

A unified process model for creativity-technique based problem solving processes

Florian Forster, Michele Brocco

Technische Universität München, Germany

Abstract. In this paper, we present a unified model for creativity-technique based processes that considers the key properties of each of these creativity techniques. For the construction of this model we first analyze processes of various creativity techniques with respect to their key properties. Afterwards, we use these findings to formalize a unified model and discuss its use for more flexible creativity support systems.

Introduction

Creativity techniques are used in many domains to guide creative problem solving processes [1]. Depending on the domains, context, problem type or people involved in the creative problem solving process, specific creativity techniques can be more or less adequate for finding appropriate solutions. Hence, their effective use is driven especially according to their strengths and weaknesses.

Collaborative tools for the support of creativity-technique based problem solving processes should address the main shortcomings of these processes which may also include typical problems for interacting groups such as the factors production blocking, group pressure and social loafing [2]. In addition certain creativity support systems (CSS) allow for the collaboration of distributed groups and virtual teams in creative problem solving processes.

As of today, CSS are tailored for a specific creativity technique (e.g. Mindmanager¹) or have a quite limited portfolio of supported techniques for idea generation (divergent phase) and idea evaluation (convergent phase) (e.g. Thinktank²). Hence, for a given context (i.e. problem type, group, situation, etc.) the available techniques may be not appropriate or be less effective than an alternative technique. Furthermore, using different techniques on the same problem can be beneficial for the process.

In order to support a broad set of creativity techniques, CSS need a unified model of creative processes that considers each of these techniques. As in any modeling process, the question of the appropriate level of abstraction has to be faced: A higher level of abstraction increases flexibility, but decreases the semantic information. E.g. a collaborative drawing tool on a virtual whiteboard may be very flexible for the use of creativity techniques. However, because of

¹ <http://www.mindjet.com>

² <http://www.groupsystems.com/technology/thinktank>

its high level of abstraction, the key properties of creativity-technique based processes are not comprehensively supported. This includes e.g. the possibility to anonymize group members or to set time limits for the divergent phase as required in specific techniques. Additionally the semantic of the single user contributions may be lost (e.g. no differentiation between ideas, evaluations or messages in the system).

In this paper, we present a unified model for creativity-technique based processes that considers the key properties of each of these creativity techniques. For the construction of this model we first analyze processes of various creativity techniques with respect to their key properties. Afterwards, we use these findings to formalize a unified model. This model can in turn be used to implement a corresponding software system that can support different creative-technique based problem solving processes.

Process analysis

The aim of the process analysis is to examine creative problem solving processes and to identify the key properties that can have a positive impact on the process outcomes. Obviously, a process model of a CSS should focus on these properties.

Creative processes are typically seen as a sequence of divergent and convergent process phases [3] [4]. In the divergent phase, the participants try to find ideas for a given problem. In the convergent phase, the participants evaluate the previously generated ideas [5]. Keeping the two phases strictly separated improves the effectiveness of the process [2]. Hence, a process model for CSS should comprise both types of process phases and avoid improper mixing of the phases.

The abstract perspective on the process is helpful to set the frame for a model for creative processes, but no advice is given on how the activities within the phases can be supported. Since creativity techniques claim to support creative problem solving processes, they can be regarded as a source for such parameters. Given their highly practical orientation, it is likely that they can give more concrete advice than an abstract process model. Based on this idea, we analyzed a multitude of common creativity techniques [6]. First, we wanted to find out if the processes the techniques imply actually fit in the cyclic model of divergent and convergent phases. Second, we were interested in the parameters the techniques impose on the different phases. Our analysis confirmed that the creative problem solving processes implied by the investigated creativity techniques could all be appropriately modeled as sequence of divergent and convergent process phases. Furthermore our analysis showed that there is a surprisingly low number of different parameters the various techniques impose on the processes.

In the meanwhile, we examined 14 additional creativity techniques³ and further refined the list of important process phase parameters in creativity-technique based processes. The following parameters were found for both phase types:

³ For descriptions of all mentioned creativity techniques in this article see [7] and [8].

- Anonymity: In divergent phases, shame or fear of rejection can inhibit the expression of unusual ideas. In convergent phases, group pressure can influence the voting behavior. It has been shown that by making anonymous contributions, this negative effects can be avoided [9].
- Time limit: Imposing a time limit on a process phase can be necessary due to organizational reasons, since this way, an upper limit for the duration of creative processes can be set. Time limits are also helpful for synchronizing creative processes in which the initial group is split up in smaller subgroups (e.g. *Brainwriting-6-3-5*).

