1990
International System Dynamics Conference

Manager Training Environment For Setting Complex Problems.

D.J. de Tombe

July 1990
University of Utrecht
Faculty of Social Science
Department of Gamma-Informatics
Heidelberglaan 1
P.O Box 80140
Utrecht
The Netherlands
030-534911/4880

Abstract.
Managers of profit and non-profit organisations are often confronted with complex problems. Ill-structured diffuse problems which involve more than one domain. Problems that are hard to set let alone to solve. They differ a lot from the structured domain related problems we used to work with in school. In education little attention is given to the training of complex domain exceeded problems.
In order to be able to set domain exceeded problems, managers should get the opportunity to experience this kind of problems setting in a special learning environment. In order to enhance transfer the learning environment should be as close as possible to the real-life situation.
A conference room can be a good learning environment in which managers can be trained to set complex problems. Setting complex problems is teamwork. It demands knowledge of various domains. Therefore different experts should work together in making a conceptual model of the problem. This can cause serious communication problems.
A free form game with a case as a prototype of a complex problem can be a good didactical instrument for training problem setting.
Problemsetting is defining the scope of the problem, the domains and the level of aggregation. What can be the role of computer programs like decision support systems, simulations, expert systems, and general problemsolvers like SOAR and ACT* in the setting of domain exceeded problems?
In problem setting the computer can play a part by information retrieval. An expert-system as front-end of a database can assist the experts to get the relevant data out of the database in order to form in cooperation with each other a conceptual model of the problem.

Manager Training Environment For Setting Complex Problems.

D.J. de Tombe

1. Introduction.
This discussion is part of a research of the use of databases in setting complex problems.
This discussion focuses on the question of transfer of teaching problem setting and problemsolving. In order to enhance transfer the training situation should be as close to the real problemsetting situation as can be. Problemsetting with cases can be used to simulate real-life problemsetting. A free-form-game can function as a semi-natural learning environment. In using cases the managers can experience all aspects of real life problemsetting like complexity, context boundness of the knowledge and data-retrieval in order to make a conceptual model of the problem.

2. Societal problems of every day life are often complex, not well-defined and involve more than one domain. They differ a lot from the well-defined problems we have solved in school.
In school children are being educated to become fruitful members of society. In order to be a fruitful member of society they should be able to make a living, be a reasonable socialized person and be able to a deal with things like bureaucracy. These are broad often implicit learning goals. The school as the major institute of education operationalizes these broad goals into smaller learning goals, cutting the goals into different disciplines, disciplines into domains, domains into subjects and subjects into paragraphs.
In teaching subject after subject one can focus on the special problems concerning that subject. Like to be able to do some reading on level A. Most educational systems focuses on strictly divided subjects, like history, economy and physic[1].
In training small subtasks of different domains we hope that there will be transfer from the learning environment in school to the every day life demands. But we are so used to deal with the different domains in education that we forget to put the artificial separated domains together again.

3. Transfer.

[1] Although there ar some educational systems which try to overcome the domain gap in focusing on domian exceeded educational projects like pollution instead of separated subjects.

Research on transfer[2] points out that transfer seldom occurs.
In teaching Latin and mathematics learning theorists hoped that beside domain knowledge the students would learn logic reasoning. But the transfer to other domains is never proved. Likewise one hoped to find some transfer on the subject of reasoning and planning capacity by letting children work with Logo. But the transfer is not proved here either.
The research of Palinscar and Brown with their reading comprehension program 'Reciprocal Teaching' proved that transfer can be obtained when general principles of reasoning are taught together with self-monitoring practices and potential applications in varied context (Palinscar,1986; Baker & Brown,1984), teaching content knowledge, domain related heuristics and metacognitive skills. There is transfer in the reading comprehension program, but reading is a general skill, that must have caused at least some of the transfer.

