Journal of Universal Computer Science, vol. 17, no. 2 (2011), 261-275
submitted: 17/9/09, accepted: 7/12/10, appeared: 28/1/11 © *J.UCS*

LaSca: a Large Scale Group Decision Support System

Gustavo Carvalho
(Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
gustavoh@cos.ufrj.br)

Adriana S. Vivacqua
(Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
avivacqua@dcc.ufrj.br)

Jano M. Souza
(Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
jano@cos.ufrj.br)

Sérgio Palma J. Medeiros
(Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
palma@cos.ufrj.br)

Abstract: Decision-making involves choosing between one ore more alternatives, to achieve one or more goals. To support this process, there are decision support systems that employ different approaches, supporting groups or not. Generally, however, these systems do not have great flexibility; their users have to follow preestablished decision methods. This paper, after exposing some decision-making processes, describes a system, LaSca (from Large Scale), to support decisions in large-scale groups. This system, besides allowing effective achievement of the benefits of deciding in large groups through the proper structuring of the group, also allows its users to define themselves how this structuring will happen, based or not in the existing theories on the subject. So, in addition to facilitate the decision-making process, LaSca also allows its users to decide how to decide.

Keywords: CSCW, decision-making, large groups, participatory design
Categories: H.4.2, K.6.1, M.0, M.1, M.5

1 Introduction

When the path to be followed is not clear, either due to human inability to deal with several variables in parallel, or due to lack of information or knowledge, or due to the uncertainty associated with them, a decision must be made. Deciding means that one should choose one or more alternatives among those that are presented in order to try to achieve the goal(s) desired.

The idea behind large groups is precisely to minimize the problems that arise from the issues raised above: in such groups, each individual adds different knowledge regarding the problem's domain being analysed and on which a decision must be made [Surowiecki, (06)]. This not only helps to better map the domain of the problem, but also reduces uncertainty, not to mention the fact that each individual focuses his attention on the problem's aspects that interests him or her the most, this way helping with the issue of the variables in parallel.

However, with many people participating in the decision-making process, other problems may occur, not only because of the conflicting goals and/or opinions, but also because of the issue of how to structure and manage the participation of all throughout the process. Reuniting all members of a group in one place can be extremely complicated and costly, particularly because of the need for reconciliation of schedules, an issue made worst by the generally long duration of such meetings. And the results generally are not what they could be, discouraging future participation of the group's members. Also, because the issue of physical space needed to reunite the people, the "large groups", often are restricted to 10, 20 individuals only. All this turns out going against the proposal of using large groups, not only by not obtaining the divulged benefits regarding the quality of the decision made at the end of the process, but also because all the inclusive aspects that should be inherent of this procedure do not happen, because only a small number of people actually participates in the process. That is, this "participatory design" quality of the process might end up not being participatory at all.

The development of information technology is helping to solve the problems outlined above, in order to make it possible to effectively obtain the benefits of a decision made by large groups, through an efficient management of these. As an example, one can cite the wide availability of Internet access, which eliminates the need for large spaces to accommodate the group to make the decision, which in turn allows for asynchronous participation, removing also the issue of conciliating the time availability of the group’s members.

The popularity of this concept of decision in large groups is growing every day, so that even Internet social network services as Orkut, or video games like Nintendo Wii exhibit poll features. The Yahoo! portal even has the Yahoo! Answers, a service in which someone posts a question, and others may answer it, in exchange for points. However, all this decision tools are bound to some fixed way of executing the process, heavily limiting the user's freedom in this sense. With LaSca (Large Scale), our proposal, then, is to allow the system's users to also decide how to decide, structuring the decision-making process or according to some of the best-known theories of decision-making in large groups, or even combining them, the way they find more convenient.

This approach has a lot in common with the participatory design concept [Muller, (02)], where participants (designers and users) can explore the problem, exchange ideas and create shared understanding, thus generating an environment that is neither completely design-oriented, nor completely usage-oriented, allowing, for example, users to intervene in the project and help create the tool being designed for them.

This paper is organized as follows: section 2 presents decision-making and large group theory. Section 3 describes a solution to the problem of decision-making in large groups and the prototype initially built. Section 4 shows an example of system usage. Section 5 presents a discussion and future avenues for work.

2 Related Work

As already said, making a decision involves choosing one or more alternatives in order to reach one or more objectives. According to [Marakas, (99)] and [Turban and Aronson, (01)], a decision-making process involves three phases:

1) intelligence;
2) design; and
3) choice.