The following parameters were found for the divergent phases only:

- Stimuli: Mental stimuli are pieces of information that are presented to participants in order to influence their thinking processes during the convergent phases. Stimuli are only rarely statically defined by the creativity technique itself. Instead, they are often contributed by the participants in previous phases (e.g. *random stimuli technique*).
- Start ideas: Start ideas are available from the beginning of a divergent phase and can directly be used to combine new ideas with. The ideas are typically generated in preceding divergent phases (e.g. *morphological analysis*).
- Constraints of idea representations: While for most techniques a verbal or textual representation of the generated ideas is assumed, a few techniques restrict the way the participants may express their ideas. The *greeting card technique* specifies the use of pictures that should be combined to compose ideas. During *brainsketching*, participants can only use sketches to explain their ideas.
- Limitation of idea quantity: Looking futile at the first glance, bounding the idea quantity may be necessary for practical reasons (e.g. *Brainwriting-6-3-5*).

The following parameters were found for the convergent phases only:

- Scenarios: A scenario is the description of a plausible future. A scenario defines a hypothetical context for the idea evaluation. The scenarios may be defined by the technique itself (e.g. *four future technique*) or may be generated by the participants in a previous process.
- Criteria: With criteria, the evaluation of ideas can be restricted to a certain aspect. Techniques mostly define static criteria (e.g. the *castle technique* defines effectivity, practicality and originality), but the criteria could also be dynamically generated by the participants in a previous process.
- Scoring: Scoring refers to assigning numerical values to ideas. Many techniques for convergent process phases imply using scores of a given range as evaluation measure (e.g. *sticking dots technique*). Idea evaluations from numerous participants expressed as scores can be easily processed and merged (e.g. by computing average values).
- Comments: By using comments, participants can evaluate ideas with free texts. With comments, participants can advance their opinion in a much more detailed way than with scoring. However, it is harder to aggregate these pieces of information than in the case of scoring.

Formalization of a unified process model

The parameters presented in the previous section can be regarded as requirements for a unified process model since we empirically found out that the processes implied by the techniques can be seen as a sequence of divergent and convergent phases based on these parameters. For being used in the context of a computer system, the process model has to be formalized. For the process phases, we suggest the formalization depicted in figure 1: **ProcessPhase** acts as

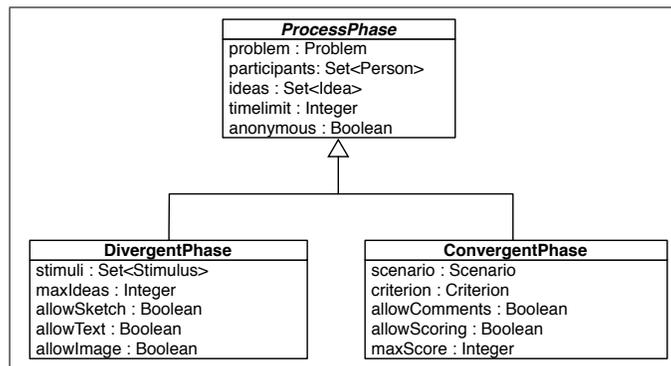


Fig. 1. Formal model of the phases of creativity-technique based processes.

the base class for the two different types of process phases. **ProcessPhase** must declare the following attributes:

- **problem**: The problem that should be solved in the process phase. Problems are commonly represented as plain text strings, but as concrete implementations of CSS (e.g. for special domains) may have additional requirements we suggest to define a dedicated class **Problem**.
- **participants**: The persons that are participating in the process phase.
- **ideas**: The ideas of the process phase. For divergent phases this attribute can be initialized with start ideas if needed, but in most cases the set is empty at the beginning of a divergent phase. In convergent phases, the set is initialized with the ideas from the precedent divergent phase so the ideas can be evaluated by the users. The results of the process analysis makes some implications about the functionality of the **Idea** class. For the divergent phases, it should support the expression of ideas as text, with images and with sketches. For the convergent phases, it should be able to store user ratings as numerical values and as text. Further adoption to a target domain of a CSS can be necessary.

The remaining attributes of **ProcessPhase** are formalizations of the creativity technique parameters found for both types of process phases:

- **timelimit**: Timelimits in a process phase are represented with an integer value that stores the remaining seconds for the phase or is undefined if no timelimit is set.
- **anonymous**: The need of anonymity can be represented by a boolean flag. If the flag is set, the CSS must keep all person-related information hidden during the process phase.

