An Ontology for Collaborative Decision Making

Jacqueline Konaté[1]([✉]) , Pascale Zaraté[2,3] , Aminata Gueye[1] ,
and Guy Camilleri[2,3]

[1] Faculté des Sciences et Techniques, Université des Sciences,
des Techniques et des Technologies de Bamako,
Colline de Badalabougou, Bamako, Mali
`jacqueline.konate@usttb.edu.ml`,
`aminata.gueye@fst-usttb-edu.ml`
[2] IRIT, 118 route de Narbonne, 31062 Toulouse Cedex 9, France
`{zarate,guy.camilleri}@irit.fr`
[3] Université de Toulouse; UPS, INSA, INP, ISAE; LAAS,
31077 Toulouse, France

Abstract. This article focuses on an ontology construction for collaborative
decision making. To do this, a state of the art on collaborative decision-making,
on ontology engineering and on collaboration engineering has been done. An
eight-step ontology development methodology was adopted and implemented to
build the ontology. A corpus made up of more than seventy-seven (77) docu-
ments was the starting point for the extraction of terms from the ontology and
the UML (Unified Modeling Language) language served as a description lan-
guage of our ontology. This ontology is intended to be the starting point for a
facilitation support system in a Collaborative Decision Making process. The aim
of the work is to produce a new system according the "Facilitator in the box"
paradigm.

Keywords: Ontology · Collaborative Decision Making · Artificial
Intelligence · Facilitation · Collaboration Engineering

1 Introduction

Among the collaborative practices in organizations, decision-making is one of the most
common and particularly important. This decision-making is done most often in a
collaborative way, hence the Collaborative Decision Making. Making collaborative
decision-making can have significant benefits such as efficiency, taking into account all
stakeholder proposals. However, a collaborative Decision Support System (GDSS) and
the attitudes that encourage interaction and sharing are necessary to achieve these
benefits. To do this, a number of approaches have been developed; most of them
emphasize the need to use an automated tool. However, the use of predefined processes
is essential because it allows gaining a substantial advantage of all the resources
(human, technology, time, …) mobilized for the decision making. An approach called
Collaboration Engineering provides tools for designing collaborative processes for

practitioners that can do without the intervention of a facilitator when using these processes [6].

Facilitation activities are central to decision-making. However, skilled facilitators are rare in organizations as they tend to become managers very quickly [6]. Our ambition is to embed facilitation skills into a system. Several works have already been made to support the activities of the participants in the decision-making process. However, efforts still need to be made to support the facilitation tasks including the parameterization of the tool. This paradigm is called "Facilitator in the box" and intends to encapsulate the skills of a facilitator in a system (GSS: Group Support System) that can self-configure according to a number of parameters [4]. In the area of decision-making, this system is called GDSS.

Designing a system that can withstand key facilitation tasks and self-configure requires a lot of conceptualization and prior definition of inference rules. Such a system is an artificial intelligence system and we considered that the development of ontology is relevant. This ontology has to be answered the necessary questions for an automatic parameterization on the basis of certain entries.

Ontology defines a common vocabulary for researchers who need to share information in a field [17].

Ontologies are used in several fields including Philosophy, Linguistics and Artificial Intelligence (AI). As the goal of AI is to make machines sophisticated enough to integrate the sense of information [16], the organization of knowledge is a very important step towards achieving this goal. This step gave birth to knowledge engineering that relies heavily on ontologies as a means of representation and organization of knowledge. The goal of our work is to conceptualize the field of collaborative decision-making to develop a collaborative decision-support tool. Thus, the definitions of ontology to which we will be concerned are related to the discipline of Artificial Intelligence.

The ontology definition language we used is UML (Unified Modeling Language) which is initially designed for the representation of the structure of a system. However, it has been shown that UML has many similarities with OWL (Web Ontology Language) and that the first one can be used in place of the second one. Cranefield is the precursor of this approach [9].

In this context, this article presents a work whose objectives are:

– the characterization of the different types of ontologies,
– the identification of the type of ontology that fits to our needs,
– the selection of a methodology for the construction of an ontology,
– the implementation of this methodology for an ontology development for collaborative decision-making.