Perkins (1989) names five points which should be met in order to get some transfer.
Transfer to new problems does take place
-when learners are shown how problems resemble each other
-when learners attention is directed to the underlying goal structure of comparable problems.
-when learners are familiar with the problem domains
-when examples are accompanied with rules, moreover when the students have formulated the rules themselves
-most important seems to be that learning takes place in a social context.
It is very much a matter of how knowledge and skills are acquired. In the real world transfer occurs only under specific conditions which are not often met in everyday life.
When they are met, transfer from one context to another often occurs (Perkins,1989).
Anderson states that transfer will occur when the underlying production rules are the same (Anderson,1989).

In order to enhance transfer the learning situation must be as close to the natural environment as possible.
The way education teaches students in a school environment is very artificial. Teaching in school differs from the way practitioners handle their problems in every day life (Brown, Collin, Duguin,1989). Schönfield attempts to overcome the artificial learning environment of school by letting the students enter the authentic world of a practitioner. In this

[2]Transfer is that knowledge, know-how and skills learned in one situation can be generalized and used in a different situation. If there can't be transfer the consequence is that everything must be learned in its own situation.

experiment the authentic world of the mathematic practice is rebuild (Schönfield,1985).
In school the focus is on domain related knowledge and skills, there is too little attention for domain exceeded learning. Teaching separated domain knowledge is only a vehicle to achieve a goal, not a goal in itself.
To be able to apply knowledge and skills, trained in separate domains, to real life complex situations, children should be trained from early age on to apply the knowledge learned in the separated domains into domain exceeded cases. To learn problemsetting with cases, in using cases as prototypes for real-life complex problems.
By using a case in training situations aspects of problem setting like complexity, diffuseness, context boundness and information retrieval can be experienced.

4. Does education succeed in problem solving.
Bruner states that education did not succeed in teaching problem solving even to the most intelligent children (Bruner,1971).
Also recent research of Dutch education points out that little attention is given to problem solving (Span & Kok).
When there is some training in problemsolving it is more a matter of applying rules to well-defined structured problems, than an exploration or creative activity in solving ill-structured problems.
"The educational system has created an environment in which students are scared to explore creative hypotheses because of their fear of failure. This cultivates a belief in a single 'correct' solution to a problem", Roger Schank (1989).
Psychologists and learning theorists have made many attempts to improve the teaching of problemsolving.
Bruner (1971) states that giving more attention to discovery learning will enhance the problemsolving capacity of children. Seymour Papert wants to enhance the problemsolving capacity of children by giving them a stimulating and inviting exploration world, the microworld of Logo, a selforganised learning environment (Papert,1980).
Roger Schank would like that the children could use the computer as a friend who one can ask questions (Schank,1989).
Education should give more attention to asking questions in stead of giving answers (Verhoeff,1987; Schank,1989).

5. How should problem solving be teached?
The last three decades psychologists and learning theorists have been discussing the question of problemsolving techniques.
The question is, should one teach domain knowledge or general heuristics with regard to problem solving within a domain.
In the fifty's and sixty's the answer to the question of teaching domain knowledge versus general heuristics was that teaching general problemsolving techniques is the best way to spend the teaching time.

Proof came from the field of chess playing. Specific domain knowledge is needed but is less important. Just some basic rules will do (Perkins,1989).
Artificial Intelligence research occupies with problem solving. Inspired by the results of problem solving research Newell & Simon build a program called 'The General Problem Solver' (Newell & Simon,1972).
Looking more closely to the nature of the chess game, it was noted that the chess masters did not only use general heuristics but also a lot of domain specific knowledge (schemata) (Chase and Simon,1973). Knowing this, Artificial Intelligence research switched from building general heuristics to building expert systems.
An expert system focuses on very specific domain knowledge and on very specific problems in the domain.
But what to do with atypical problems in the domain?
Research pointed out that experts confronted with atypical problems in their field did not only use domain knowledge but had to switch to general heuristics closely related to the domain to be able to solve the problem.
So it looks as if domain knowledge and general heuristics related to the domain would be a good combination.
Some proof for this statement comes from the field of reading comprehension from Palinscar and Brown with their reading method called 'Reciprocal Teaching' (Palinscar,1986; Baker & Brown,1984). The reason of the success of the program is, that they not only teach reading comprehension but also metacognitive skills close to the domain. These metacognitive skills can be regarded as general domain related heuristics.
Beside developing expert systems Artificial Intelligence researchers are looking for a general heuristic, for a unified theory of cognition.
Two programs which pretend to be general heuristic problemsolving instruments are ACT*[3] of John Anderson (Anderson, 1983) and SOAR [4] of Allen Newell. ACT* is a computational general psychological theory of skill learning.
SOAR is a developing system capable of general intelligence. SOAR can handle a lot of the small AI problems. The program SOAR is based on the problemspace hypotheses of Newell and Simon (Newell& Simon,1972).
These two computational theories claim both to be an unified theory of cognition.

