and Intelligent Agent

There is still big confusion in understanding, realization and application of *agents*. They are localized in various contexts, from cognitivistic modeling of human behavior, though robots and software functional components to autonomous software agents employed to selected class of tasks, for instance, information agents or mobile internet agents. The concept *agent* is used in the subject matter literature in two basic contexts.

The first is a software engineering where an agent is the softagent which interacts with only software entities in the computer software world which can be distributed over different types of computer networks. The software agents execute, in more or less autonomous, or in more or less intelligent manner, the tasks of humans and of other softagents according to the more or less human-like role-functions designed by software specialists. In this way the softagents “live” in the abstract symbolic worlds composed of programs, files, directories, drivers, all being carried by computers. They require from the human users the communication protocols expressed in terms of this world representation. In the above context, human computer interactions usually are formalized in classical software manner. Information agents, various INTERNET agents, data-base management agents are example of such understood concept of *agent*.

The key element of softagents or bots is their reciprocal communication capability in a formal ACL (Agent Communication Language) proposed by FIPA [FIPA,97] or KQML [see Internet] developed in USA .

The second context of the term *agent* is a cognitive and engineering attempt at the explanation, modeling and simulation of human mental functions. The agent or intelligent agent is analyzed as some abstraction from a human person, for the specification of various professional, social and psychological roles. Usually these agents’ environment is a vision composed with preselected aspects of the real world. The cognitive agents need to act autonomously or to support human interventions, for instance in humans decision-making processes . They “live” in various simulations or act indirectly in the physical, never completely describable domains.

Apart of their different environments, in both cases, they must model *domain of activities* and plan *actions*.

In my opinion, the personoid concept seems to be intuitively congruent and complementary to the various agent definitions being investigated in “agentology” (soft agents, intelligent agents, cognitive agents, softbot, human agent, etc., see for instance [S.Franklin, A.Graeser,1995/6]).

In the approach assumed here, we initially accept the following general functional definitions of software agent and intelligent agent:

software agent is a functional software module which uses a formal language to communicate with other software agents, is able to execute some class of external tasks, and has autonomy in its environment, during these tasks realization. It also reacts on the predefined states of its environment according to its own build-in preferences and available knowledge/beliefs.

intelligent software agent is a software agent with a capability to change and to evaluate its own preferences and knowledge, i.e. it has ability to learn and to change goals if the initial goals are not reachable.

We should notice that the last definitions of *intelligent agent* and *agent* are referred to **structural properties** of “intelligent systems”, i.e. such intelligent agent is intelligent independently on the quantity and quality its domain-knowledge/beliefs and behavior [Gadomski, Zytkow,1994]. Such **class of intelligent agents** are called *personoids*.

If we use term *agent*, its meaning refers to any software or real-world agent/intelligent-agent. For the above reason the term *personoid* has been introduced as a certain kind of abstract intelligent agents.

The TOGA concept of intelligence is founded on a structural pattern of an abstract simple agent [Gadomski,1994], this assumption is contrary to the behavior based definition of intelligence, more frequent in the subject matter literature. but it should be more efficient for the intelligent systems design. The behavioral intelligence is always visible, the structural one, if knowledge or preferences are wrong or not sufficient for task executions, can be not observable by the users.

The important advantage of this approach is that all role-agents can be constructed using the same personoid shell. Therefore the structural definition of intelligence seems to be more efficient for the design and reusing intelligent software architectures.

We should notice that TOGA is an intelligent-agent-based rather than an agent-oriented approach.

The discussed approach has evolved from the early (1989) formed the TOGA theory. It includes a general functional architecture of an **abstract intelligent agent (AIA)**. A clear distinction between a *abstract simple agent* and *abstract intelligent agent* is there performed. AIA architecture is an “essence” of a personoid, it does not depend from its physical realization.

Intelligent agents have to have an ability to **introspection**, i.e. to the observation and reasoning about their own physical and mental activities. The TOGA software simple agents are called *monads*.

3. Highly autonomous software architecture

The main idea of this paper is to develop an intelligent software architecture [Gadomski & Gadomska,90], [Gadomska,90], [Holt and Rood, 94], [Kirm, O’Hare,96] according to the personoid and human organizations structures.

An approach based on Personoids Society should enable modeling and simulation of various complex goal-oriented activities in the frame of one uniform conceptualization framework where the roles of the human and computer agents are results of allocations of functions of the predefined ideal intelligent aggregate.

The autonomy of such architecture means that its elements realize their predefined functions in the manner adequate to capabilities of their intelligent executors.

Commonly known distributed architectures of MAS [Sing,1994], [Wooldridge,1994], [Wooldridge,1995], [Internet] being developed in frame of the “naive phase” of the DAI investigations [Bond, Gasser,90] are not well suited to the design of the centralized structure of highly autonomous industrial software. They are either emphasizes the autonomy of distributed agents governed by *beliefs, wishes, desires, intentions* [Georgeff, Rao,1993] and *emotions*, or are strongly focused on agent’s communications aspects leading to the development of *agent-oriented*

languages [Finin, Weber,95], [Gaspari,Motta,1993], [FIPA,98 - Internet]. For the well defined industrial specific applications we rather require some functionally centralized architectures conceptually more similar to *human centralized organizations* than to human associations. Here, the emphasis must be putted on the roles and subordination of intelligent agents.