In the intelligence phase, after being identified (given organizational objectives), the problem is classified, which could allow the use of standard solutions to solve it. If possible, the problem should also be divided into smaller sub-problems, so that they can be more easily solved. In this phase, the person or group responsible for solving the problem is also defined.

In the design phase, the problem is modelled to be represented in a quantitative and/or qualitative form. This is done by defining choice criteria, with the variables that characterize the problem and through which possible solutions are analysed; by defining the weights, or degrees of importance, of each of these variables; and by searching for solution alternatives. With these, the decision maker(s) can try to foresee and measure what would happen if each solution were adopted.

It is in the choice phase that a decision regarding which solution to adopt is in fact made. The alternatives proposed and forecasting results produced in the previous phases are analysed, and the solution considered the best amongst the alternatives is chosen.

The first two phases, that is, problem identification, definition of variables and presentation of alternatives, can be carried through by a single individual, by smaller sub-groups of a bigger group, or even by distinct groups. It is in the choice phase that the group must take part (this being a decision-making process in large groups), but it may or may not take part in the two previous phases.

As exposed before, the great advantage of deciding with groups is that, with a bigger number of individuals, more knowledge is added to the decision-making process. This allows better understanding of the problem, reduces uncertainty, and generates better criteria to judge the alternatives proposed, and better alternatives for problem resolution. However, group members cannot be chosen randomly: they must possess knowledge related to the problem domain so that they can provide the advantages cited. Besides that, groups must also present a certain level of heterogeneity in order to get better decisions.

2.1 Delphi Method

The Delphi method [Dalkey, (69)][Dalkey, Brown and Cochran, (69)][Turoff and Hiltz, (96)] is one of the best-known methods for decision-making in large groups. This method consists of a set of procedures used to formulate a group’s judgment with regard to a subject where it lacks precise information. The Delphi method considers that only the specialists of the problem's domain are consulted, this way forming a group that would be capable of mapping this domain in the best way possible, reducing uncertainties. Thus, the bigger the group, the smaller the error in decision.

It is considered that it is easier for a person to answer a well-formulated question than to participate in a conference, or to write a text in order to express him or herself. This way, in the Delphi method, questions are formulated, and all the group members must answer them, being part of the decision-making process.

There must be a controlled feedback of each answer after a round of questions and answers. Non-controlled iterations (e.g., face-to-face discussions) degrade the results, since individuals can be led to agree with the opinions of others for reasons other than the merits of the opinion itself. That is why, in order to produce positive results, in this method, this iteration is structured. This "structuring task" is done by a coordinator, which can be a single individual or a group. The coordinator summarizes questions and answers from previous rounds of questions, so that it is not necessary for everyone in the group to read all the answers. Moreover, there can be groups, or sub-groups, collaborating, cooperating or competing in this process, which can generate the need for a moderator. Experiments showed that this iteration leads to convergence, and that most of the time (64% of the cases), this convergence leads to values that are closer to real ones.

People make judgments of facts, or judgments of value. These experiments with Delphi regarded only facts, so that it would be possible to really measure how much a result was close to a real value.

The decision-making process in large groups is closely related to the way the members of the group express themselves. This way, methods to eliminate ambiguities in language usage in the processes of judging and estimating carried through by the groups become necessary. It must be determined if one word used by someone in a certain context has the same meaning if it is used in another one, or if two different people, when using the same word, want to say the same thing. Moreover, as comparisons must often be made to determine the best solution, a conversion of these words into numerical values must also be performed for the posterior calculation of the group's final reply. Delphi has methods to eliminate such ambiguities.

Since anonymity tends to facilitate participation, it is desirable that individual contributions be done in an anonymous way. Ideas considered bad are discarded, and those who supplied them are not identified. However, it can be interesting to use aliases or nicknames, so that the group's members can better express their opinions and debate.

During the final phase of the decision-making process, anonymity may no longer be desirable, since the identification of individuals generates greater compromise with the decisions. That is, without anonymity, group members become responsible for their opinions and decisions. Thus, they are less eager to take risks.

The final answer of the group is obtained in a statistical way, combining the individual opinions to minimize the effect of dominant individuals, irrelevant communications, and group pressure. In Delphi, the opinion of each individual has the same weight than the opinions of the other members of the group.

