Yugoslav Journal of Operations Research
2 (1992), Number 2, 187-205

FORMAL MODEL OF ACTIVITIES AS A SEGMENT OF ORGANIZATIONAL STOCK OF KNOWLEDGE

Dubravka ĆEĆEZ-KECMANOVIĆ, Olivera KRUŠEVAC

Faculty of Electrical Engineering, University of Sarajevo, 71000 Sarajevo, Lukavica, Bosnia and Herzegovina

Radivoj PETROVIĆ

Institut "Mihajlo Pupin"
Volgina 15, 11000 Belgrade, Yugoslavia

Abstract. Institutionalized activities are performed with precise purpose or mission by legitimate role holders, according to the established norms. This is made possible by organizational stock of knowledge which incorporates typification of roles, their obligations and responsibilities, types of actions they perform, and norms governing the execution of activities. The purpose of computerized support to institutionalized activities is to increase their efficiency and effectiveness and, what is of particular importance to this kind of activities, to improve their conformity with norms. The *Formal model of activities* is created to provide a new communication medium for activity performance, to assist participants in perceiving and understanding social context and how it is affected by activity performance. The paper presents logic-based Formal models of *a simple activity type*, an *activity with recurrence* and *hierarchically structured activities.* They are illustrated by an example.

Key words and phrases: knowledge representation, logic-based modelling, activity modelling, organizational stock of knowledge

1. INTRODUCTION

There are numerous kinds of organizational activities. Planning processes, various decision-making processes, contracting, negotiating etc. are performed in all kinds of organizations. Others, like "Credit approval procedure" or "Funding of research projects", are typical for a kind of organization. The most significant determinant of all these organizational activities is their institutionalization: they are typified according to their mission or purpose; types of agents are entitled to act in certain situations; the actions of agents are expressed in particular vocabulary and are arranged in sequences, temporally structured. All these aspects are regulated by norms — socially appropriate rules of conduct.

Performance of activities is made possible by execution of actions taken by their agents. Although each action, being a linguistic expression, is a subjective act, expressing a free will of an agent (or role-holder), it obtains an objective sense in a particular social context. A type of activity, such as "credit approval" for the client of a bank, is recurrent and repeatable by different actors and its sense can be apprehended apart from its individual performances. Knowledge of activity types (institutionalized areas of conduct) and of corresponding roles to be played in all situations falling within the types, together with associated rights and obligations to execute actions, belongs to so-called organizational stock of knowledge.

Every organization has a body of socially available stock of knowledge which is generated by and transmitted to organizational members and interested subjects, [**4**]. Performance of organizational activities can be understood in terms of the knowledge that their participants have of it. This paper deals with organizational activities as a segment of organizational stock of knowledge. It reports the results of investigation of organizational stock of knowledge and possibilities to provide computer support for institutionalized organizational activities, [**8**, **9**, **10**, **11**, **16**, **27**]. The concept of **Formal model of activity type** is created which captures relevant segment of organizational stock of knowledge and provides support for planning, executing, control and evaluation of instances of the activity type.[1]

Organizational activities can be supported as a "task and task structure" or as a "process and process structure" [**18**]. Tasks are supported by decision modelling tools and techniques, including both quantitative and qualitative models. The well known examples of them are Group Decision Support Systems (GDSS), [**19**,**34**] and Electronic Meeting System (EMS) [**33**, **41**]. Process and process structure support, on the other hand, includes provision of language, communication channels and means for process structuring and conducting. There was lately a growing concern for this class of systems, that Ellis [**24**] called Group Process Support Systems or GPSS.

An interesting representative of GPSS class, named POLYMER [**6**, **7**] supports decomposition of goals into tasks, sequencing of tasks and cooperation of agents in performing their tasks. It assists agents in execution of actions in different problem situations. It may also initiate negotiation process and provide help to its participants.

A significant contribution to conceptualization of GPSS class of systems has been brought by Lee and Dewitz [**20**, **21**, **30**, **31**, **32**]. Their Computer-Mediated Contracting System supports legal conversation "on a performative network" in the process of formation and execution of contracts. It keeps track of the changes of deontic states and records their consequences such as obligations, permissions and prohibitions. Their approach and the formal model of legal reasoning have shown the fundamental difference between GPSS and GDSS kind of computer support.

In the same line of thought the ORDIT methodology for designing computer

---

[1] The Formal model of activity types is a part of the comprehensive model of organizational stock of knowledge being under development within the basic research project on "The New Information — Management Technologies", [**11**].

supported cooperative work (CSCW) deals with "the representation of the conversations that take place in an organizational context whichRrmediate the cooperative work and which bear thet2rntual creation of, commitment to, and discharge of the obligations that aries" (ESPRIT II project, [23]). It is based on Speech act theory [35] and enterprise modelling techniques [22].

The problems of GPSS and conversation aspect of CSCW [5], may in general be considered within wider, semiological framework introduced by Stamper, [37, 38, 39]. Rooted in the theory of signs — semiotics, Stamper's framework of six levels — *physical*, *empirical* and *syntactical* level related to information technologies and *semantics*, *pragmatics* and *social world* level related to human information function — provides fundamentally new approach to the whole field of information systems. The problem of GPSS is located at the level of pragmatics "concerned with relationships between signs (as meaningful utterances) and the behavior of responsible agents, in a social context" [39]. Our particular problem of institutionalized activities emphasizes signs used for action and social context in which they have meaning and produce changes. By focusing on pragmatic problems of signs we can not avoid the problems of meaning (semantics level) and social consequences.

Our approach to organizational activities [11] has been grounded on the Theory of communicative action [25], Speech act theory [3, 35, 36] and the Sociology of Knowledge [4]. The core element of computerized support to institutionalized activities is the Formal model of activity to be presented in this paper. It has been built upon Predicate logic [26], Temporal logic [1, 2, 17, 40], Deontic logic [30, 31, 32, 42], and Illocutionary logic [36], detailed discussion of which is omitted here.[2] Temporality of organizational activities and their dynamic properties are further investigated in [28, 29].

The explanation of a notion of an activity type in terms of the organizational stock of knowledge from which the participants draw their interpretations, is presented in the next section. The problems found in practical activity performances and the requirements for computerized support are discussed in Section 3. Section 4 describes the example which is used to illustrate formal modelling of an activity type. Section 5 presents the basic concepts and the Formal model of a simple activity. Further derivatives of this basic model — model with recurrence and hierarchical model — are briefly described in Sections 6 and 7. The concluding section summarizes the results of investigation and implementation of the Formal model of activity and briefly suggests its benefits in supporting practical performance of organizational activities.

2. ACTIVITY TYPE — A SEGMENT OF ORGANIZATIONAL STOCK OF KNOWLEDGE

Essential for the analysis of organizational activities is the distinction between an **activity type** and the individual practical performance of activities as **instances of the activity type**. The institutionally defined rules of conduct and

[2]Logical foundation of the Formal model of activity is elaborated in [27, 16].

conditions for successful performance of actions constitute the socially desirable model of an activity i.e. an **activity type**. An instance of an activity type consists of a sequence of actions taken by responsible agents. The instances of an activity type are not necessarily the same: they differ depending on the particular circumstances. However they have to be in accord with the established rules for performance of the activity type.

An activity type is an objectivated segment of organizational knowledge which provides necessary clue for social actions and their consequences in a social context. In the first place it specifies roles and actors playing this roles in activity performance. A role is identified by socially defined standard, which itself belongs to organizational stock of knowledge. A role assumes that one shares with others specific goals, knowledge, ethics and norms. **Roles** are **types of actors** with particular responsibilities and obligations for taking types of actions such as: *"An expert is obligated to evaluate the research proposal"*; *"The director is authorized to sign a contract"*. This shows that the standards for roles are based on typification of actions.