---

3 ACT* stands for adaptive control of thought.

4 SOAR means State Operate And Result.

Whether SOAR and ACT* really are unified theories of cognition is too early to tell. An unified theory means integrating and explaining all the different small-scale cognitive theories. They have not reached that level yet.
SOAR and ACT* are general heuristics, but until now they only operate on very small and specific problems (Boden,1988).

6. Teaching environment related to knowledge levels.
By setting and solving problems one should be aware that one can distinguish several levels of knowledge. Each level of knowledge needs a different learning environment and a different guiding (Klabbers,1989).
The first level is maintenance knowledge, context free learning of rules and facts. This is presented as universal time-invariable knowledge. Here the teacher is the expert in a reproducing learning environment. In schools much attention is given to the learning of facts and rules within a special domain. This is what one calls maintenance learning (Botkin, Elmandjra, Malitza, 1979). Most of the time the learning of facts and rules are just handed over to the pupils.
The second level is context dependent knowledge, so-called innovation learning. This needs a heuristic guided learning environment in which the teacher is the guide. An aspect of this level is that one should be aware that the knowledge of the first level and second level exists and that there can be blind spots in the knowledge. This requires metacognitive skills in a self-steering environment. An environment where people can be active in learning autoregulation and autocontrolled skills. In this environment the teacher is the facilitater. A free form game can function as an environment in which autoregulation and autocontrolled skills can be trained.

The problems where Artificial Intelligence and education focus on are mostly domain related problems. Problems of which the answer is known. Little attention is given to the context boundness of the knowledge, to innovation learning or to the idea of living in changing situations in a changing world (Botkin, 1989).

7. Problem setting and problem solving.
When is a problem to be solved?
A problem can only be solved when it is recognized as a problem. Who's problem is it and who are the ones who are going to solve it. Then one can ask is it possible to solve this problem and when is this problem solved?
Problems solved for one group can be the start for problems for others. A solved problem often is the beginning of a new problem.

For many social problems there are no solutions in the way of a final and an objective answer (Rittel & Webber, 1973). Problems are at utmost to be resolved over and over again.
Before one can start solving a problem the problem has to be set. Problem setting is defining the problemspace. The problemspace is the space in which the solution of the problem can be found. Defining the problemspace is defining the scope, the aggregation level and the domain(s) of the problem.

Some problems are very hard to define, they are called 'wicked' problems. For a wicked problem it is uncertain whether or when it is solved. When a wicked problem is defined the problem is 'tamed' (Rittel & Webber, 1973).

An other aspect is complexity. A common general heuristic in solving a complex problem is dividing a problem into subproblems (Newell & Simon,1972). But how does dividing a problem into subproblems relate to complexity? One cannot cut out complexity by dividing the problem into pieces and putting them together later on. Complexity is a part of the problem.

Solving complex domain exceeded problems like implementation of the computer in education or the reorganization of healthcare in a country is team work. Knowledge and experience on how to handle this kind of problems is often missing. It is not always clear which domains, which fields and which people are involved.
Setting domain exceeded problems is not a one persons job, it demands knowledge of various domains. Therefore several specialists of various expertise must work together to set the problem.
Selecting the expertise team is the first step in setting the problem. Selecting experts is directly related to the defining of the domains. Depending on which domains are involved experts are invited to join the problem setting team. Selecting people and defining the domains can be a circular process. While getting more clearly which domains are involved some people may leave or join the group. Selecting people also depends on which point of view one choices to set the problem. By selecting certain people for setting the problem one encloses and excludes already certain solutions.
After defining the domain one can define the level of aggregation. Is the problem on the micro-level, on the meso-level or on the macro-level. Then defining the countries and organizations and people who are involved.
The last step in problem setting is defining the time-scope of the problem. Is it a quick to be solved problem or a problem that can take some longer time. Is it a problem of the past that still is playing parts or is it a problem of the future.