2.2 Large Groups

In his book, James Surowiecki [Surowiecki, (06)] presents ideas that are very similar to the concepts used in the Delphi method. To him, the group is more intelligent than the individual. However, a group must not be composed only of specialists: a certain degree of heterogeneity is desirable, in order to try to move the result away from a "common place", which often happens in environments where the decision-makers think in similar ways. This would make it possible to reach decisions of better quality, decisions that would not be reached by homogeneous groups.

Another difference refers to the feedback provided to decision-makers. The author holds that no iterations should happen during the decision-making processes in order to refine the decision, because the exposure to previous results, or the contact with the opinions of more dominant individuals, could cause less secure ones to change their opinions if they are different of the group's, or the dominant individual's, result. This could produce a lower quality decision.

2.3 Meritocracy

Meritocracy, another form of facing the decision-making process is presented in [Rodriguez, Steinbock, Watkins, Gershenson, Bollen, Grey and deGraf, (06)]. This concept is different from traditional democracy, in which the weights of the votes of the individuals of a group possess the same value (as in the two other decision concepts already presented). In meritocracy, relations of trust are established between members of a group. An individual may choose not to vote, but he or she can choose to transfer his or her decision power to a trusted individual, who would be able to better judge the situation. This other individual, if he wishes, can also pass his decision power on to others. Thus, the more people trust a certain member of the group regarding a subject, the greater will be the decision power of this individual. This "type" of meritocracy is called *Dynamically Distributed Democracy*, or *DDD*, in the paper.

In the *Proxy Vote* "type", the *DDD* concept is extended in the following way: the power, or the weight, of an individual's vote, can also increase according to the amount of other group members that trust him or her. That is, only by trusting a certain individual, someone is already increasing this individual's decision power (this "someone" does not have to give up his decision power so to increase another's). Thus, the greater the amount of people that trusts you, the greater the value of your vote. And, of course, as the *Proxy Vote* is an extension of the *DDD*, the "giving your decision power to someone else you trust by choosing not to exercise it" concept is still valid.

3 The LaSca Approach to Decision Making

The idea behind the creation of the LaSca system is to allow its users to decide themselves which is the best way to make a decision, choosing between the theories exposed, or even combining them, when possible and desired, and then go on with the decision process itself, once it is established. For example, when exposing a topic to be decided upon, the "Creator" of this "Problem" will choose if it will or will not be possible to know which user participating in the decision process emitted which opinion about the problem exposed.

Originally [Carvalho, Vivacqua, Souza and Medeiros, (07)], the first system prototype was developed in a peer-to-peer platform, the COPPEER [Miranda and Xexéo, (05)] [Miranda, Xexéo and Souza, (06)], an agent based framework for peer-to-peer applications. However, putting the application to work required a certain effort and knowledge that could alienate from the system users less willing to deal with "computational affairs". But allow anyone to be able to use the system without great difficulties; to give his or her contribution regarding any subject exposed with it, is exactly its idea. Because of that, it was decided that the classic client/server approach should be used instead, so that anyone with a browser installed in a computer and, of course, Internet connection, would be able to use LaSca. So, there would be "complications" only during the server installation, and only a single (or a few) user(s) would have to deal with it, instead of everyone, as would occur with the peer-to-peer approach.

It is important to note that LaSca follows the Multi-Attribute approach, being single objective.