The remain of the paper is organized as follows: Sect. 2 presents the state of the art in collaborative decision-making, ontological engineering and collaboration engineering; Sect. 3 presents the ontology implemented through the methodology and finally Sect. 4 presents the conclusions of this work and the associated perspectives.

2   Background

2.1   Collaborative Decision Making Process

In this section, we are going to highlight the main phases of a generic collaborative decision making process. Figure 1 presents a general view of it.

The **preparation phase** consists of defining the problem of decision-making according to the ontology. In other words, we must define the purpose, the domain, the current context of the problem as well as the criteria and possible constraints.
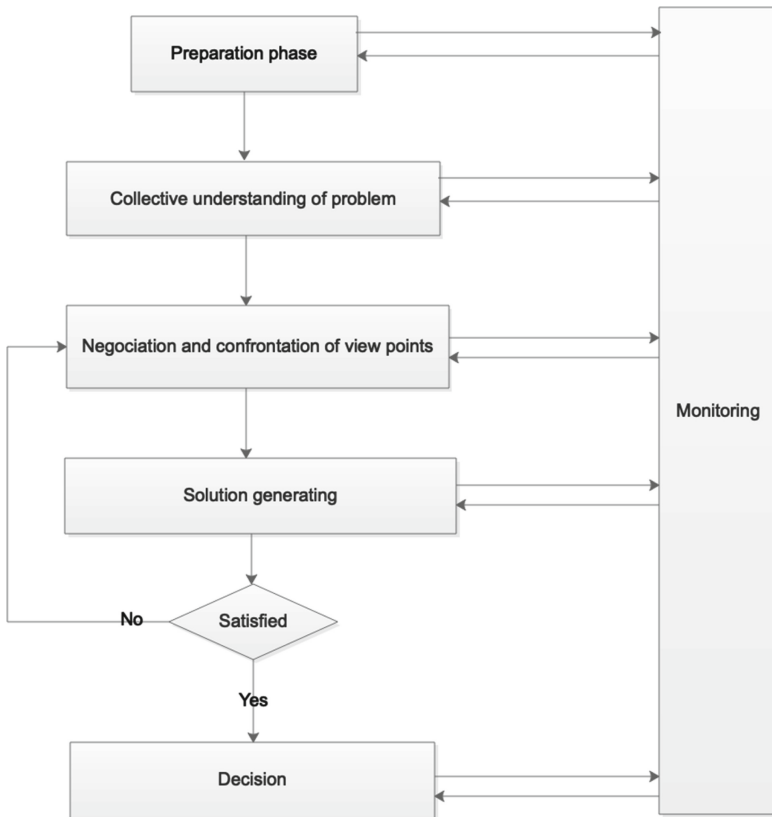

Fig. 1. Generic model of the collective decision-making process [1].

The **phase of collective understanding of problem** can be considered as an extension of the preparation phase. Indeed, it consists of sharing a common vision of the problem with all the participants and finding an agreement on how to implement the designed process [19].

The **solution generation phase** is the beginning of the treatment and it consists to produce alternative ideas to solve the problem.

The **phase of negotiation and confrontation of viewpoints** comes after that of generation to allow participants to elaborate their contributions by arguing them in order to win the support of the greatest number.

The **decision phase** occurs after the negotiation and confrontation phase of the solutions. It consists in selecting, according to the criteria previously defined, the ideas which have been approved by maximum participants, or which will have made the consensus within the group.

The **monitoring phase** covers all the decision making process so that any problem can be timely fixed. It also includes generating a report on all decision-making process and ensures their implementation. To do that, a document will be generated at the end of the decision-making process and will serve as a basis for further work.

As announced below, our purpose is to design a self-configurable tool for collaborative decision making which embedded facilitator skills. To do that, we need an ontology for main concepts of domains involved in this process. So, the next sections are devoted to ontologies and how to develop them.

2.2   Ontology Definition

The ontologies are useful for knowledge representation and are very important for Knowledge Engineering. The name «ontology» has been used for more than two decades and applies to a wide range of fields such as Artificial Intelligence. An ontology must be designed to respond to the concerns that prevailed in its development. For example, an ontology for computer maintenance is supposed to answer any question relating to the diagnosis of a computer failure based on the data provided by a user.