The intelligent software agents employed in the emergency-management support rather should *know* than *believe*. Such approach requires a redefinition of the terms as *intentions* and *freedom* in a new abstract and formal conceptualization system. More, many terms commonly used in cognitive sciences, not only are not useful in software engineering, but they should not be used by the reason of their misleading psychological feeling.

In this context we need to distinguish the meaning of the terms: *distributed* and *centralized*, what will be discussed in the next paragraph..

For example, the main functions of ADSSs (Active Decision Support Systems) [Gadomski at al.,1995] are: ***domain-information management, diagnostics, foreseen, decision-making, planning, and communication*** with human users. In the conceptualization of the POF architectures, they are transformed in *monad* and *personoid* roles.

Therefore using this framework, we are able to define two levels multi-agent where every personoid is the first level MAS system and a Personoids Centralized Organization describes the second level MAS architecture.

Personoids may use common, passive or active software tools/means/resources, which can be located inside and outside of the reasoning kernel.

4. Personoids

4.1 PARADIGMS AND ARCHITECTURE

Cog is a humanoid robot. The motivation behind creating Cog at the MIT Artificial Intelligence Lab.[Cog, INTERNET] has been the hypothesis that "Humanoid intelligence requires humanoid interactions with the world".

Personoid does not need a human-like physical body it is rather an abstraction of some functions of the human mind which can be considered as a basic entity for anyone, always goal-oriented, intelligent system.

The personoid shell is a carrier of "reasoning" frames. Of course, various reasoning "mechanisms" is possible to insert into the personoid architecture.

The TOGA personoid architecture is not intended as an exclusive unique representation of personoids as an abstract "species" but it is a concrete indication of they realization. In course of the paper, if we use the term personoids then we rather think about the TOGA personoids.

The construction of personoids is founded on elementary relations between following basic concepts:

information, i, inf: how situation looks (before, now, in the future)

knowledge, k : how situation may be classified and modeled, and what is possible to do

preferences, p : what is more important

goal, g : what should be achieved.

All mentioned concepts are relative and always refer to a predefined *domain of activity* (d-o-a) which is real or abstract. The state of d-o-a is represented by *information*.

Let us clear the meaning of some terms which are used in the subject matter literature.

domain-of-activity of the agent is the reference domain of its/his knowledge and, from

the point of view of an external observer it can be called *knowledge reference domain*.

information - a conceptualization either of the states of the d-o-a itself, or of the state of another world of objects which are symbolically represented in this d-o-a.

knowledge - an abstract carrier of reasoning processes, verified in adequate knowledge-domain

or acceptable after a rational meta-reasoning.

Knowledge has two components:

- *descriptive knowledge* (physically passive) which describes possible interrelations, states, and situations in the d-o-a, and
- *operational knowledge* (physically active) which conceptualizes/describes possible actions/ operations on the d-o-a.

belief - not verified/ falsified reasoning carrier, it is frequently used as a knowledge. More rigorous, belief may relate to all components of a personoid, i.e. to its preferences, information and knowledge.

intentions - hypothetical states of the domain-of-activity which are candidates to be intervention-goal.

All above basic concepts have *object-property*, can be aggregated and decomposed according to an abstract objects framework.

One of the fundamental TOGA assumptions is that **i, p, g, k** are defined only all together by three generic reasoning processes executed by:

- the **Domain representation System, DS**; it consists of a representation of d-o-a and conceptualization mechanism. ADS transforms signals from d-o-a in *information*. The information is memorized and is sent to the Preferences System
- the **Preferences System, PS**; it is activated by information coming from the *Domain representation System*. PS consists of preferences rule bases, PRB, and an *intervention-goal* generating mechanism.
- the **Knowledge System, KS**; it is activated by goal coming from the Preferences System. KS consists of knowledge bases and a mechanism of intervention-procedure generating.

Monad¹ is an abstract simple agent, it is a trial system composed from the above complex objects.

The interrelations between monad components are illustrated on the fig. 1.

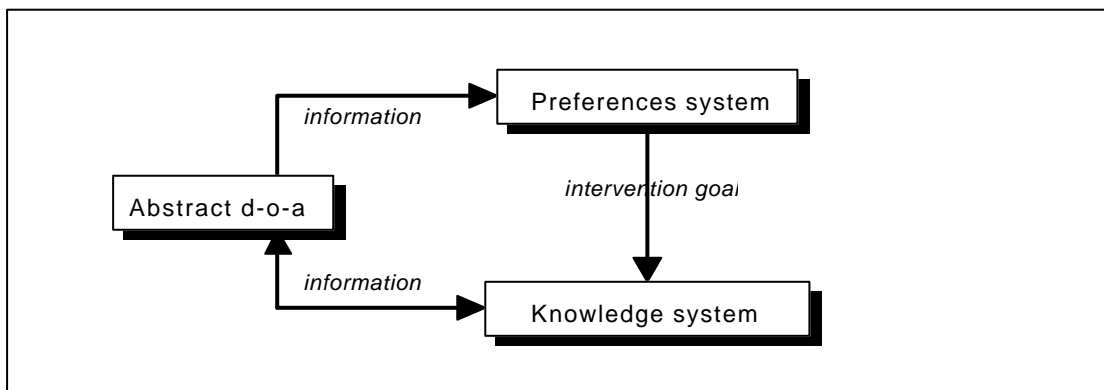


Fig. 1 Monad as a basic functional module of a reasoning process (an abstract simple agent).