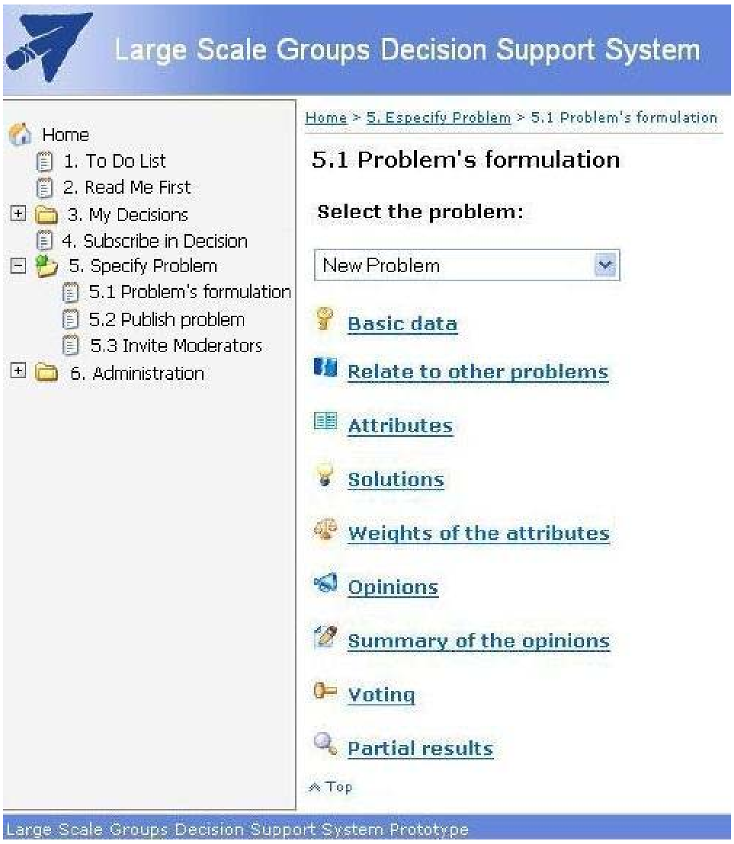


Figure 1: LaSca's Problem's formulation scree

3.1 Using LaSca

When logged on, the user can assume three roles within the system:

1) Participant;

2) Moderator;
3) Creator.

3.1.1 Participant

To be a Participant, all the user has to do is see the Problems already exposed in the system, and click on the "Participate" button next to the Problem the user wants to help with.

With the Problem already exposed, and with users assigned to help with it, the intelligence phase of the decision process is finished. Now, as a Participant, the user will be able to help with the design phase. As said before, with LaSca, the user can choose the way he or she wants to decide. Being so, the person who exposed the Problem (its Creator) may choose, for example, to do himself the design phase, alone. If he chooses to do so, the Participants will be able to act only at the choice phase. However, to make this example complete, we will describe everything the Participant can do.

At the design phase, first, the participant will expose the Attributes he or she thinks that help characterize the Problem. Then, the Participant will propose possible Solutions to the Problem. As the number of Attributes and Solutions can be enormous, they have to be moderated. After this happens, the Participant will give weights to the attributes that help describe the Problem. The weights can be relative; absolute, as a grade given to each Attribute; or absolute as a quantity of points the user can distribute among the Attributes. The way of weighting the Attributes is decided by the Creator when he or she exposes the Problem. Now the user can also give his or her opinion regarding any aspect of this particular decision-making process that he wants (that is, if opinions are allowed by the Creator). When all this is done, the design phase is finished.

At the choice phase, a matrix Attribute X Solution is shown, and the Participant then judge each Solution by giving them grades according to each Attribute. The grades in each column of the matrix are multiplied by the value of the weight of the Attribute of the column in question (if they are not unitary). That done, the elements of each row of the matrix (already multiplied by the correct weights) are then added, thereby generating the results for each Solution. Now, then, the choice phase is complete, and the results can finally be visualized.

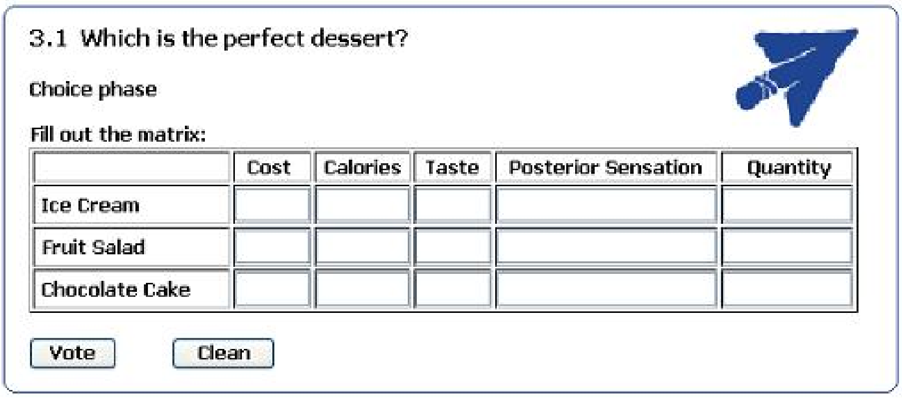


Figure 2: An example of LaSca's Attribute Vs Solution matrix

All this may or may not be done in an anonymous way; the way it can, or has, to be done, is decided by the Creator, when he first exposes the Problem.

3.1.2 Moderator

To be a Moderator, all the user has to do is verify if he or she received an invitation to be a Moderator of a certain Problem, and accept the invitation.