Now one can start making a conceptual model of the problem.
Making a conceptual model can be a first step in solving a problem. Most people have already some kind of mental model of the problem. But in order to be able to start handling the problem the whole group of problem solvers should have more or less the same conceptual model of the problem. There doesn't have to be a consensus, but at least some kind of agreement of what the conceptual model of the problem is.
When the problem is defined one can see whether the whole problem or a part of the problem can be modelled in some kind of computer aided decision support system. Like making a causal model of the relations and dataflow of the entities, variables and parameters. It can be a static or a dynamic model. A dynamic model can be represented in a system dynamic model like a simulation model on the computer. In making a simulation model of the problem one can try to form a picture of the consequences of the interventions one will do.

8. Databases.

For making a conceptual model of the problem one needs relevant data. Each expert involved with setting the problem should be able to get the right information about the entities concerning her domain.
How to get the right data needed for making a conceptual model of the problem?
If relevant data concerning the major aspects of the problem is available the expert can consult a database.
When there is no updated database available and the problem to be solved is not so urgent then there is time to collect the data concerning the different aspects of the problem.
After collecting the data, the data can be put into a database. The database can be build in the traditional way.
But for new urgent problems in fast changing situations relevant data are often missing. This is often the case with complex domain exceeded societal problems.
For a fast collecting of data one can't take enough time to collect data the traditional way and let a database be build by database-experts. In this case the content expert himself should be able to make a database. For each domain a special content expert should fill a database with relevant data. The content expert is able to collect relevant data about the latest developments concerning this problem, but is not a database expert therefore she needs a guide to help her filling the database. An expert system as front-end of a database can help the expert filling the databases in the right way. The expert system can also help selecting the right data for the database and implement the data into the database. As it is nor clear on forehand what kind of data is needed, text or numbers or pictures one needs a database

software tool that can handle all kind of data. That could be something like Hypercard. Hypercard is an easy to handle database software tool for multi-data storing which works on the Apple Macintosh. On the 'cards' of Hypercard one can store all kind of data like figures, texts, numbers, graphics, paintings etc.. It is a rather slow but very easy to handle databasetool.
After the content expert has collected the data, the data should be send to the problemsetting team.

9. How to extract the right information from a database in order to make a conceptual model of the problem.
For making a conceptual model of the problem each expert of the problem setting team needs relevant data of her/his own field. A possibility to help the content expert to get the right information out of a database is to build an expert system as front-end on top of a database which helps the expert to retrieve the information that is needed. An expert system as front-end can be put on a traditional database and on an easy the handle software tool as described above.

10. Real societal problems are often complex and domain exceeded. Like the problems now in East-Europe now or the changes concerning 1992 in Europe.
The social changes are going so fast that one hardly has the time to consider the problems in all its aspects. New societal structures replace old ones Old and new structures are entangled in new and still unknown situations, situations that require control. To many new problems old solutions don't work.

In education there is not much opportunity to get acquaintance with setting and solving complex problems.
To be able to handle these kind of problems managers should get some opportunity to train setting complex problems in an educational setting. In order to enhance transfer this educational setting must be as close to the real situations as possible. One should look for a learning environment where the real situation can be simulated. A conference room as learning environment, where a free form game is played with a case as a complex problem all aspects of problem setting can be trained. Cases imbedded in a free form game can be a good semi-natural learning environment in which context boundness, different knowledge levels and information retrieval can be trained.
Setting a case like healthcare or implementation computers into education can simulate the problemsetting of a real societal problem. In these cases one must in cooperation with other people define the domains, the aggregation level, the involved organizations, the time scope and train data retrieval in trying to make a conceptual model of the problem.