They can be represented as follows :

$$inf_{11} := AD \ inf_{00}; \quad goal_{00} := PR \ inf_{11}; \quad inf_{22} := KN [goal_{00}] \ inf_{12};$$

where: := means 'becomes',

¹ **monad**: The number one, unit; ultimate unit of being in philosophy of Leibniz; simple organism assumed as first term in genealogy of living being;...[from *The Concise Oxford Dictionary*]

AD, PR and *KN* - denote mathematical operators which, subsequently, are the properties of **ADS, PS, KS** systems.

goal - denotes *intervention-goal* of the agent in *j*-th domain, and it is a control complex parameter of *KN*.

inf_{ij}, for *j*= 0,1,2,3,.. - denotes *information* about currently modified element of the domain of activity, *D_i*, from *i*-th processing level (fig.1).

D_j is a symbolic representation of the real domain of activity (d-o-a) of physical agents, this representation is an element of every ADS. In this way, *PR* produces a *goal* and this specified goal activates *KN* which produces new *information*.

Monad may modify ADS but it is not able to change its own knowledge and preferences, therefore every choice of the intervention-goal depends on the current information.

More flexible are *personoids*. In the above context, a personoid is an abstract software entity which is able to reason about, and to modify its own knowledge and preferences (i.e. to learn and to change goals).

Personoids consists of the hierarchical pyramidal structure of monads. For the modifications of *KN* and *PR* of the basic monad, these two systems must became domains of activity for two ASAs located on the higher *meta-level*. Every next meta-level may include more monads. This structure is illustrated on fig. 2. up to three levels.

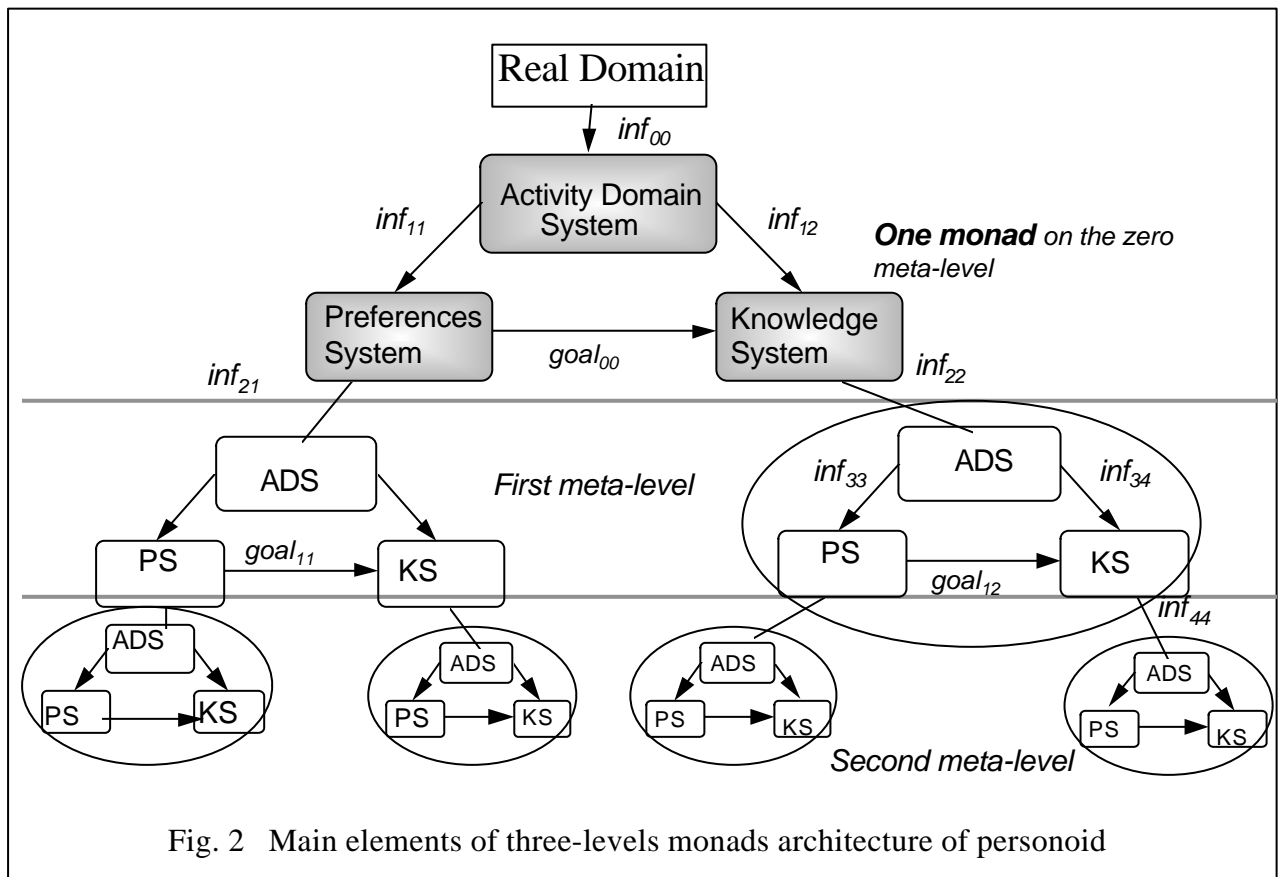


Fig. 2 Main elements of three-levels monads architecture of personoid

In this way, for example, the generic representation of learning process is the following:

$$inf_{44}(\mathbf{KN}_1) := \mathbf{KN}_2 [goal_{12}] inf_{34}(\mathbf{KN}_1);$$

where

\mathbf{KN}_n is the operator of a n-th meta-knowledge system,

$inf_{ij}(AD_n)$ is a i-th information which modifies the abstract d-o-a, AD_n ,

$goal_{12}$ is a goal created on the 1st meta-level of AIA, and \mathbf{KN}_1 can be modified by inf_{44} :

$$\mathbf{KN}_1 := \mathbf{KN}_1 \oplus inf_{44}(\mathbf{KN}_1),$$

where $X \oplus Y$ denotes the meta-operation of a *structural sum* of Y and X systems.

The presented simplified personoid model is a one-domain model.

In practice, personoids should act in two and more domains. Usually, the second domain of personoid activity is the domain of communication managed, for example, by communication monad. In this case, multi-domain personoids has three dimensional architecture. Such architecture is now under our investigation.

A simpler solution of the communication management, in the case of a one-domain personoid, is an implementation of its communication protocols into the domain Knowledge System, and communication strategies into the meta-knowledge system..

4.2 DECISION-MAKING

We need to distinguish between an *inference* and its various more or less formal descriptions.

Inference is an algorithmic process. Unique carriers of personoids' inference are their current knowledge and preferences. They are activated by information.

Let $R_k(a_1, a_2); R_{k+1}(a_2, a_3); \dots R_l(a_n, a_{n+1}); \dots$ will be a representation of an inference string, where $R_m(a_i, a_j)$ denotes a m-th ordered relational dependency between a_i and a_j , and a_i and a_j represent states or changes in d-o-a.

Then, for example, a decision-making is necessary if after $R_l(a_n, a_{n+1})$ the personoid has an available menu: $\{R(a_{n+1}, x)\}$.

Referring to the personoid architecture, a generic D-M (Decision-Making) process is defined as follows:

Decision-making is a personoid reasoning activity implied by the request/necessity of a choice caused by received information or task, started when either choice criteria are unknown (α type) or alternatives are unknown (β type), and finished when the choice is performed.

[Gadomski, 1995].

A *decision* is the result of a choice and refers to the state of currently analyzed domain-of-activity: This domain can be a personoid knowledge or preferences on different meta-levels.

The *choice criteria* are elements of the preferences system - PR_{n+1} , and the *alternatives* are included in the knowledge system - KN_{n+1} . As the consequence, D-M is performed on the $(n+2)$ meta-level. A formal model of the personoids' D-M is analyzed in the paper [Gadomski, 1995].

5. Personoids' Society and Organization

The personoid society is a system composed of individual personoids and personoids' organizations.

In classical sociological perspective there are many models of a *society* and *organizations*. In the present work, we omit the state of the art in this field. We only illustrate its main pragmatic aspects. Information Technology (IT) and the organization research can be together seen from two basic perspectives [Leinard,1977], [Flores, at al.,1988]:

- **IT tools for society**, i.e. computerization of various social and office activities , for instance: communication and information exchange by e-mail, specialized data banks, internet, managerial decisions by information systems, data processing , optimizing, simulation.
- **Society models for IT**, i.e. design of IT tools by an analogy to the selected functions of human organizations.

New AI possibilities, and, especially, agent technologies enable to extend those applications. For example, an application of abstract intelligent agents to the modeling of emergency management and environment management has been proposed in the paper [Gadomski and Gadomska, 1990]. From the autonomy and cooperation perspective, Castelfranchi deeply analyzes various social behaviors, building the models of complex interactions between various abstract-social-intelligent-agents [Castelfranchi, 1995].

Inverting the problem, for the design of an architecture for distributed intelligent artifacts, different but not all, models of an organization may be used. This context is discussed in details in the JRC Ispra report [Gadomska,1990] and the ENEA's report [Gadomska,1993]. In these works, we can notice many aspects of human organizations which could be employed in the personoids organization modeling, for example, in robotics, and in decision support systems for operators and human managers of high risk systems.

According to Max Weber (1947) an ideal type of rational organization is *bureaucracy*, i.e. an hierarchical system based on specialization and expertness. In bureaucracy, humans are seen as purely rational instruments to achieve organization goals (foundation-goals, and temporal intervention-goals). There is no conflict between individual motivations, the members roles and bureaucratic procedures. This simplified model is not very useful for real modern human organizations, but from the perspective of personoids society it may be practically applied to the distributed cooperative decision-making. Also, the concept of actors, as an ideal representation of human roles, is involved in the identification of various visions of organizations [Freeman, 1979], [Masuch, LaPotin,1989]. Application of the personoid with build-in structural intelligence and with learning capability gives a concrete carrier for the formalization of the above mentioned models.