As a Moderator, the user can, now, if he wants, invite other users to also be Moderators of this particular Problem.

Being a Problem's Moderator, the user will receive the Attributes and Solutions given by the Participants regarding the Problem to be solved, so that he can eliminate the ones that are repeated, or alike, and then pass them on to the next level of Moderators (until the Problem's Creator is reached). If he is a Moderator of the lowest level (that is, if he did not invite any other users to be Moderators, or if no user accepted such invitation), he will receive the Attributes and Solutions directly from the Participants. Otherwise, he will receive the ones that were already analysed by Moderators from a level immediately bellow.

The Moderator also has the role of summarizing the Participants opinions regarding the Problem.

3.1.3 Creator

Finally, to be a Creator, the user has to formulate a new Problem, and publish it, so that users can become Participants of this Problem, if they want to.

To specify a new Problem, the Creator has to give it a name, and briefly describe it. He also has to say if this new Problem is somehow related to other problems already exposed (it can be derived, or the generator, of other Problems). Also, it is by using this relations that one can simulate the iterations as they are described in the Delphi Method: one creates a new Problem, and then declares that it is derived from an anterior Problem with the same name and description, which makes it the anterior iteration.

The Creator also has to say if the Participants will help assign Attributes and/or Solutions to the Problem or not. He also has to specify if the Solution will or not be a boolean Solution, that is, if, when voting, the Participant will only be able to choose one of the given Solutions, or he will be able to vote by ranking these Solutions.

It is also necessary to specify if the weights of the Attributes are unitary or not, and, if not, if the Participant will help to choose the weights or not. The Creator also has to say if the weights will be relative, or absolute, for example (as already mentioned).

Regarding the Participants opinions, the Creator will have to specify if it will be possible to Participants to see the opinions of others. Being so, he will have to decide whether these opinions can be visualized during the voting phase, or only after the decision process is finished. And he will also have to decide if he will accept only signed opinions, only anonymous, or both.

About the summaries of the Participants opinions, the Creator will have to decide if he will or will not permit their visualization as well.

He will also have to decide if the voting process (the choice phase) will be signed, anonymous, or if both cases will be accepted.

And finally, the Creator will have to decide if he will allow or not that visualization of partial results of the voting process is permitted. Then, the new Problem can now be published, and the Creator can start inviting Moderators to it.

It is by specifying all these parameters that the Creator defines which of the decision methods presented will be employed in the decision-making process in question (or if a combination of these methods will be used).

Now, as a Creator, the user will act as the last level of Moderator, finally accepting the Attributes and Solutions given by the Participants, and publishing them. It is also the Creator's role to change the status of the ongoing Problem, that is, if the Problem is now in the phase of giving weights to the Attributes, the Creator can, for example, determine that this phase is finished, and then start the voting phase, in which the Participants have to judge the Solutions regarding the Attributes assigned to the Problem, for example.

It is because of this flexibility, this large amount of possible ways of defining how a decision process can occur, that LaSca can be used do make decisions according to the theories exposed, or by combining them. It is all up to the system's user!

3.2 Tree of Moderators

As already described, it is possible for the Creator of a Problem to invite Moderators to it; and these Moderators can also invite others, and so on. This way, a tree of Moderators is formed.

LaSca's goal is to be a decision support system that deals with large crowds. Being so, it is correct to assume that the amount of input the system would receive from its users would be enormous. And it is this fact that makes the concept of "crowds of moderators" important: such crowd is essential to achieving the best possible decision in the fastest possible way. The bigger the amount of Moderators, the faster the information given by the system's users regarding a certain Problem would be filtered, thus accelerating the decision process.

This tree structure generated by inviting Moderators is also extremely important to guarantee a fast moderating process. Suppose, for example, that between the Participants many inputs that have to be filtered and the Problem's Creator there is only one level of Moderators. If there are many Moderators, they would receive few information to filter, so a lot of moderation work would have to be done by the Creator, who would still receive lots of information to filter from this single level of Moderators. However, if there are only a few, they would have to do almost all the job (which is also true in the tree of Moderators structure). Thus, considering many available Moderators, and arranging them is a tree structure, the amount of work each one would have to do would be better balanced, and this would make the moderating process go on faster.

The Moderators filter Attributes, Solutions, and Participant's opinions (all this given by the Participants regarding a particular Problem). To exemplify how this tree structure works, Attributes will be chosen.