The divergent phases are represented by the class **DivergentPhase**, having the following attributes in addition to the ones from the base class:

- **stimuli**: A set of mental stimuli which must be presented to the participants by the CSS. If the value is not set, no stimuli are used in this divergent phase. Since stimuli are often generated by the group in previous phases and thus are formulated as ideas, the **Stimuli** class should be modeled similar to the **Idea** class or the same classes should be used.
- **maxIdeas**: Integer value that limits the number of ideas that may be generated in the process phase. If the value is not set, the number of ideas is not limited.
- **allowSketch**: If the boolean value is set, the users may sketch to express their ideas.
- **allowText**: If the boolean value is set, the users may use text to express their ideas.
- **allowImage**: If the boolean value is set, the users may use images to express their ideas.

The convergent phases are represented by the class **ConvergentPhase**, having the following attributes in addition to the ones from the base class:

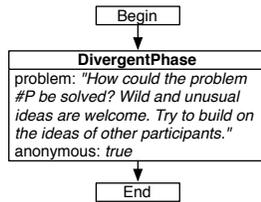
- **scenario**: The scenario that should be considered when evaluating the ideas in the process phase. In most cases it will be sufficient to represent scenarios as a string describing the particular scenario, but a dedicated class is preferable. If no scenario should be considered, the value is not set.
- **criterion**: The criterion that should be evaluated in the process phase. For the representation of criteria, the same considerations apply as for scenarios. If no criterion is set, the idea is to be evaluated as a whole.
- **allowComments**: If the boolean value is set, the users may evaluate the ideas using textual comments.
- **allowScoring**: If the boolean value is set, the users may assign scores to the ideas.
- **maxScore**: If scoring is allowed, this value defines the scoring range (from 0...maxScore).

The simplicity of the presented formalization for process phases of creativity-technique based processes makes it easy to be used in CSS implementations, yet it is powerful enough to represent creative problem solving processes of numerous creativity techniques and even combinations, since it contains all the key properties found in our process analysis.

Example processes

To illustrate the use of the unified process model, we present formalizations of creative processes that correspond to some well-known creativity techniques. As explained beforehand, a creative process is understood as a sequence of divergent and / or convergent phases, each having a particular set of attribute values. The complete set of possible attributes was presented in the previous section. In this example section, for clarity we will omit attributes that are not important for the given creativity technique. For the not-listed attributes, default values (e.g. "no restriction" for restricting attributes or "arbitrary number of participants" for the participant attribute) can be assumed.

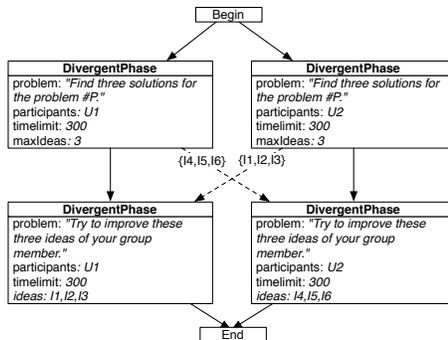
Brainstorming (and variants)



The major principle of the brainstorming technique is to avoid any idea evaluation during the idea generation phase. Since idea evaluation is not possible in divergent phases of the proposed model, this principle is achieved by modeling brainstorming as a divergent phase. Additional information to the given problem (here labeled #P) in the **problem** attribute makes the participants aware of the remaining brainstorming principles (wild ideas, building up on ideas of others). The classic brainstorming needs no further attribute values in the divergent phase, since no further restrictions are made by

the technique. Alternative brainstorming variants can be achieved by setting attributes of the phase, e.g. Anonymous Brainstorming by setting the **anonymous** attribute (as depicted in the figure on the left) or Brainsketching by setting **allowSketch** to true and **allowText** as well as **allowImage** to false, so only sketches can be used for expressing ideas.

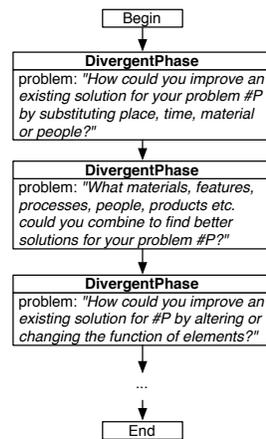
Brainwriting 6-3-5



A creative process with two participants U1, U2 based on the Brainwriting 6-3-5 technique can be modeled by **DivergentPhase** objects as shown in the figure to the left. In the first round (upper two phases), the participants are asked to find three solutions for the given problem. The participants have to work separately on their solutions, so U1 and U2 are in separate phases (**participants** attribute). The technique imposes a timelimit of 5 minutes, which is modeled with a value of 300

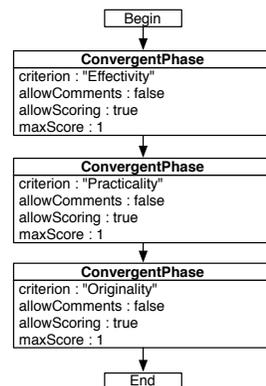
for the `timelimit` attribute, and sets an upper limit of 3 ideas with the `maxIdeas` attribute. When the `timelimit` has exceeded, the ideas generated by the participants are exchanged and placed in the `ideas` attributes of the successive phases. The participants are now asked to improve the received ideas instead of generating completely new ones.