From the perspective of DAI (Distributed Artificial Intelligence) all human organizations are always distributed. but in the theory of organization the concepts of *distributed* and *centralized* are used in the different meaning; centralized or distributed are here power, responsibility and subordination.

From such point of view, a personoids' organization can be a centralized-hierarchical, or distributed-highly autonomous, for example, military and emergency management organizations are strictly centralized in many countries. But the modern commercial, business companies and environment-management organizations tend today or have distributed open structures. In the both cases, they are goal-oriented systems where roles of the organization nodes are dependent on common system's foundation-goals.

In contrarily to the weakly connected menu-driven systems, such as CADs and text editors, where the distributed architecture is strongly suggested, in the ADSSs (active DSS) for emergency-management, the system hierarchical functions are subordinated to the system design-goals.

We argue on mixed centralized-distributed architecture based on “cooperating” personoids. It means that we can organize personoids in human-organization-like system architecture. The solutions based on various, only agent-based system-architectures have been suggested and discussed by many authors, see for example [Gadomska,90], [Holt and Rood,94], [Gadomski and DiCostanzo, 96].

A kernel of intelligent DSSs is neither distributed nor fully autonomous in the sense of the organization theory but. the metaphore of the human organization is possible to use for the separation its internal functions. Therefore, from the software system perspective, personoids perform system *functions* according to tasks received from their manager. Realization of the tasks depends on their autonomy range specified in *personoids roles*.

The role is defined by:

- ❖ *duties*,
- ❖ *responsibility*,
- ❖ *competence*, and
- ❖ *availability/access* to information.

At a consequence, behavior of personoids depends on external *tasks* and their own *motivation* (by inserted preferences) and competences (knowledge).

The formal relations between personoid roles, controlled by a supervisor/manager personoid, and the personoids' architecture is illustrated on the Fig. 3.

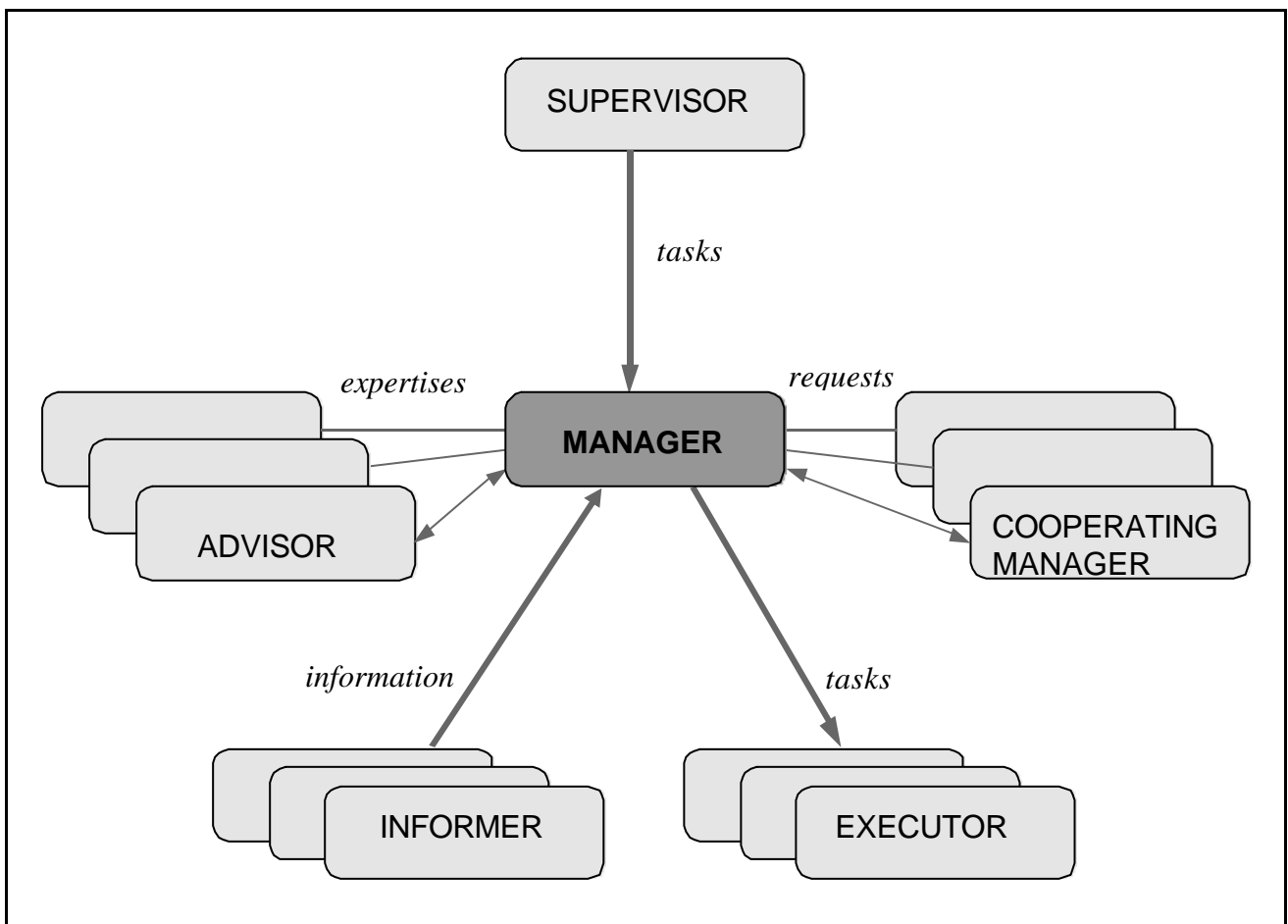


Fig.3 A structural cluster of the *subjective roles* of personoids in the POF approach.