The **type** or **pattern of action** is a speech act, characterized by its internal form and structure, usual or generally accepted vocabulary, preconditions for its successful execution and its social consequences enforceable by organizational rules. Successful execution of an action changes the stage of performance or the **state of an activity**. The meaning of the state is derived from both the informative elements of actions leading to the state and social consequences implied by the normative context. The former — informative elements of actions — determines the **informative structure** of the state and the later — social consequences of actions — **performative structure** of the state [**14**, **16**, **27**].

When an agent, playing certain role in an organizational activity, issues a linguistic expression i.e. performs an instance of an action type, the other participants of the activity, and possibly other interested subjects, normally understand its propositional content as well as performative effect arising from it. This is possible due to collective stock of knowledge which is presupposed in all institutionalized activities.

No action in institutionalized activities can be performed without collective stock of organizational knowledge that has been socially produced and objectivated. Moreover, new experiences from actions performance are objectified and incorporated into existing stock of organizational knowledge, and in this way "transmitted" to the (present and future) actors in conversations.

3. THE ROLE OF FORMAL MODELS IN ACTIVITY PERFORMANCE

Investigation of practical performance of activities in different kinds of organizations (business and governmental) showed some typical problem situations [**10**]:

— When an agent brings about an action the question is whether s/he is authorized to do this i.e. whether s/he is legitimate role-holder entitled to perform a certain type of action;

— Who is responsible or entitled to take an action may be disputable in a particular situation (state of an activity);

— When an action is taken or is tempted to be taken it is judged according to its rightness — whether it conforms to the norm ("Is it an appropriate action type for the actual state?", "Is it executed in time?", "Are the required conditions fulfilled?");

— The meaning of an action in a social context can be controversial as to its consequences and social force it implies; this is due to vague and imprecise specification of norms;

— Any instance of an activity type may be judged with respect to its validity by legitimate authority; the available evidence is usually insufficient to judge its conformity to a norm;

— Conflicting situations may arise when two or more activities attempt to use nonsharable resource (e.g. room, equipment) at the same time or when one depends on the document produced by another;

— Planning of a particular activity and especially a number of interconnected activities is difficult due to poor prediction and lack of the historical evidence of activity performance.

The analysis of these problems motivated further investigation of a nature of organizational activities, organizational knowledge structures used in their performance, goals to be attained in the performance of activities and the needs of actors in performing and conducting the activities in practice, [**11**]. The research results collected from different kinds of organizations provided a significant empirical evidence for research of roles of formal models in activity performance and conceptualization of a type of computer based support to the actors of organizational activities.

The investigation showed that the goals of the performance of activities are: a) norm-conformity, b) effectiveness and c) efficiency. Classical data base systems, electronic message systems and document management contribute to the increase of efficiency and to some extent improve effectiveness of activities. However, norm-conformity has not been considered as the goal to which the computer support should be targeted at all.

Performance of the activities depends on participants competence and their understanding of the social context and how it is affected by their actions. The availability and transferability of organizational stock of knowledge is the basic assumption for the performance of activities according to the established norms. We conceive of the **new technological medium**, based on information and communication technologies, through which organizational stock of knowledge will be available to participants [**11**, **15**]. The knowledge assumed by the notion of an activity type is an essential segment of the organizational stock of knowledge to be captured by the **Formal model of activity type**.

Based on the empirical evidence of the performance of activities and the needs of actors the requirements for supporting the agents in performing the activities are specified and the concept of **Formal model of activity** is created to:

a) *provide a new communication medium for the utterance and execution of actions in the social environment*;

b) *assist agents in perceiving and understanding social context and how it is affected by social actions (their own and of the others)*;

c) *support agents in planning and selecting their actions by providing obligatory or alternative courses of action in a given situation*;

d) *provide the evidence for testing the rightness of a certain action and the whole activity in relation to a given norm system*;

e) *monitor performance of activities and creation and recreation of the relevant segment of organizational stock of knowledge.*

The concept of the Formal model of activity designed according to these requirements, together with its role and usage, are elaborated in the remainder of this paper.

4. EXAMPLE OF AN ACTIVITY TYPE

The problem of activity modeling and the Formal model of activity will be illustrated by an example, well known in scientific community: FUNDING OF RESEARCH PROJECTS. Different agencies for funding science perform this type of activity in more or less similar ways. We shall refer to such an agency as Governmental Scientific Foundation (GSF). It is usually headed by a director, it employs various officers for different scientific fields and it engages numerous experts for evaluation of research proposals. All parties involved in this activity — GSF responsible for funding science and the research institutions applying for funds — have their rights and obligations regulated by law. In addition GSF prescribes the rules concerning required documentation for a research proposal, deadline for applications and time constraints for particular actions. The activity is usually repeated every year.

The activity FUNDING OF RESEARCH PROJECTS starts when GSF announces conditions for applications of research projects for the particular period. The research institutions submit research proposals, which are evaluated by GSF. If a research proposal is accepted for funding GSF and the research institution sign a contract. The contract specifies mutual commitments and rights and obligations of both parties: the research institution is obliged to realize a project according to the accepted program and to report about research results to GSF; GSF has an obligation to provide financial support planned for project realization and has the right to monitor and evaluate the research results. After the completion of the research project and the acceptance of its final results the activity regularly ends. However, if the research institution does not meet contract conditions the activity ends irregularly. (The activity type "FUNDING OF RESEARCH PROJECTS" is presented in Fig. 1, which will be explained later).

The evaluation of a research proposal has several stages. First, the regularity of a proposal is checked in terms of correctness and completeness of submitted documentation. Checking period in our example is limited to 7 days after the submission of a proposal. The correct research proposal is further evaluated by an expert according to the established scientific criteria. The expert is to finish it within a month. If the research proposal is found irregular the research institution is permitted to correct it in a certain time period. If the correction is completed successfully the research proposal proceeds further. If it is not, the research proposal fails and the activity ends, this time irregularly. (The subactivity type "EVALUATING OF RESEARCH PROPOSAL" is presented in Fig. 2)

This example is somewhat simplified to fit this paper, but still expressive enough to represent different aspects of activity modelling.

5. FORMAL MODEL OF ACTIVITY TYPE

In order to perform certain activity, such as FUNDING OF RESEARCH PROJECTS, the agents take actions (e.g. "GSF officer checks correctness of a research proposal"), one after the other, and gradually proceed toward the completion of the activity. In other words, by taking actions agents change the stage or phase of performance of an activity. Since the Formal model of activity is aimed to capture knowledge of an activity type it should be able to represent basic concepts of an activity such as: agent, action, stage of performance and change of stage. In this section we define first these basic concepts, which are further used to build the Formal model of a simple activity type.

5.1. Elements of an Activity Model

An activity type is defined using the following basic concepts:

Agent

Institutionalized activities are based on typification of roles in activity performance. Roles are associated with a set of responsibilities, professional knowledge and goals, as well as norms and moral values. (The existence of role is another piece of knowledge, defined by normative system, [16], which is not discussed here). An acting person apprehended not as an individual but as a **role holder** is called **agent**. A particular person may play more then one role, e.g. may be a *professor*, a *principal investigator* of a research project and an *expert* engaged by GSF.

An agent is defined by identification of **role**, **person** and **validation period** in which the role is assigned to the particular person. The definition of a type of action refers to a role as an acting subject which means that any person holding the role is entitled to perform the action. In other words the action is to be executed by the proper agent.