The total amount of Attributes suggested by the Participants is divided among the lowest level of Moderators (that is, the Moderators that did not invite any other Moderator, or that had all the invitations that they sent refused). This level of Moderators will then, for example, verify if there are no repeated Attributes, or if there are ones that are alike and can be rewritten as a single one. This task done, the result of their work will be passed on to the next level of Moderators.

This process is repeated until the Problem's Creator is reached. He is the final Moderator. After he filters the Attributes he received from the Moderators under him, he can publish them. And, after the same process is repeated with the Problem's Solutions, the Attribute Vs Solution matrix of this Problem will finally be made.

It is important to notice that there is no need for the tree of Moderators to be a balanced tree or anything alike.

3.3 System Details

As already said, the client/server approach was chosen to make this version of LaSca possible. We chose the Apache Tomcat 5.5 to act as the application webserver because it would be simpler to work with it, integrating it with Eclipse. The programming language used to develop LaSca was Java.

The Model-View-Controller architectural pattern was employed to develop this system, helping organize it and enabling each aspect of the application to be developed in an independent manner, thus receiving the adequate attention.

LaSca's model layer was all generated within the PostgreSQL database management system, because of its proven reliability, and because of its license of use. Special care was given to the system modelling, so that it would be truly possible to emulate all the characteristics of the decision theories presented in this paper.

For the controller layer, the Java language was used. It was chosen because of the agility it gives to the production of code.

Regarding the view layer, the default layout of the system was created with the help of the Mindjet Mindmanager tool. After the layout was generated, the html code produced was extensively altered to suit our needs, with even the addition of some javascript code. A great deal of what was here produced was also transformed in JSPs pages, and all was stored within Tomcat.

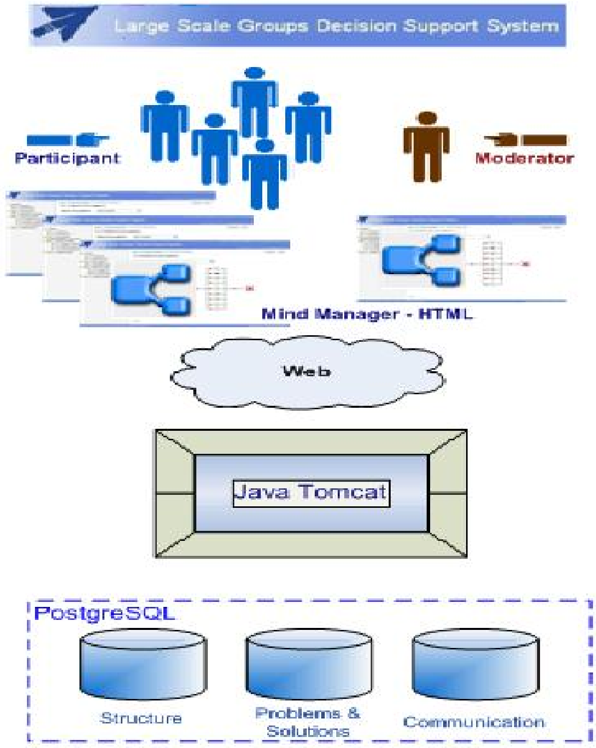


Figure 3: LaSca's architecture

A great deal of effort was spent with the system's interface, so that it would be simple to anyone to use the application. After all, we need the user to feel comfortable with the system, so that he will keep using it, proposing new Problems and helping solve others, sharing his knowledge, and contributing with whatever LaSca is being used for, and wherever it is being used; at the user's work, helping decide the best ways of spending next years budget, for example. This way, the interface of LaSca was worked and reworked many times, until a small group of people, potential users, approved what they saw, all this resulting in the interface being now used.

4 Exemplifying LaSca's Usage

To illustrate how the system works, an example is necessary. So, suppose an individual wants to know which is the best superpower for someone to have. This individual, a LaSca user, to know the answer to his question, has to specify this problem, filling all the items shown in [Fig. 1]. After he performs this task, he can publish the Problem, so that other LaSca's users can become its Participants. He can also start inviting Moderators, if he wishes.

Now, let's assume the problem's Creator structured this decision process based on the Delphi method. So, when formulating the problem, he did not allow the visualization of the Participants opinions and of partial results, but allowed the visualization of the opinion's summaries.