A general definition of ontology is "an explicit and formal specification of a shared conceptualization" [3]. The elements of this definition should be understood as such [16]:

– *conceptualization*: an abstraction of a phenomenon, obtained by identifying the concepts appropriate to this phenomenon;
– *formal*: it indicates that ontologies are interpretable by the machine;
– *explicit specification*: it means that the concepts of ontology and the constraints related to their use are defined declaratively;
– *shared*: refers to the fact that ontology captures consensual knowledge.

Through the elements above, we can deduce an ontological terminology, a common vocabulary, a domain name, the concepts and the relations between them.

In the next subsection, we present the different types of existing ontologies and their characteristics.

2.3   Types of Ontologies

In this section we do not intend to give an in-depth typology of ontologies. On the other hand, we want to give a general view on the main types of ontologies [13]:

- **High level ontology:** this type of ontology presents concepts regardless of context or situation. In doing so, since the concepts are out of context, they must be widely accepted in general by people. All or almost everything could be represented using this approach.
- **Domain Ontology:** as its name suggests, this type of ontology is conceived in a specific context. It is an organization of knowledge related to an area that could be agriculture, livestock, civil engineering.
- **Tasks Ontology:** this type of ontology is used in problem-solving context to represent concepts related to the specific tasks that are executed. These tasks may be those related to the design, those related to a surgical operation.
- **Application ontology:** an application ontology is quite particular because it associates two specific ontologies, one of which is a domain ontology and another one is a task ontology. In other words, it is a conceptualization of the tasks carried out by the actors of the field when they are at work.

Since our goal is only to design an ontology for a particular domain: the collaborative decision-making. Our ontology will be thus of the fourth type: an application ontology with a domain straddling decision-making and Collaboration Engineering.

In the next section, we will discuss the process of building an ontology.

2.4   Ontology Engineering

To conceive ontology, it is necessary to determine the reasons to build ontology and to adopt a methodology to construct this ontology. In the following, we present these two steps.

Reasons to develop ontology. According to Noy and McGuinness [17], the needs to develop an ontology are diverse and varied and we can mention:

- *The need for a common understanding among software designers*
  According to Gruber, this reason is one of the most common reasons to develop ontologies [12]. Indeed, although software is designed for a given domain, its use is transversal to several domains. For example, there are websites that manage online bookstores. These web sites can share the same ontology, then search engines can easily extract information from these different sites.
- *The need for reuse of knowledge on a domain*
  This need has been instrumental in advancing ontology research. Indeed, cross-domain notions tend to be defined differently across domains. When specialists agree to design general ontology for users that can be used as basis for specific ontologies.
- *Make explicit what is considered implicit on a domain*
  There are several notions that are perceived differently by the actors of a domain because they are not clearly defined. To help stakeholders better understand themselves or even to increase their knowledge of the field, it is important to explicit the notions.

- *The need to distinguish knowledge on a field and operational knowledge*
  This is the difference between a domain ontology and a task ontology of the same domain. The first one is a representation of the concepts, so the knowledge on the domain and the second one is a representation of the operational concepts of the domain.
- *Analyze knowledge on a domain*
  In essence, ontology gives details on concepts, especially the important concepts are explicitly and accurately defined. It is naturally a propitious basis for the formal analysis of the concepts of the field in which the ontology has been elaborated.
  It is important to see that domain ontology is not always a goal. However, the concepts that it contains and their structure are likely to be used by other programs [17].
  The following section presents the methodology for building an ontology.

A Methodology for Ontology Building. Works done by Hernandez and Mothe [14] and Lechchine [16] allow to distinguish eight (8) stages for ontology building:

- **Step 1: Ontology requirements specification:** this step consists in determining the domain of knowledge of the ontology, its type, the objectives which it aims and its future use. It also determines the scope and techniques that must be used for concept extraction.
- **Step 2: Choosing the corpus:** this is to select documents from which the concepts will be extracted.
- **Stage 3: Linguistic study of the corpus to extract the terms and their relations:** this step allow to identify the terms representative of the domain as well as the relations which bind them.
- **Step 4: Normalization of the results of step 3:** it consists in identifying and defining the concepts and the semantic relations between the concepts and the terms.
- **Step 5: Modeling the ontology:** it consists of using an ontology language to formally represent the semantic network deployed in step 4.
- **Step 6: Ontology structure building:** it is drawing up a hierarchy of ontology concepts without redundancies and ambiguity and possible new relations.
- **Step 7: Validation of Ontology by domain expert(s):** the work from the previous stage must be approved by the specialists in the field.
- **Step 8: Ontology updating:** Since any domain is subject to change/ evolution, it is important that the related ontologies are regularly updated either by adding new concepts or by reformulating others.