Role can also be defined by a set \mathbb{R} of relations between domain-state and available actions.

$$\mathbb{R} = \{s, ac\},$$

where: s - denotes a perceived information, ac - denotes available activities.

\mathbb{R} is divided on 3 subsets:

- first represents *responsibility* and *duties*, and is localized in the *Preferences Systems*
- second represents *competence*, and is localized in the *Knowledge System*
- third represents *access to information*, it includes access to the domain of activity and depends on the availability of the communication channels.

Roles of intelligent agents in organization are closely connected with concept of its *functions*. both are goal-oriented. Every role requires from intelligent agents a capability for the activation of some set of functions planed by the system designer.

In such perspective, we may formalize *role* in the context of the interrelation between an organization *foundation-goal* and its *structure*. This interrelation is valid for every artificial goal-oriented system. An approach which formalizes this *Goal-System Interrelation* is called System-Process-Goal Approach (SPG) and may be used for the specification of the POF architecture for integrated plant-operator supervisory systems [Gadomski, 1988], as well as, for human organization modeling [Gadomski, Gadomska, 1990], [Gadomski, Nanni, 1993].

In the current paper we only illustrate a *subjective cluster* of the personoids roles in the POF approach. Role dependent reciprocal interrelations are domain- and system-goal- independent. They represent a point of view of every hidden DM personoid node in the personoidal software architecture. The personoids' roles are relative in frame of IDSS kernel, they could be called according to its top task and can indicate their subordination in the organization structure. We can distinguish the following relative roles of personoids:

task manager,
cooperation-manager
advisor,
supervisor,
informer, and
executor.

Their complete interactive structure is presented on the Fig. 3.

Of course, in the personoids organization network, every node can realize more then one subjective role.

6. Applications of Software Personoids Organization

A domain where we currently analyze applicability of the software Personoids Organization, is a class of Intelligent Decision Support Systems for emergency managers.

We hypothesized that an intelligent kernel for IDSS systems may be constructed employing an analogy to human centralized organizations.

The basic functions of the kernel can be divided on the personoid roles. Taking under consideration subjective reciprocal roles of the nodes, the following domain-independent (relative) roles of software modules are distinguished:

Supervisor,
Data Manager,
Consequence Evaluator,
Diagnoser,
Decision-Maker,

*Communication Manager,
Common-Knowledge Manager.*

Here, the cooperating personoids may be called:

- ❖ Consequence Evaluator,
- ❖ Diagnoser,
- ❖ Decision-Maker.

A common-knowledge manager has a role of an advisor for all before mentioned managers. Data manager has the role of an informer, communication manager is an executor, and serves for cooperating managers.

Fig. 4. illustrates an example of such agent organization in the kernel of a hypothetical IDSS (Intelligent Decision Support System).

We argue that similar applications of POF can be feasible in the case of highly autonomous robots design, for instance, for various types of intelligent mobile robots.

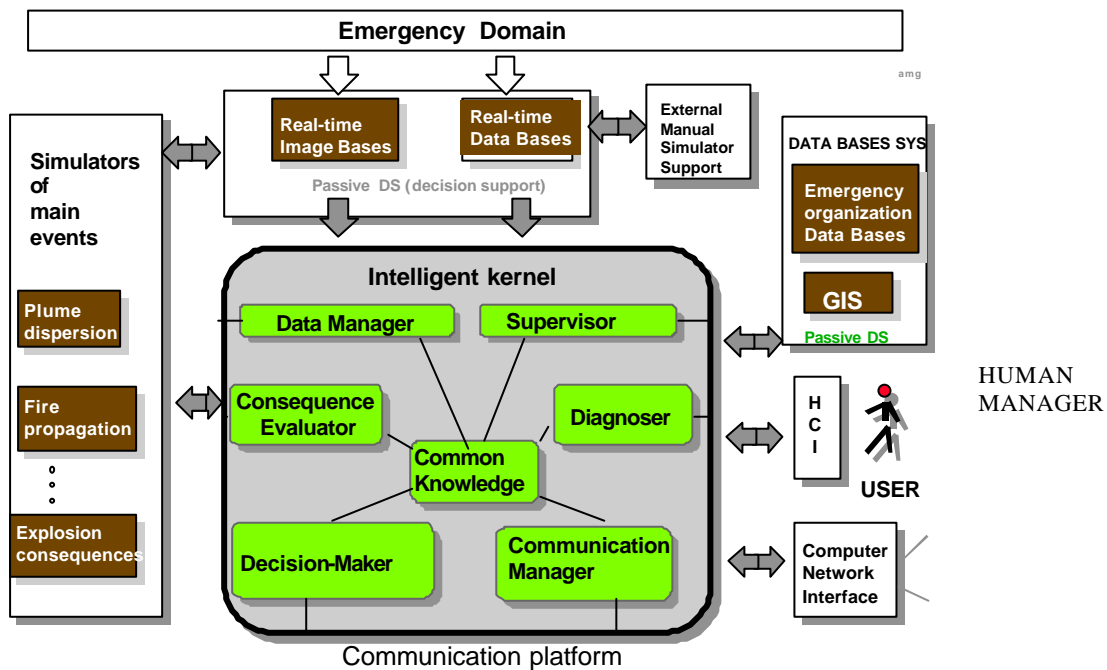


Fig. 4. An example of the general architecture and main data flow in a IDSS for emergency managers, based on a personoids' organization kernel.