SCAMPER



The SCAMPER technique can be modeled as a sequence of divergent phases, where in each phase a slightly different approach towards a solution is suggested using the `problem` attribute of the phase object. The figure on the left shows the first 3 phases of the SCAMPER technique (Substitute, Combine, Adapt), the remaining phases (Modify, Put to another use, Eliminate and Reverse) are modeled similarly. This way, all type of checklist-based technique (e.g. Osborn-checklist, CATWOE) can be represented. Since the techniques impose no further restrictions, the remaining phase attributes are not set. By setting some of the attributes, combinations of techniques could be achieved: e.g. setting the anonymous attribute in the phases leads to an Anonymous SCAMPER process.

Castle Technique



The Castle Technique is an evaluation technique that suggests to judge ideas in sense of their effectivity, practicality and originality. To speed up the decision process, the participants may only say if the criterion is met or not. Formalized by the unified process model, a castle technique process is a sequence of convergent phases (`ConvergentPhase` objects). In each phase, the participants have to decide if the given ideas (in general coming from a previous divergent phase) fulfill the criterion defined by the attribute `criterion`. To express their decision, participants may only use score values (`allowScoring` is true while `allowComment` is false) and can only make a binary decision, since the `maxScore` attribute is set to 1.

Conclusion

In this article we proposed a formalization for creativity-technique based problem solving processes as sequence of process phases. We first described how creativity techniques are currently supported in creative support systems. Then, we argued why a unified process model is a way towards more flexible CSS. In the second section we summarized the results of our analysis of a large set of creativity-techniques with respect to their key properties. Finally, we presented a formal unified model comprising the key properties of creativity-technique based problem solving processes and illustrated the approach with some example process formalizations.

Since creativity techniques guide creative processes by affecting the parameters we identified in the process analysis, they can be interpreted as presets of attribute values in the process model (e.g. a specific configuration of the attributes `timelimit`, `stimuli` etc.). Following this concept, a formalization of creativity techniques (in contrast to concrete creative processes) can be achieved.

Furthermore, the key properties of creativity techniques can be regarded as a framework for analyzing creativity techniques themselves: by varying single attribute values of the specific process phases it is possible to investigate the effects on the produced outputs. For instance by varying the `timelimit` attribute value in different creative processes, a better general understanding on the effect of time limits in creative problem solving processes can be gained.

As a framework for evaluation, we are currently implementing a CSS prototype which is based on the here discussed unified process model. Thereby, our goal is to assess the completeness of the proposed key components.

References

1. Fernald, L.W., Nickolenko, P.: The creative process: Its use and extent of formalization by corporations. *Journal of Creative Behaviour* **27**(3) (1993) 214–220
2. Isaksen, S.G.: A review of brainstorming research: Six critical issues for inquiry. Technical report, Creative Problem Solving Group, Buffalo, New York (1998)
3. Parnes, S.J.: Source Book for Creative Problem Solving : A Fifty Year Digest of Proven Innovation Processes. Creative Education Foundation (1992)
4. Finke, R.A., Ward, T.B., Smith, S.M.: *Creative Cognition: Theory, Research and Application*. Bradford, The MIT Press (1992)
5. VanGundy, A.B.: *Idea Power*. AMACOM (1992)
6. Forster, F., Brocco, M.: Understanding creativity-technique based problem solving processes. In Lovrek, I., Howlett, R.J., Jain, L.C., eds.: 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems. Volume 5178 of LNAI., Heidelberg, Springer (2008) 806–813
7. VanGundy, A.B.: *Techniques of Structured Problem Solving*. Van Nostrand Reinhold (1988)
8. Mycoted: Creativity & innovation, science & technology. <http://www.mycoted.com>
9. Connolly, T., Jessup, L.M., Valacich, J.S.: Effects of anonymity and evaluative tone on idea generation in computer-mediated groups. *Manage. Sci.* **36**(6) (1990) 689–703