Action

When an agent issues a linguistic expression with an intention to do something, e.g. "to decide to accept a research proposal for funding", and when this linguistic expression produces effects in the social environment, it is called an **action**. An action can be treated as *illocutionary speech act* which, according to Speech act theory, has its propositional content and illocutionary force, [**35**, **36**]. By issuing a linguistic expression, under certain conditions, an agent takes action and in this way changes state of affairs and creates, modifies or deletes commitments and obligations.

The execution of an action changes the stage of performance of an activity. An ordered **sequence of actions** constitutes an **activity**.

An action is determined by the agent who performs the action, the expression — performative verb together with related informative elements — defining the deed, and the time of execution. In practical performance of activities this is given in a natural language using precise terminology. The formal representation of an **action type**, in computable form, is the following:

$$\begin{gathered}\textbf{ai.k (Agent: action\_predicate } (\text{argik1}, \ldots, \text{argikz}) \textbf{ ON } \text{t}) \\ \mathbf{k} = 1, 2, \ldots, m\end{gathered} \tag{5.1}$$

where **ai.k** is denominator of the action type **k** of activity type **i**. The performative expression is given in a compact form: **action_predicate** identifies the type of act and the arguments (argik1, ... , argikz) define related informative elements, such as organizational entities, agents, documents, actions etc. Variable t refers to point in time when the action is executed. Since an **action_predicate** uniquely identifies an action type a denominator is redundant. The reason for using denominator is its short code, suitable for naming actions in graphical representation of activity.

In our example of activity "EVALUATING OF RESEARCH PROPOSAL" action types (denoted by **ai.k**) are specified in Fig. 2b. For example an action type by which GSF officer checks regularity of documentation of a project proposal, identified by Ident_no, and decides about its Regular_value (which may be "regular", "irregular" or "with minor deficiencies") is formally represented by:

a2.2. (GSF_officer: regularity_check (Ident_no, Document_no, Regular_value) **ON** t2)

The kind of action described above is so called user's action because it is performed by a user. Another kind of action is automatic or system's action. This action uses knowledge accumulated during activity performance and communicates it to social environment. System's action does not add knowledge or cause change in activity performance.

The above syntactic form of an action type enables representation of explicit knowledge brought about by action. The meaning of the action in a social context and the implicit knowledge it invokes upon execution will be considered later.

State

Meaning of stage or phase of performance of an activity is expressed by the concept of **state**. A state is identified by the **state_predicate** that describes the stage and the validation time (when the state becomes valid or true). A state becomes valid or true when all necessary conditions for its existence are fulfilled. A state is formally expressed by the clause:

$$\textbf{Si.j (state-predicate } (\text{argij1, argij2}, \ldots, \text{argjn}) \textbf{ IN } (t_{\text{BEG}}, t_{\text{END}})) \quad j = 0, 1, \ldots, n \tag{5.2}$$

where **Si.j** is denominator of state **j** of an activity type **i**, **state_predicate** uniquely identifies the state and arguments (argij1, ... , argijn) represent **informative structure** of the state.

The **informative structure** of a state consists of all information generated by the sequence of actions preceding the state. This is the knowledge accumulated from execution of actions leading to the state. If alternative sequences of actions lead to the same state, then each of them generates different informative structures of the state. Hence the arguments of the **state_predicate** are not necessarily defined in advance.

In our example of subactivity "EVALUATION OF RESEARCH PROPOSAL" (Fig. 2) the state **regular_proposal_submitted (S2.2)** can be reached by two different sequences of actions. First sequence consists of submission of research proposal by research institution (a2.1) and regularity checking by GSF officer (a2.2). When the proposal passes regularity check successfully the activity attains the state **S2.2**, having the following informative structure:

S2.2 (**regular_proposal_submitted** (Ident_no, "Proposal_title", Investigator, Period, GSF_officer, Document_no, Submission_date) **IN** $(t_{\text{BEG}}, t_{\text{END}})$)

If the research proposal has minor deficiencies the research institution is permitted to correct it. After the submission of a new, corrected version of the research proposal (a2.3) GSF officer performs regularity checking again and consequently reaches the state **S2.2**. In this case the informative structure of the state **S2.2** is not the same as in the first case:

S2.2 (**regular_proposal_submitted** (Ident_no, "Proposal_title", Investigator, Period, GSF_officer, Document_no, Submission_date, Correction-date) **IN** $(t_{\text{BEG}}, t_{\text{END}})$)

Besides bringing new knowledge into the performance of an activity, the execution of actions implies also *commitments*, *obligations*, *permissions* or *prohibitions*. The informative structure of a state is only one part of the meaning of a state, the one that can be derived explicitly from the actions leading to the state. A complementary to this part is the **performative structure** (meaning) of the state, that expresses consequences of actions leading to the state, enforceable by normative context. The consequences depend on the type of action or more precisely on its illocutionary point (in the terminology of Searle and Vanderveken [**38**]):

- If an action is **assertive speech** act its execution implies agent's commitment and "latent" obligation to provide argumentation or evidence if necessary,

- An action which is **directive speech act** (order, command, request) produces obligation for the agent to whom it is directed;
- **Commisive speech act**, such as promise, acceptance or approval, creates an obligation to perform an action in the future;
- An action which is **declarative speech act** changes social relations: establishes a new role, assigns role to a person, changes legal status of a document etc. and as a result creates obligations, permissions or prohibitions for the person to perform certain actions.

This knowledge is derived from the social context. The performative structure of a state represents the meaning of the state in the social system.

Reasoning about obligations,niRrpermissions and prohibitions is based on deontic logic, [16].

Transition

A successful execution of action causes the change of state of an activity. Necessary conditions for successful execution of action causing the change of state and consequences effected by that change are defined by the **transition rule** or **transition**. Transition rule from the state u to the state v is specified by:

$$\mathbf{T}^{\mathrm{uv}}{}_{\mathrm{i}} = (\mathbf{Si.u}, \mathbf{Si.v}, \mathbf{ai.k} \cup \varnothing, \mathbf{CD}^{\mathrm{uv}}, \mathbf{CQ}^{\mathrm{uv}}), \quad \mathrm{u}, \mathrm{v} = 0, 1, \dots, n \qquad (5.3)$$

with the following meaning:

When actual state is	**Si.u** (**state_predicate**(argiu1,ariu2, ...) **IN** (t_B, t_E))
and if action is performed,	**ai.k** (**Agent: action_predicate**(argik1,...,argikz) **ON** t)
and conditions	$\mathbf{CD}^{\mathrm{uv}}$ are fulfilled
then actual state is	**Si.v** (**state_predicate** (argiv1,argiv2, ...) **IN** (t_B, t_E))
and consequences are	$\mathbf{CQ}^{\mathrm{uv}}$.

Conditions $\mathbf{CD}^{\mathrm{vv}}$ for a succesful execution of action include authorization of the agent,time limits for exection of action (e.g. application before deadline), availability of a document which is subject of action, etc. When an action is performed it does not automatically change the state, because it is necessary that all required conditions are fulfilled.

Social consequences $\mathbf{CQ}^{\mathrm{uv}}$ implied by the change of state, derived from normative context, constitute performative structure of a state. The performative structure of a state, like its informative structure, depends on the transitions performed to reach the state.

Transition is usually initiated by user's action. In a particular case a transition may be provoked by the expiration of time, when a time limit is defined by the transition rule. This is why the set of actions **ai.k** is extended by the empty set $\varnothing$ in the transition rule (5.3). (The expiration of time may be considered an empty action.)

Specification of the transition rules of subactivity type "EVALUATION OF RESEARCH PROPOSAL" is given in Fig. 2b.