As this case would be the first iteration, the Creator decided that the voting (and everything else) would be anonymous. This way, it would not be possible to know who suggested determined Solution or Attribute. As the first iteration, this Problem would not be related to any other Problem already published in the system.

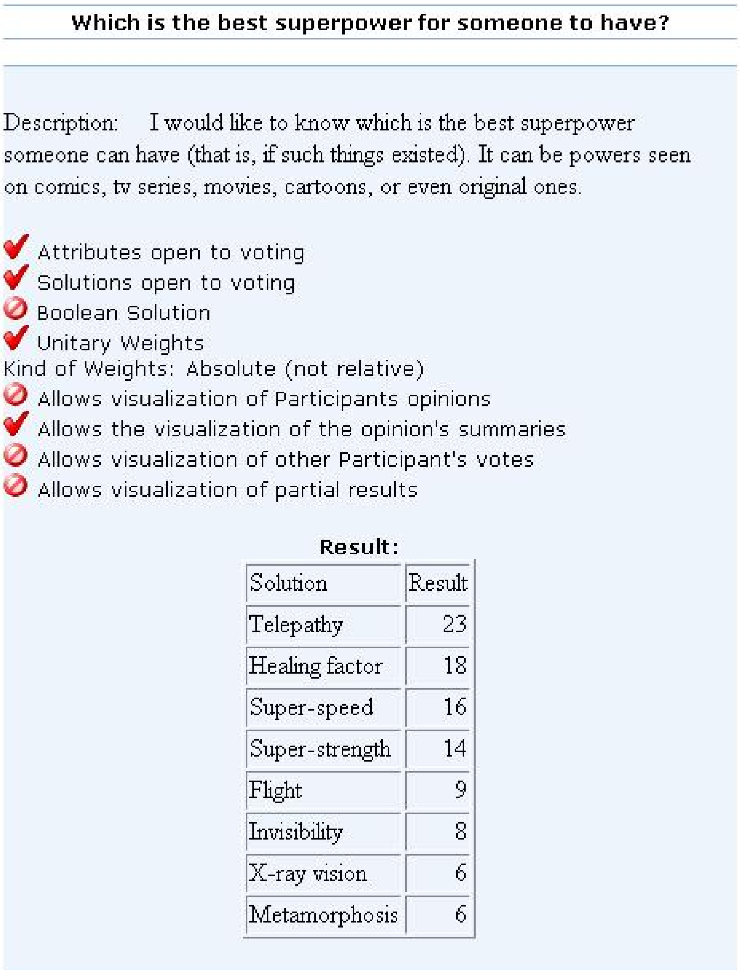
The Creator also decided that the Participants could help with the design phase, proposing Attributes to characterize the Solutions, and the Solutions that solve the Problem. Regarding the solution's choice, during the voting process, it was chosen the method of distributing points among the solutions, that is, it was not a boolean method (also possible), in which the Participant has to choose only one solution among the ones displayed.

To simplify this example, the Weights of the Attributes are unitary.

The Problem published, the Participants can now start proposing some Attributes. They do so by writing down in the appropriate space the characteristics they think a superpower should present to be considered useful, or interenting, in their opinion. This phase finished, the Moderators can check the ones proposed, and verify if there are repeated or similar ones, removing or adapting them. As examples of proposed Attributes, we have "Allows saving the world", "Allows conquering the world", "Can make you rich", "Can make you famous", "Allows personal gain", "It would be fun to have", and "Has side-effects".

Then, the Participants can propose Solutions, the same way the Attributes were proposed. Some examples are "Flight", "Super-strength", "Super-speed", "Invisibility", "X-ray vision", "Telepathy", "Metamorphosis", and "Healing factor". To illustrate the Moderator's job, suppose a Participant proposed also "Ability to become invisible". The Moderator would notice that it is the same as "Invisibility", and choose only one among these two to be one of the Problem's possible Solutions.

After all this, the voting itself, that is, the choice phase, can happen. The Participants will now choose the superpower they think is the best amongst the ones listed, according to the Attributes exposed. This, they will do by filling out this Problem’s Attribute Vs Solution matrix. Then, when the Creator closes the Problem, finishing the choice phase, the final result is calculated (in the way already exposed in the previous section), and presented do the Participants. This example's result can be seen in [Fig. 4].