7. Conclusion

Let us to summarize shortly the presented approach and its context.

1. Agents represent a very large class of *autonomous* systems.
2. Software Agents are programs or their modules, which can be, consider autonomous with capability of a communication with other agents.
3. Intelligent Agents are agents with *high autonomy*.
4. Software Intelligent Agents are specific Software Agents with high autonomy.

5. Abstract Intelligent Agents (AIA) are essence of intelligence, abstract systems that can be realized using different “materials” and techniques, for example, various software, biological materials, nanotechnologies and some others.

6. Intelligence of AIA may be defined in different manners. We distinguish two types of intelligence.

- The first is *behavioral intelligence*, i.e. if it is visible/observable then it exists - it can be realized as a sufficiently numerous set of system functions for “every occasion”. Intelligent, in this sense, agent can learn but such capability is not always necessary.
- The second is a *structural intelligence*, i.e. it does not depend on particular domain-knowledge and can only be visible if the agent knowledge and preferences are adequate to the received tasks. It needs the possibility of meta-reasoning and a reason on meta-knowledge levels. Learning and a capability to the modification of its own preferences are key for this type of intelligence.

7. The TOGA abstract intelligent agents have structural intelligence.

8. Personoids are specific software intelligent agents with structural intelligence.

9. Every personoid is composed of monads.

10. Monads are simple agents with a particular architecture of objects. These objects have: databases, knowledge-bases, associative networks, procedures and inference engines.

11. Last level (most abstract) of personoid architecture is modifiable by designer and by personoid itself.

12. Personoids can be organized in human-like organizations and societies. The structure of such macro-architecture depends on the particular objectives of the designed software systems.

13. Personoids have roles and realize the functions of the whole system autonomously. From the designer perspective, in this context, we distinguish four *fundamental design phases* in the software life cycle:

- *Specification functional architecture of the whole system,*
- *Specification of the roles of personoids,*
- *Building external tools for personoids,*
- *Instruction and teaching of personoids in a various manner.*

14. Final remarks about *personoid*:

First of all, after the Object-Oriented paradigm and after intuitive bottom-up approaches to agent architectures, a structural repetitive-incrementally generic intelligence is a new paradigm for the development of complex modifiable, “cooperating” intelligent software systems.

It is based on the exchange of messages among highly autonomous, distributed or not, software units/objects. These units should also copy with incomplete, uncertain, fuzzy data for the achieving build-in design-goals, or for the performing currently received tasks. The capability and quality of personoids actions depends on the particular construction of one monad. Its perfectioning or a modification does not change other properties of the whole system. In general, various AI technologies can be employed in the building of specific personoids roles. Multi-domain personoids architecture is currently under formal investigation.

15. About expected vantages of the *Personoids Organization Framework*:

- ◇ personoids architecture enables flexibility in agent intelligence which can be role dependent and can increase during the project development;
- ◇ the *competencies* of personoids may be perfectionned locally - the architecture may be easy analyzed;
- ◇ in the case of system malfunctioning, a proper responsible personoid may be identified by a supervisory personoid;

- ◇ application a centralized architecture of personoids organization (in sense of their roles, not in the sense of their physical distribution), enables more sophisticated, long-term planning, and coordination of whole organization, what is important, for instance, for specialized personal computer advisors of high-risk decisions.