5.2. Formal Model of a Simple Activity

Let **Ai** be an activity type from a set of activity types $\mathcal{A}$ of an organization. Each activity type **Ai** is determined by its set of states $\mathcal{S}i$, set of actions ai and set of transitions $\mathcal{T}i$, hence the Formal model of an activity type **Ai** can be defined as a set of these sets:

$$\forall \mathbf{Ai} \in \mathcal{A};\ \mathbf{Ai} = \{\mathcal{S}i, ai, \mathcal{T}i\} \tag{5.4}$$

where

$$\mathcal{S}i = \{\mathbf{Si.0}, \ldots, \mathbf{Si.n}\}$$

$$ai = \{\mathbf{ai.1}, \ldots, \mathbf{ai.m}\}$$

$$\mathcal{T}i = \{\mathbf{Ti^{dl}}, \ldots, \mathbf{Ti^{rp}}\} \qquad \mathbf{d, l, p, r} = 0, 1, \ldots, n$$

and **Si.j** has the form given by (5.2), **ai.k** has the form (5.1) and $\mathbf{Ti^{dl}}, \ldots, \mathbf{Ti^{rp}}$ have the form (5.3).

The set of states $\mathcal{S}i$ has two specific states — *beginning* and *ending state*. We shall designate the beginning state **Si.0** and the ending state **Si.end**. By definition, both **Si.0** and **Si.end** are unique for a particular activity type **Ai**. There can be however several paths to reach the ending state of an activity. Some of them are regular ends while the others are irregular ends of an activity. For example: when subactivity "EVALUATION OF RESEARCH PROPOSAL" is in the state **S2.3** and if the research institution does not correct the proposals, i.e. does not execute action **a2.3** within 15 days, the subactivity neds irregularly. On the other hand, from the state **S2.2**, after the execution of **a2.4** the subactivity reaches its regular end. As a result different informative and performative structures may be associated with the ending state of an activity.

For the sake of simplicity we assume that the following holds for every state of an activity type:

$$\forall\, \mathbf{Si.j} \in \mathcal{S}i:\ \mathbf{Si.j} \neq \mathbf{Si.0}, \mathbf{Si.end} \implies \mathbf{0 < j < end} \tag{5.5}$$

meaning that index of starting state is the lowest and index of the ending state is the highest index of the state in an activity **Ai**.

The ordering of the states within an activity type is determined by the rules describing transitions from one state to another. As we have seen, each state is uniquely identified by its predicate. However, we also attach the denominator **Si.j** to each state, which in principal can be arbitrary. In order to simplify the explications and to ease perception of models we shall accept the following convention:

Each transition is performed from the state with lower index to the state with higher index i.e.:

$$\forall\, \mathbf{Ti^{uv}} \in \mathcal{T}i \implies \mathrm{u} < \mathrm{v} \quad \mathrm{u, v} = 0, 1, \ldots, \mathrm{end}. \tag{5.6}$$

We can now derive two characteristics of beginning and ending state:

$$\forall\, \mathbf{Ti^{uv}} \in \mathcal{T}i \implies \mathbf{Si.m} \neq \mathbf{Si.0} \tag{5.7}$$

meaning that, there is not transition to starting state, and:

$$\forall \mathbf{Ti}^{uv} \in \mathcal{T}i \implies \mathbf{Si.u} \neq \mathbf{Si.end} \tag{5.8}$$

meaning that, there is no transition from ending state.

Instance of an activity type

A concrete activity that is performed according to the model of activity type is called an **instance** of the activity type. An instance is created by the sequence of actions which causes the transition from the beginning state to the ending state.

Let $Inst(\mathbf{Ai})$ denotes the set of all instances of an activity **Ai** performed in an organization. Each particular instance **Ii.w** from this set is specified by the sequence of states and actions:

$$\forall \mathbf{Ii.w} \in Inst(\mathbf{Ai}): \ \mathbf{Ii.w} = (\mathbf{Si.0}, \mathbf{ai.f}, \mathbf{Si.g}, \ldots, \mathbf{ai.q}, \mathbf{Si.y}). \tag{5.9}$$

Formal model of an instance of the activitytype represents sequence of states and actions that beginstrwithstarting state of the activity **Si.0** and ends with state**Si.y**. If y = end then this instance of activity is completed.

Each instance of action is checked against the Formal model of the activity type. This allows for different conclusions: whether an action conforms to the type, and whether it can be regularly executed; what are the consequences of the execution of an action; what is the actual state of the instance of activity and what are the obligations of an agent in this state; what can or should be the next action, how the ending state can be reached, etc.

The recording of an instance is immediately produced when the actions are executed. This enables monitoring of ongoing instances of activities and their mutual interdependence.

The evidence of execution of instances in the above form is necessary for subsequent analysis and evaluation of activities. It may also be put in archive in order to collect some statistical data, to improve the formal model, (more details about it to be found in [**27**]).

5.3. Graphical Representation

The number of states and transitions determines the complexity of the Formal model of activity type. The number of states in real life activities usually exceeds twenty and may often be 50, 70 or more. Therefore formal representation of a model given above, and particularly its computer applicable version, may be very complex and difficult to perceive. In order to enable comprehensive and compact representation, suitable for human users, we developed a graphical language for representation of activities — ACT NET (ACTivity NETwork)[3] [**14**, **15**, **27**, **28**].

---

[3]ACT NET is generalization of State transition diagram and PERT diagram, similar to PETRI net but considerably simpler, [**27**].