Setting a case can be imbedded in a free form game.
A free form game is a game with as little rules as possible, where in a non threatening situation people can learn to practice problem setting of complex domain exceeded problems. A free form game gives the participants the opportunity to experience the context boundness of each others knowledge. The context boundness is the personal knowledge of each participant, the knowledge that is coloured by experience, culture, position and discipline by which she or he considers the problem. Beside this the participators may have divergent interests and different power.
This context boundness can cause serious communication problems.
In playing a free form game one can learn to deal with hidden agenda's, divergent interests, experience the blind spots in the knowledge, experience changing levels from a outsiders view point to the insider point of view and the complexity of the problem.
In setting complex problems one meets not only the boundaries of ones own knowledge, but also the boundaries of the knowledge in the field.
In order to get full profit of the training the actors should be able to handle in accordance to their own capacities and to their own interest.
In a free form game with a special case as a problem setting item the problemspace will not be narrowed by a teacher to the space in which one must search for a solution. The actors can try to define the problemspace themselves.
In this kind of free form game the game-operator has a role as a facilitater.
The debriefing at the end of the game can be used to enhance learning
The actors should be made conscious of their own behaviour during the play. Metacognitive activities as autoregulation and autocontrol should be enlistened to enhance transfer.
In a free form game the participants have the chance to deal 'real life cases' with missing data using rules of thumb under time pressure like in real-life, without making too much accidents .
The conference room with cases as prototypes for complex problems can function as a learning environment in which managers can carrying on policy exercises. In a conference room, which can look like a boarding room the managers can be trained in a semi-natural learning environment.

Playing a free form game with a case as a prototype for problem setting in a conference room one can simulate the natural problem setting situation in semi natural learning environment.

In this way we hope to find some transfer of the trained knowledge and skills for setting complex domain exceeded problems to the setting complex real-life problems.

Literature

Anderson, J.A., 1989. *Psychology and Intelligent Tutoring in Artificial Intelligence and Education*. in Bierman, D., J. Breuker, J. Sandberg eds., Amsterdam: IOS.

Anderson, J.A., 1983. *A Spreading Activation Theory of Memory*, Journal of Verbal Learning and Verbal Behaviour, nr.22, 261-295. Academic Press, Inc..

Anderson, J.A, 1982. *Acquisition of cognitive Skill*, Psychological Review, nr.89, pp. 369-406, The American Psychological Association.

Anderson, J.A, ,1983. *The Architecture of Cognition*. Cambridge: MA: Harvard University Press.

Baker, L. & Brown, A.L.,1984. *Metacognitive Skills and Reading*, in R. Pearson, Handbook or Reading Research. New York: Longman.

Bruner, 1971. *The Relevance of Education*.

Batenson, G., Mind and Nature: 1979. *A Necessary Unity*. New York: Dutton.

Boden, M.A, 1988. *Computer models of mind, Computational approaches in theoretical psychology*. Cambridge: Cambridge University Press.

Botkin, J.W., Elmandjra, M., Malitza, M., 1979. *No limits to Learning: Bridging the Human Gap*. Oxford: Pergamon.

Brown, J.S., Collins, A., Duguid,P.,1989. *Situated Cognition and the Culture of Learning*, in Educational Researcher vol.18, nr 1,32-42. Washington.

Chase, W.C., & Simon, H.A., 1973. *Perception in Chess*, Cognitive Psychology,4, pp 55-81,.

Collins, A. & Smith, E.E., *Readings in Cognitive Science*. A perspective from psychology and Artificial Intelligence. San Mateo, California: Morgan Kaufmann Publishers, inc..

Doublait,S. & Lelouche, R. 1989. *Automatic acquisition of problem-solving tactics using imperfect knowledge*, lecture presented at the Earli Conference in Madrid, Spain.

Evans, R., January 1990. *Expert Systems and Hypercard*. in BYTE, (317-324).

Huizinga, J., ,1974. *Homo Ludens*. Groningen: Tjeenk Willink.

Klabbers, J., 1980. *De toekomst als spel*. Leiden: Universitaire pers.

Klabbers, J., Improvement of (self)steering through support systems

Klabbers, J., 1988. *Spelen op Onzekerheid*, in J. Klabbers (red), Kennen en Organiseren in Informatie: over Machines en Actoren. Deventer: Van Loghum Slaterus.