References

- A.H. Bond and L. Grasser. *Readings in Distributed Artificial Intelligence*, Morgan Kaufmann, 1990.
- A. Rodney, Lynn, A. Stein, *Building Brains for Bodies*, MIT AI Lab Memo #1439, August, 1993.
- C. Castelfranchi. "Guarantees for Autonomy in Cognitive Agent Architecture", In M.J. Wooldridge and N.R. Jennings.(editors) *Intelligent Agents*, Springer, 1995.
- Cog: <<http://www.ai.mit.edu/projects/cog/Text/cog-robot.html>> Cog ia a "humanoid robot" being built at <<http://www.ai.mit.edu/projects/cog>, in the MIT Artificial Intelligence Lab.
- T. Finin, J.Weber at al.."Specification of the KQML Agent-Communication Language", finin@cs.umbc.edu. , 1993.
- F. Flores, M. Graves., B.Hartfield, B.Winograd. "Computer Systems and the design of Organization Interction". *ATM Transactions on Office Information Systems*, V.6,N.2, 1988.
- S. Franklin , A.Graesser . "It is an Agent, or just a Program?: A Taxonomy for Autonomous Agents", <http://www.msci.memphis.e...Prog.html#classifier>, 1995/6.
- M. Gadomska. *Sociological Theory Perspective on Computer System Design and Applications*. Commission of the European Communities, J.R.C. Ispra, ISEI, T.N.I.90.112, 1990.
- M. Gadomska. *Technology and Society: Impact and Adjustment Perspective*. Quaderni Studi,N.17, Rome, Printed by ENEA, 1993.
- A.M. Gadomski. "Application of System-Process-Goal Approach for description of TRIGA RC1 System". *Proceedings of 9th. European TRIGA Users Conference* , Oct., 1986, Roma. Printed by GA,Technologies,TOC-19, USA. 1987, also the ENEA Report RT TIB/88/2, 1988.
- A.M. Gadomski, M.Gadomska. " Environmental and emergency communication and decision: confrontation of shallow models from the point of view of computer-support designing". Materials of The Second Europe Conference of the Society for Risk Assessment , IIASA, Laxemburg, Austria (1990).
- A.M. Gadomski, V.Nanni. " Intelligent computer aid for operators: TOGA based conceptual framework". In: H.Adeli (Editor), The Proceedings of ICARCV'92 "*Second International Conference On Automation, Robotics, and Computer Vision*", Singapore ,1992a.
- A.M.Gadomski,V.Nanni. La Rappresentazione Integrata di Sistemi Artificiali Complessi: L'Approccio System-Process-Goal (SPG), The ENEA Internal Report N. RTI/INN(92)10. 1992c.
- A.M.Gadomski, V.Nanni, S.Taglienti. "Some Theoretical and Practical Aspects of Modeling of Abstract Intelligent Agent: ENEA 's Experiences", In the Proceedings of *the Second Symposium on Intelligent Information Systems*,Augustow, Poland, June ,1993.
- A.M. Gadomski: "TOGA: A Methodological and Conceptual Pattern for Modeling Abstract Intelligent Agent" . In the *Proceedings of the First International Round-Table on Abstract Intelligent Agent (AIA'93)*, Jan.1993, Rome. A.M. Gadomski (editor): Printed by ENEA, Feb.1994.
- A.M. Gadomski, J.M. Zytkow. "Abstract Intelligent Agent: Basic Concepts and Problems". In the *Proceedings of the Second International Round-Table on Abstract Intelligent Agent: Situation Assessment (AIA'94)*, Rome. A.M. Gadomski , C.Balducelli (editors).Printed by ENEA.,1995a.
- A.M. Gadomski S. Bologna, G. Di Costanzo. "Intelligent Decision Support for Cooperating Emergency Managers: the TOGA based Conceptualization Framework." *In Proceedings of "TIEMEC 1995 The International Emergency Management and Engineering Conference* . Nice, Fr, May 9-12, 1995b.

- A.M. Gadomski, "Decision Making in Emergency Management: A Model for Intelligent Decision Support System". *The document of the ENEA's research contract*, ERG-ING-TISPI, Casaccia, March, 1995c,
- A.M. Gadomski, "Personoids Organizations: An Approach to Highly Autonomous Software Architectures", The paper accepted on *11th International Conference on Mathematical and Computer Modeling and Scientific Computing, : Concurrent Engineering Based on Agent-Oriented and Knowledge-Oriented Approaches*, March 31 - April 3, 1997, Georgetown University, Washington (unpublished).
- M. Gaspari , E. Motta," AI Cooperation Strategies: An Operational View". In the *Proceedings of the First International Round-Table on Abstract Intelligent Agent (AIA'93)*, Jan.1993, Rome. A.M. Gadomski (editor): Printed by ENEA, Feb.1994.
- M.P. Georgeff, A.S. Rao. "An Abstract Architecture for Rational Agents", In the *Proceedings of the First International Round-Table on Abstract Intelligent Agent (AIA'93)*, Jan.1993, Rome. A.M. Gadomski (editor): Printed by ENEA, Feb.1994.
- C., E .Guilfoyle. Warner."Intelligent Agents: the New Revolution in Software." *The Ovum Report*, May, 1994.
- P.Harmon (editor). "Intelligent Agents. In Intelligent Software Strategies." *The Monthly Newsletter*. Cutter Information Corporation. January, 1995.
- J.Holt and M.G. Good. "An Architecture for Real-Time Distributed Artificial Intelligent Systems". *In Real-Times Systems Journ.* 6, 263-288 (1994), Kluwer Pub.
- S. Kirn, O'Hare, *Towards the Intelligent Organization*, 1996, Springer.
- S. Leinhart. *Social Networks: A developing Paradigm*. J.Wiley, N.Y., 1977.
- M. Masuch, P. LaPotin. "Beyond Garbage Cans: An AI Model of Organizational Choice". *Administrative Science Quarterly*, 34, 1989.
- R.S. Michalski., L. Kerschberg, K.A. Kaufman. Mining for Knowledge in Databases: The INLEN Architecture, Initial Implementation and First Results. *Journal of Intelligent Information Systems*, Vol.1, pp. 85-113, 1992 .
- V. Lesser (editor). *Proceedings of the First International Conference on Multi Agent Systems*, San Francisco, AAAI Press/The MIT Press,1995.
- Y. Shoham. Agent-oriented programming. In *Artificial Intelligence*, 60, 51-92,1993.
- M.P. Singh . *Multiagent Systems*, Lecture Notes in AI 799, Springer-Verlag, 1994.
- M.J. Wooldridge and N.R. Jennings. *Intelligent Agents*, Springer, 1995.