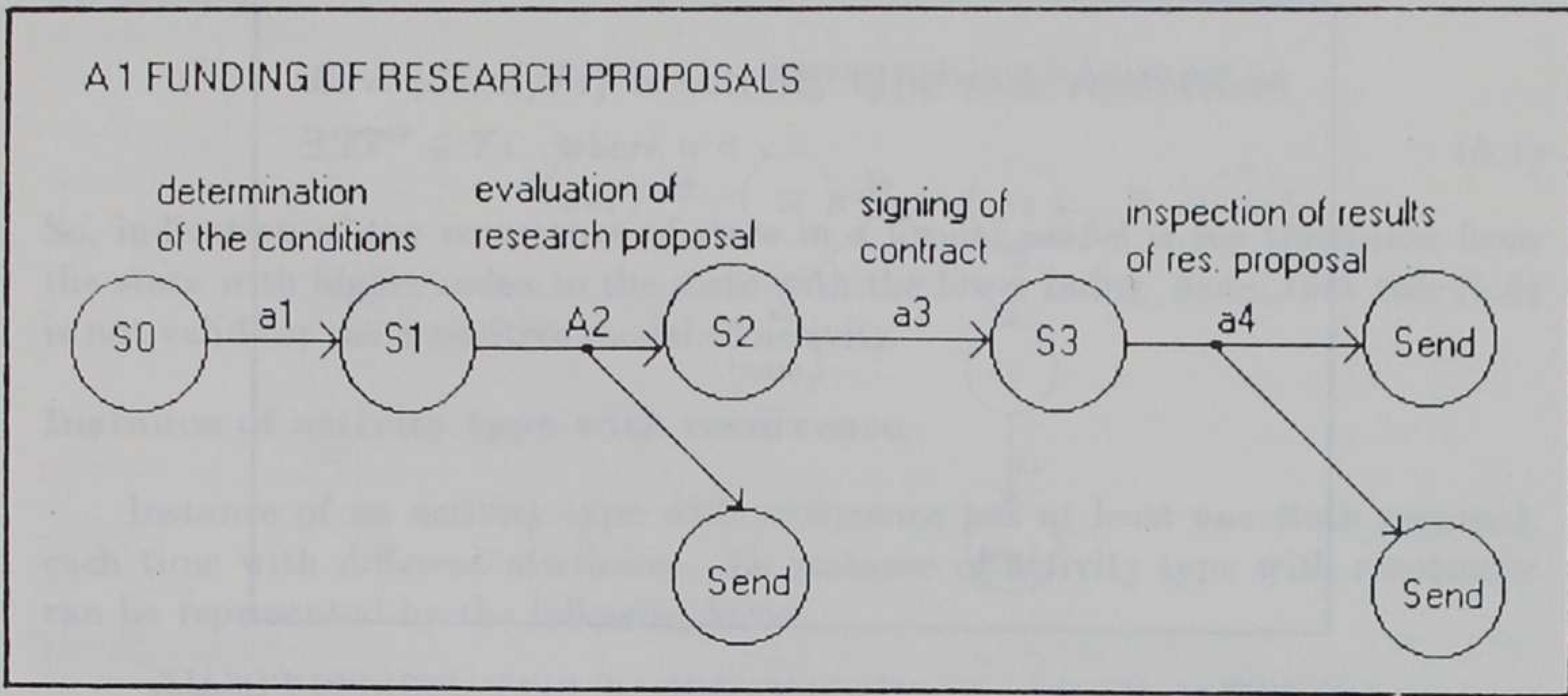


Fig. 1. ACT NET for the activity type "FUNDING OF RESEARCH PROJECTS"

ACT NET is a finite oriented graph in which a state is a node — *circle labeled by state denominator*, and a transition is *directed arc labeled with action denominator*. ACT NET of an activity type gives an overall view: by indicating type of actions and states and omitting all the details of informative and performative kind. For those familiar with a full content of an activity type ACT NET is a compact, transparent representation, easy to perceive, virphich is very useful in designing and redesigning activity types. ACT NET of the activity type FUNDING OF RESEARCH PROJECTS is represented in Fig. 1 and 2.

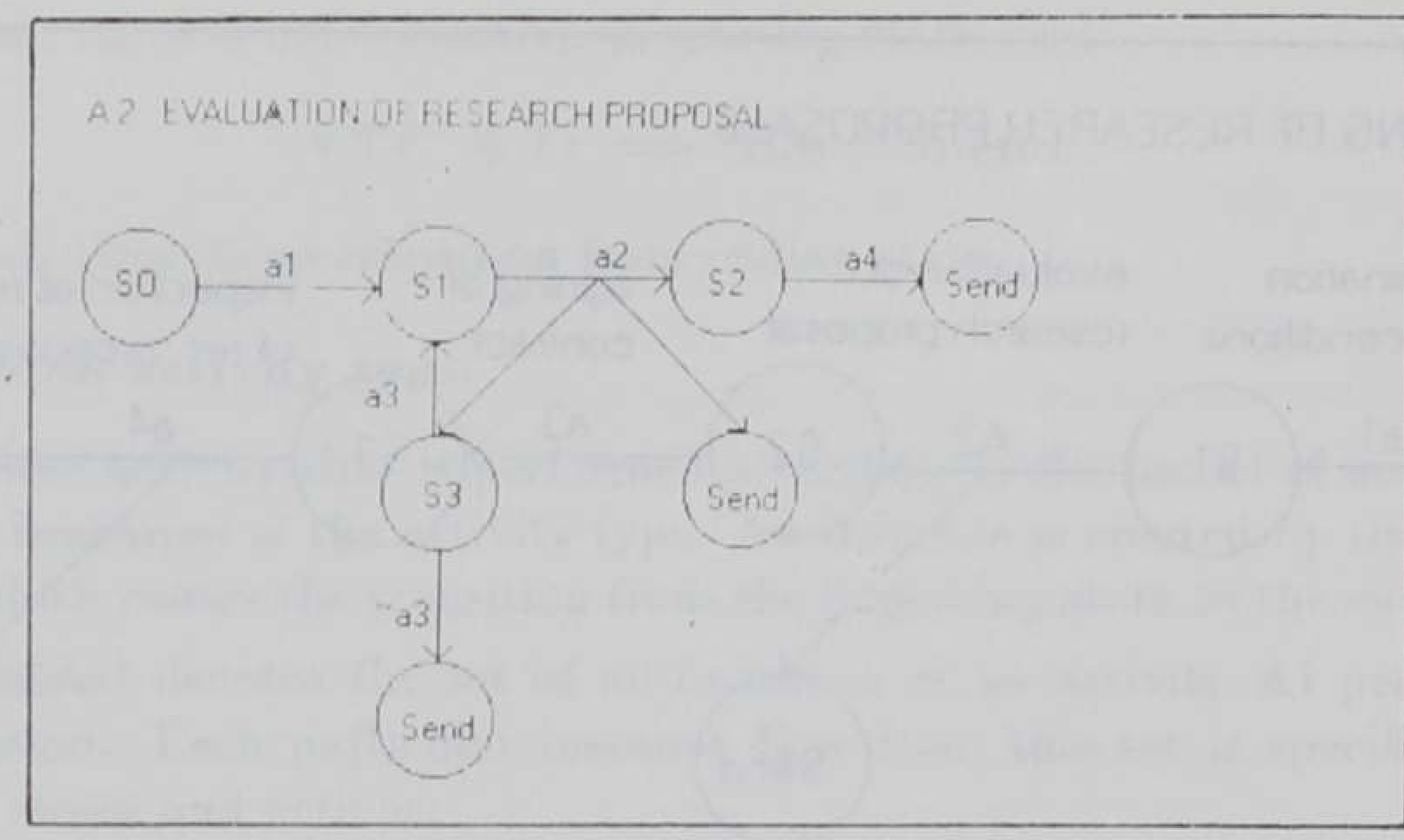
When an instance is presented by ACT NET it looks as a chain from beginning state to an intermediate or ending state. (Two instances of subactivity A2 are represented in Fig. 3). It enables visual monitoring of ongoing instances of activities.

The formal model of a simple acitivity type, presented in this section, will be further extended to represent activity type with recurrence and hierarchical model of an activity type in the following sections.

6. FORMAL MODEL OF ACTIVITY TYPE WITH RECURRENCE

There are a lot of organizational activities in reality, where the subset of actions and states recurs under certain conditions. These activities are called *activities with recurrence*. Consider the example of subactivity "EVALUATION OF RESEARCH PROPOSAL" when GSF officer checks regularity of a research proposal (**a2.2**) and founds that it has minor deficiencies, which causes transition from **S2.1** to **S2.3**. The research institution performs a correction (**a2.3**) which produces transition form **S2.3** to **S2.1** (see Fig. 2a). In this case the sequence (**S2.1**, **a2.2**, **S2.3**, **a2.3**, **S2.1**) occurs once, and may be repeated several times.