2.5 Collaboration Engineering

Collaboration Engineering is an approach to design collaborative work practices for high value recurrent tasks, and the deployment of these designs for practitioners to perform them themselves without the need for collaboration professional facilitator [6]. This approach is very interesting for us because its objective is to empower participants in a collaborative decision-making process. So, the concepts it promotes can be included in the design of the decision-making tool that we have to design.

Collaboration Engineering aims to provide some of the benefits of professional facilitation to groups that do not have access to professional facilitators.

Some Key Concepts. Here, we describe the concepts of Collaboration Engineering approach [4, 7, 10, 15].

- *Collaboration* is joint effort toward a common goal. It is necessary that the participants in the collaboration make the effort to achieve the goal of the group.
- *Goal* is a desired state or result.
- *Work practice* is a set of actions performed repetitively to accomplish a particular organizational task.
- *Collaborative task* is one that success depends on the joint efforts of several individuals.
- *High value task* is one that brings substantial advantage to an organization or one that avoids a substantial loss by its successful completion. The practice of the task produces a high value over time for organization.
- *Recurrent task* is one that can be conducted repeatedly through a similar process each time it is executed. In other words, it indicates that a similar approach can be used each time the task is executed even with some variations in its parameters.
- *Collaboration Process* is a series of activities performed by a team for a specific purpose within a given time frame.
- *Facilitator* is someone who both designs and conducts a dynamic process that involves managing relationships, tasks and technology, as well as structuring tasks and contributing to the achievement of the outcome of collaboration process. This work is called Facilitation.
- *Practitioner* is a specialist in a task and must perform important collaborative tasks such as risk assessment or decision making as part of his or her professional duties. He executes a specific collaborative process on a recurring basis, so he does not need any facilitation skill.
- *Collaboration Engineer* is the one who designs and documents collaboration processes that can be easily transferred to a practitioner. This means that the practitioner can run a process without the help of a collaboration engineer or facilitator.
- *ThinkLet* is the smallest unit of intellectual capital needed to create a collaboration pattern.
- *Reusability* is the property of a thinkLet to be used to solve problems different from those for which they were originally created.
- *Predictability* is the property of a thinkLet, when executed as prescribed, to create similar variations in overall collaboration patterns and deliverables with a variety of teams, tasks, and conditions.
- *Transferability* expresses the degree to which people who have never created a thinkLet can learn it, remember it, and execute it successfully.

About thinkLets. Design patterns were introduced by Alexander who is architect by a 1970s observed a recurrence of problems that occur in the architectural design phase. He imagined the concept of pattern as follows: "A pattern describes a problem that repeats and repeats itself again and therefore describes the core of the solution to this

problem, so that you can use this solution million times without never do it the same way twice" [2]. Existing patterns are about two hundred and fifty three (253) and cover all aspects of building construction.

Pattern concept was adopted for software design by Gamma [11] who helped to prove the value of the concept and made him a reference in computing field.

The concept of design pattern has also been taken up by collaboration engineering theorists to propose a new Design Pattern language for collaboration called thinkLets. They are the best practices of expert facilitators to support groups in their collaborative efforts to achieve goals. ThinkLet is defined as the smallest unit of intellectual capital needed to create a collaborative pattern [6]. They are reusable, predictable and transferable facilitation techniques that can be used to drive a group through a process towards its goal [10]. Each thinkLet is an instantiation of one following six general patterns, also with sub patterns [7]:

- "Generate" allows the move from having few to having more concepts,
- "Clarify" allows moving from having less to having more shared meaning of the concepts under consideration,
- "Reduce" is the opposite of the Generate pattern,
- "Organize" is moving from having less to having more understanding of the relationships among the concepts,
- "Evaluate" is to move from having less to having more understanding of the utility of the concepts priority with respect to goal attainment.,
- "Build Consensus" is to move from having more to having less disagreement on the course of action.