Klabbers, J., , 1989. *Methodologische aspecten van het ontwerpen van leeromgevingen* in : KLep, J. & Kommers (red.), Courseware en leerplan ontwikkeling , Symposium Didactische Systeemanalyse, Studies in leerplanontwikkeling 14, SLO-Enschede.

Klerk, de L.F.W., 1983. *Onderwijspsychologie*. Deventer: Van Loghum Slaterus.

Kok, W.A.M., 1989. *Characteristics of education for gifted students in the Netherlands*, paper reading on the Earli conference. Madrid: Spain.

Koning, J.A. 1989. *Koppeling van een expertsysteem aan een DBMS*,Uit: Kennissystemen, jrg1,nr 5.

Laird, J.E., Newell, A. Rosenbloom, P.S.1987. *SOAR: An Architecture for general Intelligence*, in Artificial Intelligence. Elsevier Science Publishers b.v. North-Holland.

Mars, N.J.I., 1988. *Onderzoek van niveau: kennistechnologie in wording*, in Informatie jaargang 30, nr.3, (81 t/m 176).

Meer,van der, Q. en Bergman, H., 1977. *Onderwijskundigen van de 20ste eeuw*. Groningen: Wolters Noordhoff.

Michon, J.A., 1989. *Allen Newell.* in De psycholoog, jaargang 24, nr. 6 juni.

Newell, A. and Simon, H.A., 1972. *The Theory of Human Problem Solving*.

Newell, A. and Simon, H.A., 1963. *G.P.S., A Program that Simulates Human Thought*, in E.A. Feigenbaum & J.Feldman (eds.), Computers and Thought, pp 279-293, R. Oldenbourg KG..

Rittel, H.W.J., & Webber, M.M., 1973. *Dilemmas in a General Theory of Planning*, in Policy Sciences 4 page 155-169, Elsevier Scientific Publisihing Company Amsterdam, Scotland.

Rumelhart, D.E., 1984. *Schemata and the Cognitive Sytem.* In R.S. Wyer & T.K. Srull (eds), Handbook of Social Cognition. Vol.1.Hillsdale, N.J., Erlbaum, pp 161-188.

Palincsar, A.S. 1986. *Reciprocal Teaching*, in ASCD Teaching Reading as Thinking, NAK Production Associates Washington D.C..

Papert, S., 1980. *Mindstorms*, Basic Books Inc..

Piaget, J., 1952. *The Origins of Intelligence in the Child.* London: Routledge & Kegan Publishers.

Perkins, D.N. & G. Salomon, 1989. *Are Cognitive Skills Context-Bound?* in Educational Researcher, vol.18, nr.1, pp 16-26, Washington, USA.

Romiszowski, A.J., 1985. *Developing auto-instructional materials*, Instructional Development, part 2. London: Kogan Page.

Schank, R., and D.J. Edelson, 1989. *Discovery Systems in Artificial Intelligence and Education*, Bierman, D., J. Breuker, J. Sandberg eds., Amsterdam.

Schön, D.A., 1983. *The reflected practioner*.

Schönfield, A.H., ,1985. *Mathematical problemsolving*. New York: Academic Press.

Span, P., Nelissen, J.M.C., et al (red), *Onderwijzen en leren*.

Turing, A.M., 1963. *Computing Machinery and Intelligence, Mind,* n.s.59, pp 433-460, 1950. Reprinted in E. A. Feigenbaum & J. Feldman(eds), Computers and Thought (11-35). New York: McGraw-Hill.

Takkenberg, C.A.Th.,1989.*The deliberation room*, in J.H.G. Klabbers, & J. Faber (eds.), International Meeting, Scope 2000. Utrecht: Department of Gamma-Informatics, University of Utrecht.

Waldrop, M.M., 1988. *Toward a Unified Theory of Cognition*, in Research News, July , Science, vol.241, pp 27-29.

Waldrop, M.M., *SOAR: A Unified Theory of Cognition.*

Wester,P., 1989. *Guru, kennistechniek in een databanksysteem*, in Kennissystemen, jrg1,nr 4.