According to the formal model of activity, recurrence means that there exists the transition to some previous state i.e. to the state that has already been achieved.



a. ACT NET of subactivity type "EVALUATION OF RESEARCH PROPOSAL"

```
a1 (Research_institution: submit_proposal (Ident_no, "Proposal_title",
                                  Investigator, Period) ON t1)

a2 (GSF_officier: Regularity_check (Ident_no, Document_no, Regular_value)
                                  ON t2)

a3 (Research_institution: Correction_of_proposal (Ident_no) ON t3)

a4 (Expert: Evaluation_of_proposal (Ident_no, Result_of_evaluation,
                                  Priority_of_project) ON t4)

T^{0,1} = (S0,S1,a1,(t1 <= Deadline),{OBLIGATE(GSF_officier,Regularity_check)
                                  WITHIN 15 days})

T^{1,2} = (S1,S2,a2,(Regular_value = Yes),{OBLIGATE (Expert,Evaluation_ of
                                  proposal) WITHIN 15 days})

T^{1,3} = (S1,S3,a2,(Regular_value = Minor_deficiannces),{PERMIT (Research_
                    institution,Correction_of_proposal) WITHIN 15 days})

T^{1,end} = (S1,Send,a2,(Regular_value = No), {Action (Cancelation_of
                    procedure) ON TIME(Send),Document(Cancelation_of
                    procedure) ON TIME(Send),Result_of_activity =
                                  IRREGULAR})

T^{3,end} = (S3,Send,ø,(NOW > Deadline), {Action(Cancelation_of_procedure)
                    ON TIME(Send), Document(Cancelation_of_procedure)
                    procedure) ON TIME(Send), Result_of_activity =
                                  IRREGULAR})

T^{3,1} = (S3, S1, a3, (t3 <= Deadline), ø)

T^{2,end} = (S2, Send, a4, ø, {Result_of_activity = REGULAR, Document (Expert_
                                  report) ON TIME(Send)})
```

b. Examples of action types and transitions

Fig. 2. Formal model of subactivity type "EVALUATION OF RESEARCH PROPOSAL"

An activity:

$$\mathbf{Ai} = \{\mathcal{S}i, ai, \mathcal{T}i\} \text{ is activity type with recurrence}$$
$$\exists\, \mathbf{Ti}^{uv} \in \mathcal{T}i \quad \text{where } u < v. \tag{6.1}$$

So, indication of the recurrence of state in a formal model is the transition from the state with higher index to the state with the lower index. Note, that rule (5.6) is not valid for the repetitive model of activity.

Instance of activity type with recurrence

Instance of an activity type with recurrence has at least one state recurred, each time with different attributes. An instance of activity type with recurrence can be represented by the following form:

$$\mathbf{Ii.w} = \{\mathbf{Si.0}, \mathbf{ai.f}, \mathbf{Si.g}, \ldots, \mathbf{ai.h}, (\mathbf{Si.e}, \ldots, \mathbf{ai.y}, \mathbf{Si.x})^{*}, \mathbf{ai.l}, \mathbf{Si.z}\} \tag{6.2}$$

$(\mathbf{Si.e}, \ldots, \mathbf{ai.y}, \mathrm{Si.k})^{*}$ represents sequence of states and actions repeated at least ones. Second instance in Fig. 3 is the one with recurrence.

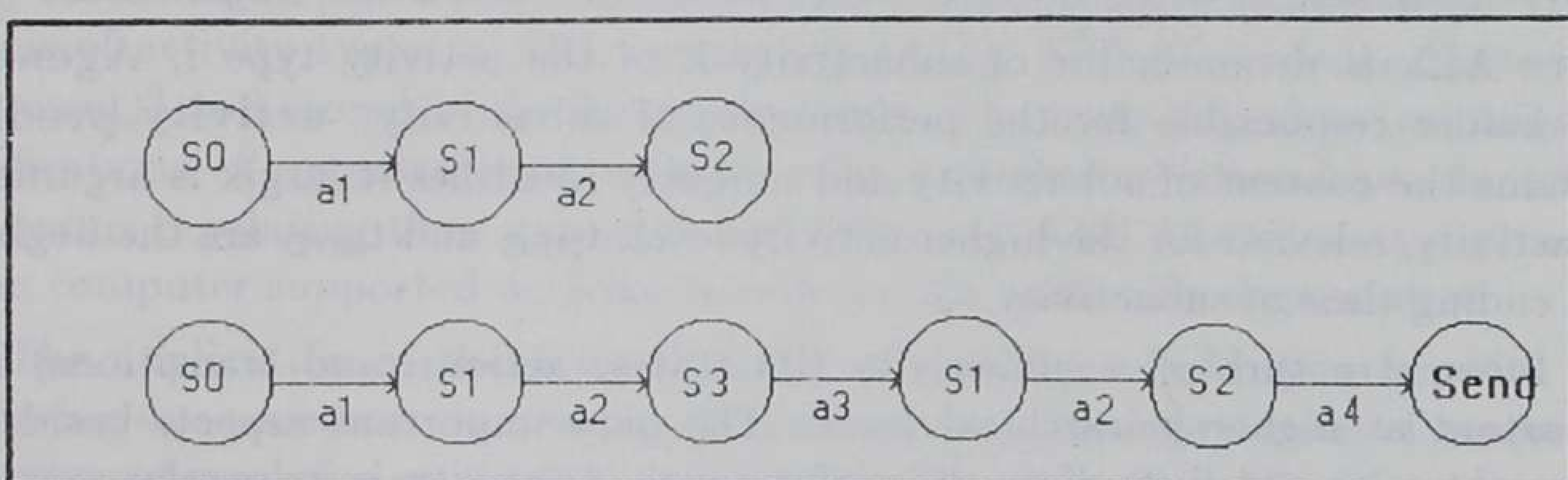


Fig. 3. Instances of an activity type

7. HIERARCHICAL MODEL OF ACTIVITY TYPE

One of the most important characteristic of organizational activities is their complexity. In order to cope with complexity organizational members create hierarchical structures of activities. An activity may be composed of subactivities, described at the lower abstraction level. Conversely an activity defined at a certain level can be subactivity of the activity at the higher hierarchical level. (For example activity "EVALUATION OF RESEARCH PROPOSAL" represents subactivity of the activity type "FUNDING OF RESEARCH PROJECTS", Fig. 1 and 2). This kind of hierarchical structure usually corresponds to role structures. Responsibility for the whole activity may be assigned to a role at a certain level, while responsibility for a subactivity to a role at lower level.

Although any structure of activity may be represented by a flat model of a simple activity type, a hierarchical structure requires more differentiated model. Users at the particular hierarchical level "see" the activity at their abstraction level. They consider subactivity (defined at lower level) as a single action and do