For more details on thinkLets, see the paper from Briggs and de Vreede [5].

In the next section, we present how to design a collaboration process based on the concepts we presented earlier.

3   Developing an Ontology for Collaborative Decision Making

Our goal is to design a GDSS with the possibility of setting it automatically. In this part, we will present the different steps that we followed to develop our ontology, and then we will present our ontology.

3.1   Our Ontology Building

In this part, we will identify the questions that our ontology must answer, then apply the methodology presented in Sect. 2.4 to build our ontology.

Questions Must be Answered by Ontology. After a process setting effort, five (5) main parameters were identified. These are: (1) goal of decision, (2) purpose and scope of decision making, (3) number of decision makers, (4) process duration, and (5) anonymity. Based on these parameters, the questions to which the ontology must respond have been developed. They are presented below in order of priority:

1. Is decision-making multi-criteria or single-criteria?
2. Which collaboration patterns of should be used for this process?
3. What kinds of decision makers (skills, qualities, characteristics, …) should partic-ipate in this process?
4. What is the appropriate number of decision makers?
5. What should be the duration of the process?
6. Is anonymity required for this process?

Above questions bring to understand that the reason to develop the ontology is «*The need to distinguish knowledge on a field and operational knowledge*» according to Sect. 2.4 presented above. We will proceed to apply the methodology to build an ontology that meets the needs specified above.

Methodology Implementing.
Step 1: Ontology Specification
Our ontology is between several **areas of knowledge** that are: Decision Making (DM), Decision Support Systems (GSS, GDSS) and Collaboration Engineering (CE).

The type of ontology is **Application Ontology:** two domain ontologies have been selected and are: the GDSS and CI domains and one tasks ontology which is the domain of Decision Making (DM)

The **objectives** for the ontology are essentially: identify the different parameters of a decision process and identify the appropriate collaboration patterns (thinkLets) for a given situation

The **scope of the ontology** covers the important concepts of Decision Making, Decision Support Tools (GDSS) and Collaboration Engineering (CE).

The **techniques** used to develop ontology are extracting concepts, relationships and axioms.

About the **use of the ontology**, it concerns the facilitation, the automatic param-eterization of the collaborative decision-making assistance tools (GDSS).
Step 2: Choosing the corpus:
We have selected more than 177 papers consisting of research papers and technical reports from various sources including conference proceedings and journals through Google Scholar, IEEE Xplorer, ACM, JSTOR, ScienceDirect, ResearchGate, GDN, and also direct contact with authors. We used keywords like: Collaboration, Decision Making, Collaboration Engineering, Groupwares, DSS and GDSS.
Step 3: Linguistic study of the corpus to extract representative terms of the domain and their relations (lexical and syntactical)
We used the extraction tool named Termostat to extract the significant terms of the different domains as well as the relationship between them.
Step 4: Normalize the results of step 3
After extracting the concepts from the previous step, we identified those that con-tributed to answering the six (6) questions identified in Subsect. 3.1.1. Then, we defined these concepts and related them each other.
Steps 5 & 6: Ontology Modeling and Building Ontology Structure
To formally represent ontology, we used UML (Unified Modeling Language) to rep-resent the concepts and relationships from the previous step and to build the hierarchy of concepts for the three domains. Indeed, there are several works in the literature that

argue UML can be used as an ontology description language because of their similarity [8, 9, 18]. In addition, we selected this solution because it allowed us to merge steps 5 and 6 of the methodology. In addition, UML would facilitate the implementation of the group decision making tool.

Ontology does not contain redundancies in relationships. Each concept has been defined explicitly to remove any ambiguity in its meaning. Similarly, the relationships between concepts are named to help understanding and the enrich ontology. For more details on this work, see Sect. 3.2.

Step 7: Validation of Ontology by domain expert(s)

Ontology was evaluated by specialists from domains of knowledge concerned with ontology: Decision Making, Collaboration Engineering, Group Decision Support Systems and Ontology Engineering. Each of the experts made comments and observations on the work done. Efforts have been made to improve work according to their contributions (see section 3.2).

Step 8: Update Ontology

For the evolution of our ontology, we plan to set up a community around the proposal. This community will animate the various activities aimed at adapting the ontology to the needs of each other.

3.2   An Ontology for Collaborative Decision Making

The implementation of the methodology produced a result whose schematic representation is given in Fig. 2.

1. **Decision Problem:** the subject for which a decision must be made. For example, there may be resistance from users to adopt a new system.
2. **Constraint:** A requirement that must be satisfied by the process or an element of the process. For example, the minimum or maximum number of participants in the decision-making process.
3. **Criterion:** an element allowing making a decision.
4. **Decision making process and subProcess:** this term refers to the successive states through which the group passes to arrive at the decision. It is a continuation of different phases of a decision-making.
5. **Report:** provides a detailed presentation, a restitution of all activities, and also contains the final result (the adopted decision).
6. **Alternative:** a choice, a possible solution, a probabilistic result provided by the decision makers. Choices having different consequences, the final choice will be made according to specified criteria.
7. **Decision_Result:** it is the final choice, the adopted alternatives, the (optimal) options selected according to the criteria (if there is any).
8. **Resource:** Technology, tools and other physical resources such as money or a meeting room can be used in this process.
9. **Phase and Reunion:** is a step in a process that may consist of one or more subphases that will be called Reunion. A reunion represents the various groups, the assemblies of people created to discuss about topics of decision-making.
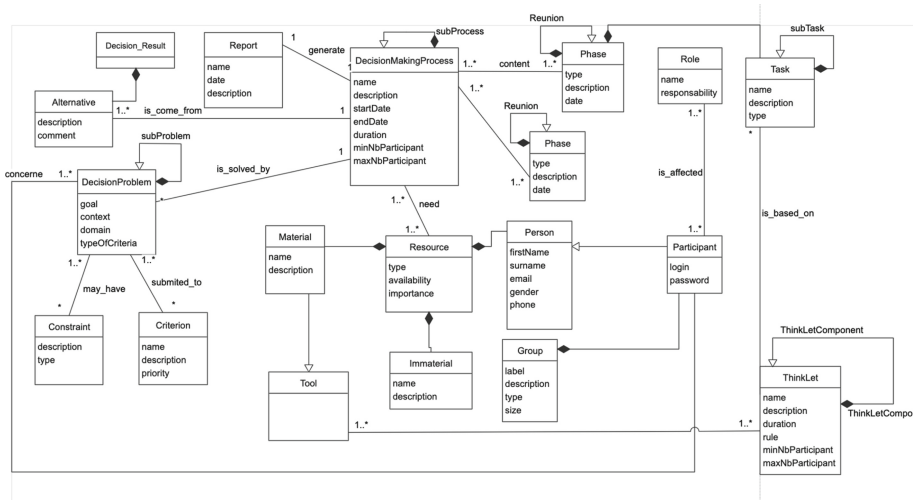
Fig. 2. An ontology for collaborative decision making

10. **Task and subTask:** is a job to be done by a participant. A task can have of subtasks.
11. **Role:** the function, the right of use that a user has, the different tasks that he must perform. One of the important roles is that of the facilitator who is a participant that drives the decision-making meetings.
12. **Participant** (or agent or actor or stakeholder or user): is any individual who intervenes and plays a role in decision-making.
13. **Group:** is a set of decision-makers with common factors, a collective goal that collaborates in decision-making.
14. **Person** (or human or human resource): characterized by first name, surname, gender, email, phone number.
15. **Material:** nonhuman material resource.
16. **Immaterial:** non-material and non-human resource.
17. **Tool:** means used to perform one or more parts of the decision-making process.
18. **ThinkLet, ThinkLet_Component, ThinkLet_Composite:** refers to the different collaboration patterns used in decision-making. The ThinkLet can be compound or elemental.

4   Conclusion

This article presents ontology for collaborative decision-making. It is an application ontology built from the following areas: collaborative decision-making, decision support systems, collaboration engineering. The development approach that we adopted has eight stages and the starting corpus consists of seventy-seven (77) journal articles, conferences papers and technical reports from Google Scholar, Elsevier, IEEE explorer,

ScienceDirect, etc. The tools used are Termostat for terms extraction from the corpus and UML for ontology hierarchy building and its standardization. The resulting ontology has eighteen (18) concepts that have been defined with their properties and thier relationships. The use of this language is justified by the purpose of our work: the development of a group decision support system.