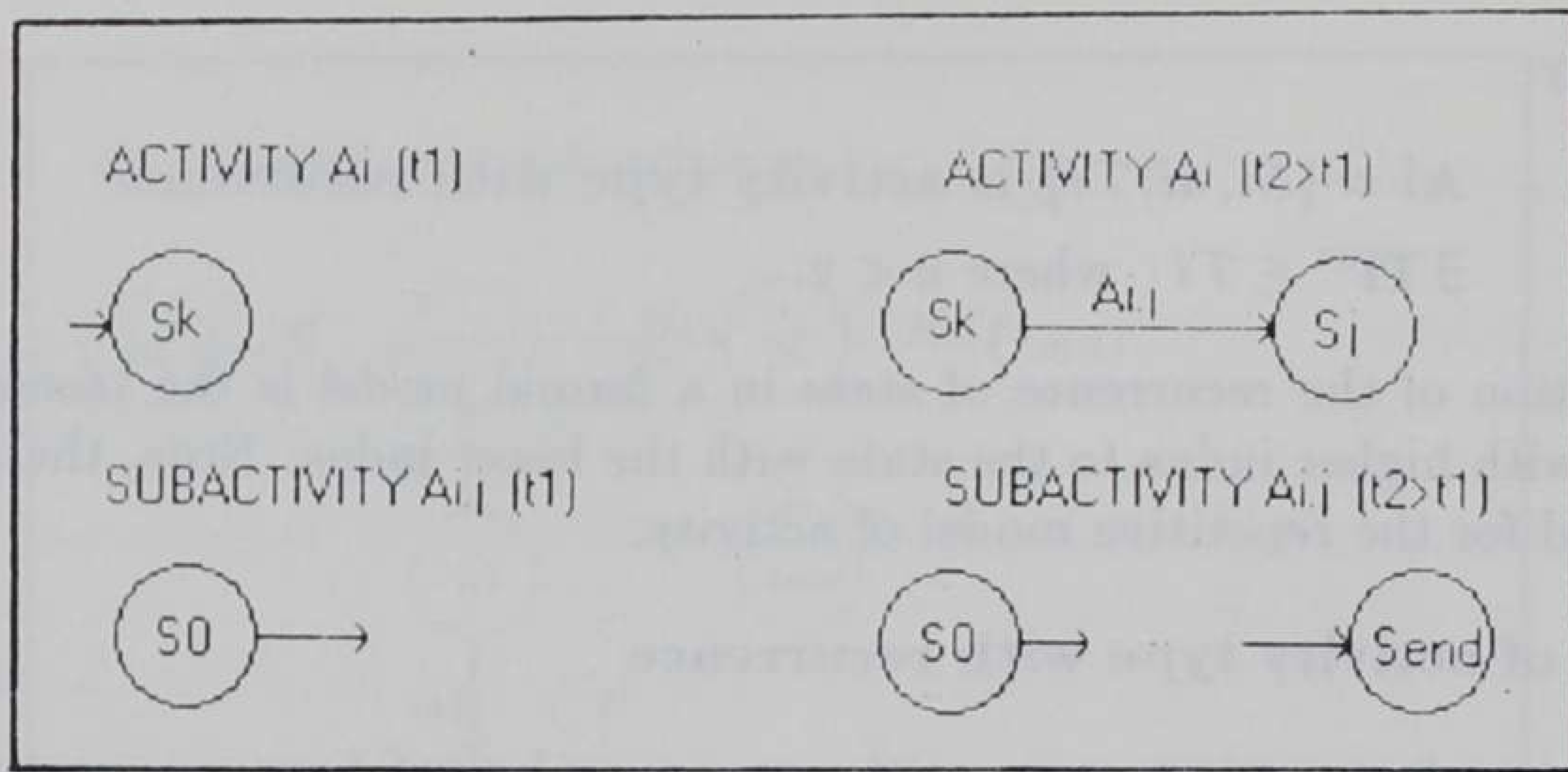


Fig. 4. Instances of composite activity type

not recognize its inner structure. This suggests that the Formal model of activity should be able to portray the actual hierarchy of an activity type.

In order to represent a subactivity as an element of the activity type we shall use the following syntactic form:

Ai.k (**Agent: activity_predicate** (argik1, ... ,argikz) **IN** (t_{BEG}, t_{END}) (7.1)

where **Ai.k** is denominator of subactivity **k** of the activity type **i**; **Agent** is a role holder responsible for the performance of subactivity: **activity_predicate** explains the content of subactivity and uniquely identifies it; argik is argument of subactivity, relevant for the higher activity level; t_{BEG} and t_{END} are the beginning and ending time of subactivity.

Inner structure of a subactivity (its states, actionstrand transitions) is not important at higher heirarchical level. The only important aspect, besides the argument values "delivered" to the performance of activity, is its regular execution, within defined time limits.

An activity type is called *composite* if it contains at least one subactivity. Internal structure of a composite activity type can therefore be represented by:

$$\mathbf{Ai} = \{\mathcal{S}i, ai, \mathcal{A}isub, \mathcal{T}i\} \tag{7.2}$$

$\mathcal{A}isub$ is subset of the set of activities $\mathcal{A}$, i.e. $\mathcal{A}isub \subset \mathcal{A}$ (**Ai** doesn't belong to $\mathcal{A}isub$); $\mathcal{S}i$, ai and $\mathcal{T}i$ represent sets of states, actions and transitions as in the simple activity model. Every subactivity has its internal structure:

$$\forall\ \mathbf{Ai.k} \in \mathcal{A}isub:\ \mathbf{Ai.k} = \{\mathcal{S}i.k, ai.k, \mathcal{T}i.k\} \tag{7.3}$$

where $\mathcal{S}i, k$, $ai.k$ and $\mathcal{T}i.k$ represent respectively sets of states, actions and transitions of the subactivity **Ai.j**. These sets are different from states, actions and transitions of the composite activity **Ai** e.g.

$$\mathcal{S}i, k \cap \mathcal{S}i = \varnothing, \quad ai.k \cap ai = \varnothing \quad \text{and} \quad \mathcal{T}i \cap \mathcal{T}i.k = \varnothing. \tag{7.4}$$

A subactivity of an activity behaves as an action since it initiates transition from one state to another:

$$\forall\, \mathbf{Ai}.k \in \mathcal{A}isub,\ \exists\, \mathbf{Ti}^{uv} \in \mathcal{T}i:\ \mathbf{Ti}^{uv} = (\mathbf{Si.u}, \mathbf{Si.v}, \mathbf{Ai.k}, \mathbf{CDi}^{uv}, \mathbf{CQi}^{uv}). \tag{7.5}$$

Instance of a composite activity and instance of its subactivity are interconnected in the following way. If subactivity **Ai.j** provokes a transition from **Si.u** to **Si.v**, then the main activity **Ai** remains in the state **Si.u** until **Ai.k** riches its ending state. Then state **Si.u** finishes and **Si.v** becomes valid. This situation is illustrated in Fig. 4.

8. CONCLUSION

The Formal model of activities, presented above, enables capturing of a part of organizational stock of knowledge required for the performance of institutionalized activities. The formal, logic-based language enables representation of both socially available knowledge and users's actions that invoke and create this knowledge. Owing to its declarative nature the formal language allows evolutionary building of the Formal model of concrete activities in organizations and their permanent modifications that take place in the social system. Moreover, the Formal model of activities is aimed to provide the new medium for social creation and recreation, distribution and usage of a part of organizational stock of knowledge.

The computer implemented version of the formal logic-based language for activity modelling is called ARL language, [**12**, **13**, **27**]. Specifications of the Formal models of activity types in ARL are interpreted by KSE (Knowledge System Environment)[4] and stored in the knowledge base. The knowledge base contains also specifications of organizational entities, roles and their relationships, documents and norms regulating their existence and behavior. KSE functions as a generator of the computer supported activity system for the particular organization.

The application of the Formal model of activities could gain benefits from improving due processes, increasing the consistency of interpretation of norms, discovering and resolving contradictions in the performance of an activity and possible conflicts among interconnected activities, improving predictions of future dynamics of an activity and a number of interconnected activities, and increasing the efficiency of ongoing activities. The evidence of performance of activities in practice and its comparison with Formal models may further support organizational learning and initiate restructuring and redesigning of activity models.

The research of organizational activities and their formal modelling in the future will shift toward less normatively regulated activities, concerned with understanding, argumentation, sense making, and negotiation.

REFERENCES

[1] J.F. Allen, *Towards a general thery of action and time*, Artificial Intelligence **23** (1984), 123–154.

[2] J.F. Allen, *Maintaining knowledge about temporal intervals*, Communication of the ACM **26** (1983), 832–842.

[3] J.L. Austin, *How to DO Things With Words*, Harvard University Press, Cambridge, 1962.

---

[4]KSE is exprimentally implemented on VAX and RDB/VMS, using programming language MODULA-2, [**11**].

[4] P.L. Berger and T. Luckmann, *The Social Constructuin of Reality — a Treatise in the Sociology of Knowledge*, Anchor book (Doubleday, 1966).

[5] L.J. Bannon and K. Schmidt, *CSCW: Four characters in search of a context*, in: *Proceedings of the First European Conference on Computer Supported Cooperative Work*, 1989.