For the validation of the ontology, specialists in decision-making, in Collaboration Engineering, in collaborative tools for decision-making, in ontology engineering.

Our goal is to build a collaborative decision support system that automatically configures itself based some entries. This would greatly assist the decision-making groups in their work by giving them some autonomy from the professional facilitators, but also by relieving them of the repetitive tasks of configuring the tools. This is the paradigm called "Facilitator in the box".

References

1. Adla K.: Aide à la Facilitation pour une prise de décision Collective: Proposition d'un Modèle et d'un Outil. Thèse de l'Université de Toulouse, soutenue le, 8 June 2010 (2010)
2. Alexander, C., Ishikawa, S., Silverstein, M.: A Pattern Language: Towns, Buildings, Construction. Oxford University Press, New York (1977)
3. Borst, W.: Construction of Engineering Ontologies. Unpublished doctoral dissertation, University of Tweenty (1997)
4. Briggs, R.O., Kolfschoten, G.L., De Vreede, G.J., Lukosch, S., Albrecht, C.C.: Facilitator-in-a-box: process support applications to help practitioners realize the potential of collaboration technology. J. Manag. Inf. Syst. **29**(4), 159–194 (2013)
5. Briggs, R., de Vreede, G.-J.: ThinkLets: Building Blocks for Concerted Collaboration. Briggs and de Vreede, Nebraska (2009)
6. Briggs, R.O., De Vreede, G.J., Nunamaker, J.F.: Collaboration engineering with thinklets to pursue sustained success with group support systems. J. Manag. Inf. Syst. **19**(4), 31–64 (2003)
7. Briggs, R.O., Kolfschoten, G.L., de Vreede, G.-J., Dean, D.L.: Defining key concepts for collaboration engineering. In: Rodríguez-Abitia, G.I.A.B. (ed.) AMCIS, vol. 17 (2006)
8. Brockmans, S., Volz, R., Eberhart, A., Löffler, P.: Visual Modeling of OWL DL ontologies using UML. In: International Semantic Web Conference, pp. 198–213 (2004)
9. Cranefield, S.: Networked knowledge representation and exchange using UML and RDF. J. Digit. Inf. **1** (2001)
10. De Vreede, G.J., Briggs, R.O.: Collaboration engineering: designing repeatable processes for high-value collaborative tasks. In: Proceedings of the Annual Hawaii International Conference on System Sciences (2005)
11. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Pattern. Addison-Wesley, Boston (1995)
12. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **6**, 199–221 (1993)
13. Guarino, N.: Formal ontology and information systems. In: Proceedings of Formal Ontology in Information System, pp. 3–15. IOS Press (1998)
14. Hernandez, N., Mothe, J.: TtoO: une méthodologie de construction d'ontologie de domaine à partir d'un thésaurus et d'un corpus de référence, Rapport interne, IRIT (2006)

15. Kolfschoten, G.L., Briggs, R.O., de Vreede, G.J., Jacobs, P.H.M., Appelman, J.H.: A conceptual foundation of the thinkLet concept for Collaboration Engineering. Int. J. Hum Comput Stud. **64**(7), 611–621 (2006)
16. Lekhchine, R.: Construction d'une ontologie pour le domaine de la sécurité: Application aux agents mobiles, Rapport de master, Université de Mentouri, Algerie (2009)
17. Noy, N.F., McGuinness, D.L.: Ontology development 101: a guide to creating your first ontology. stanford knowledge systems laboratory technical report KSL-01-05 and stanford medical informatics technical report SMI-2001-0880, March 2001 (2001)
18. Pătraşcu A.: Comparative analysis between OWL modelling and UML modelling. Econ. Insights Trends Chall. **IV(LXVII)**(2), 87–94 (2015)
19. de Terssac, G., Chabaud, C.: Référentiel opératif commun et fiabilité. In: Leplat, J., de Terssac, G., (eds.) Les facteurs humains de la fiabilité dans les systèmes complexes, pp. 111–139. Octarès, Toulouse (1990)