[6] W.B Croft and L.S. Lefkowitz, *Task support in an office system*, ACM Transactions on Office Information Systems **2** (1984), 197–212.

[7] W.B. Croft and L.S. Lefkowitz, *Knowledge-based support of cooperative activities*, Proc. of the Twenty-first Annual Hawaii International Conference on System Science, 1988.

[8] D. Ćećez-Kecmanović, *Organizational knowledge systems: an investigation on knowledge communication uncertatinty*, Computer Networks and ISDN Systems **14** (1987), 221–230.

[9] D. Ćećez-Kecmanović, *Morphological approach to organizational knowledge systems*, in: M. Carnevalle, M. Lucertini and S. Nikosia (eds), *Modelling the Innovation: Communications, Automation and Information Systems*, North-Holland, 1990.

[10] D. Ćećez-Kecmanović, *Activity support systems*, IFORS Conference on Decision Support Systems, Bruges, Belgium, 1991.

[11] D. Ćećez-Kecmanović, *New information management technologies*, Progress report of the basic research project, Conference PRODUCTICA, Sarajevo, april, 1992.

[12] D. Ćećez-Kecmanović, V. Sretenović and O. Kruševac, *Formal models supporting organizational activities*, International Symposium COMPUTER AT THE UNIVERSITY, Cavtat, Yugoslavia, 1990.

[13] D. Ćećez-Kecmanović, V. Sretenović, O. Kruševac and A. Sadiković, *Activity representation language — the means for formal communications*, Mediterranean Electrotechnical Conference MELECON'91, Ljubljana, Yugoslavia, 1991.

[14] D. Ćećez-Kecmanović et O. Kruševac, *Modélisation de la Conversation Formelle*, ERGO.IA'92, Ergonomie et Informatique Avancée, Biarritz, France, 1992.

[15] D. Ćećez-Kecmanović, O. Kruševac, *Računarski posredovna konverzacija*, XXXVI Konferencija ETAN 1992, Kopaonik. 1992.

[16] D. Ćećez-Kecmanović, O. Kruševac, I. Stanišević and A. Sadiković, *Norms modelling in organizational knowledge systems*, The Third International Conference: Informtion Systems Developers Workbench: Methodologies, Techniques, Tools and Procedures, Gdańsk, Poland, 1992.

[17] L.T. Dean, D.V. McDermott, *Temporal data base management*, Artificial Intelligence **32** (1987), 1–55.

[18] A.R. Dennis, C.K. Tyran, D.R. Vogel, and J.F. Nunamaker, Jr., *An evaluation of electronic meeting systems to support strategic management*, in: J.I. DeGross, M. Alavi, and H. Oppelland, H. (eds), *Proc. of the Eleventh International Conference on Information Systems*, Copenhagen, 1990.

[19] G.L. DeSanctis and R.B. Gallupe, *A foundation for the study of group decision support systems*, Management Science **33** (5) (1987), 589–609.

[20] S.D. Dewitz, *Contracting on a performative network: using information technology as a legal intermediary*, in: R.K. Stamper, P. Kerola, R. Lee and K. Lyytinen (eds), *Collabortive Work, Social Communications and Information Systems*, Proceedings of IFIP TC8 Working Conf., North-Holland, 1991, 271–293.

[21] S.D. Dewitz and R.M. Lee, *Legal procedures as formal conversations: contracting on a performative network*, Proceedings of the Tenth International Conference on Information Systems, Boston, Massachusetts, 1989.

[22] J.E. Dobson and M.J. Martin, *Modelling the real issues in dependable communications systems*, in: W.J. Quirk (ed), *Safety of Computer Control Systems (Safecomp 86)*, Pergamon Press, Oxford, 1986, 123–128.

[23] J.E. Dobson, M.J. Martin, C.W. Olphert, and S.E. Powrie, *Determining requirement for CSCW: the ORDIT approach*, in: R.K. Stamper, P. Kerola, R. Lee and K. Lyytinen (eds), *Collabortive Work, Social Communications and Information Systems*, Proceedings of IFIP TC8 Working Conf., North-Holland, 1991, 333–353.

[24] S.C. Ellis, *The socialization of computers*, in: R.K. Stamper, P. Kerola, R. Lee and K. Lyytinen (eds), *Collabortive Work, Social Communications and Information Systems*, Proceedings of IFIP TC8 Working Conf., North-Holland, 1991, 373–385.

[25] J. Habermas, *The Theory of Communicative Action, Vol. 1.*, T. McCarty (trans.), Beacon Press, Boston, 1984.

[26] R. Kowalski, *Logic for Problem Solving*, North-Holland, 1979.

[27] O. Kruševac, *Logic based model of organizational activities — dynamic aspect*, Master Thesis, Faculty for Electrical Engineering, University of Belgrade, Belgrade, 1992.

[28] O. Kruševac, D. Ćećez-Kecmanović, *Temporalno rezonovanje kao podrška organizacijskim aktivnostima*, XIX Simpozijum o operacionim istraživanjima SYMOPIS'92, Beograd, Jugoslavija, 1992.

[29] O. Kruševac, D. Ćećez-Kecmanović, *Temporal structures in formal model of activity*, submitted to Yugoslav Journal of Operational Research YUJOR, 1992.

[30] R.M. Lee, *International contracting — formal language approach*, Proc. of the Twenty-first Annual Hawaii International Conference on System Science, 1988.

[31] R.M. Lee, *A logic model for electronic contracting*, Decision Support Systems **4** (1988), 27–44.

[32] R.M. Lee and S.D. Dewitz, *Finding international contracting opportunities: AI extensions of EDI*, in: J.I. DeGross, M. Alavi, M and H. Oppelland (eds), *Proc. of the Eleventh International Conference on Information Systems*, Copenhagen, 1990.

[33] J.F. Nunamaker, Jr., D. Vogel, A. Heminger, B.B. Martz, R. Grohowski, and C. McGoff, *Experiences at IBM with group support systems: a field study*, Decision Support Systems **5** (2) (1989), 183–196.

[34] L.S. Richman and S.L. Jarvenpaa, *Computer support for groups: a search for theoretical models*, Twenty-second Hawaii International Conference on Systems Science **3** (1989), 310–320.

[35] J.R. Searle, *Speech Acts — An Essay in The Philosophy of Language*, The University Printing House, Cambridge, 1969.

[36] J.R. Searle and D. Vanderveken, *Foundation of Illocutionary Logic*, Cambridge University Press, 1985.

[37] R.K. Stamper, *A logic of social norms for the semantics of business information*, in: T.B. Steel and R. Meersmann (eds), *Database Semantics*, North-Holland, Amsterdam, 1985.

[38] R.K. Stamper, *Semantics*, Chapter 3 in: R.J. Boland and R.A. Hirschheim (eds), *Critical Issues in Information Systems Research*, John Wiley and Sons, 1991, 43–78.

[39] R.K. Stamper, *Signs, organizations and information systems*, Australian National Information Systems Conference, Wolongong, 1992.

[40] L. Turner, *Logic for Artificial Intelligence*, Cambridge University Press, 1984.

[41] D. Vogel, J.F. Nunamaker Jr., J.F. George and A.R. Dennis, *Group decision support systems: evolution and status at the university of Arizona*, in: R.M. Lee, A.H. McCosh and P. Megliarese (eds), *Organizational Decision Support Systems*, New York: North Holland, 1989, 287–305.

[42] G.H. von Wright, *An essay in deontic logic and the general theory of action*, Acta Philosophica Fennica **21**, North-Holland, Amsterdam, 1968.