Solution	Result
Telepathy	23
Healing factor	18
Super-speed	16
Super-strength	14
Flight	9
Invisibility	8
X-ray vision	6
Metamorphosis	6

Figure 4: LaSca's result screen

5 Discussion and Future Work

In this paper, the decision-making process in large groups subject was discussed. The advantages of deciding in large groups were presented, as well as the difficulties of managing the process. Then were exposed some methods to structure this process, such as the Delphi method, and then it was presented an initial implementation of our solution, LaSca, a system oriented to support those processes. With this system, it becomes possible to not only build and manage a decision-making process in large groups, but also to decide the way to manage the process itself, following or not the theories presented.

In a future implementation, the meritocratic process, in which one can transfer his decision power to someone he trusts, should be supported. Furthermore, it should be possible to establish the trust network (necessary to the occurrence of the meritocratic process) in accordance with the domain of the problem being addressed, after all, you can rely on someone to make decisions regarding a particular subject or two (or maybe even more), but not regarding everything.

Another interesting idea, not yet implemented, is to allow the user to review his or her vote, if he or she changes his or her mind due to the acquisition of new information, or new discussions. That would be allowed only while the decision-making process is still open, of course. But in a meritocratic environment, such functionality could become impractical if, for example, an individual who has previously transferred his right to vote to another changes his mind and decides to exercise it.

It should also be studied a system of rewards, in order to stimulate the use of the system by the users, besides the acquisition of the best decisions, which is already a great stimulus in itself, of course. There should also be a reward system to stimulate users to act as Problem's Moderators as well.

It would also be interesting to expand the system to also support negotiation, this way better helping to deal with conflicting goals or opinions the users might have, thus allowing the achieving of decisions that would please more people.

However, due to the extreme importance a proper shared understanding of concepts has for the whole process of decision-making (in LaSca, the Moderator has to eliminate similar Attributes or Solutions, and the Participant has to choose a Solution according to its characteristics), ontology integration is essential. And, if trust networks are indeed formed to allow the meritocratic approach, with the question of the domains of the problem being addressed, ontologies would become even more important! This way, a proper method for ontology integration, considering also the aspects of its computational complexity [Duong, Jung, Nguyen and Jo, (09)], should be studied for future versions of the LaSca system.

Still following the idea of using ontologies, and considering the huge database of decision-making processes already made, for future implementations of LaSca, CBR (Case-Based Reasoning) could be used (together with ontologies) [Bergmann, Schaaf, (03)] [Hefke, (04)] to retrieve information from the database, helping the system's users through new processes of decision making. This would be done, for example, by suggesting which Attributes and/or Solutions could be proposed for a given Problem, or even by suggesting how to fill the decision matrix, or to which other Problems from the database the Problem in question is related, thus producing bigger chains of related solved Problemas (that is, decision-making processes already finished), which would be representative of more complex decision-making processes. Thus, it could be possible to even try to identify the existence of new Problems to be solved.

With today's technological advances, it is not only possible but also very important and recommended that all those who are involved in a given situation can express their opinions and contribute with the decision-making process regarding this situation that concerns them. The idea behind the system proposed here, then, is precisely this: help in the decision-making process in large groups, giving LaSca's users the freedom to decide how to decide, thus allowing more egalitarian decisions to be made.

Acknowledgements

This research was partially supported by CAPES and CNPq.

References

[Bergmann, Schaaf, (03)] Bergmann, Ralph, Schaaf, Martin: Structural Case-Based Reasoning and Ontology-Based Knowledge Management: A Perfect Match?. Journal of Universal Computer Science, vol. 9, no. 7 (2003)

[Carvalho, Vivacqua, Souza and Medeiros, (07)] Carvalho, Gustavo, Vivacqua, Adriana S., Souza, Jano M., Medeiros, Sérgio P. J.: Large Scale Decision Making in Participatory Environmental Design. CSCWD 2007 (2007)

[Dalkey, (69)] Dalkey, Norman C.: The Delphi Method: An Experimental Study of Group Opinion. Rand Corporation (1969)

[Dalkey, Brown and Cochran, (69)] Dalkey , N., Brown , B. and Cochran , S.: The Delphi Method, III: Use of Self Ratings to Improve Group Estimates. Rand Corporation (1969)

[Duong, Jung, Nguyen and Jo, (09)] Duong, Trong H., Jung, Jason J., Nguyen, Ngoc T., Jo, Geun S.: Complexity Analysis of Ontology Integration Methodologies: a Comparative Study. Journal of Universal Computer Science, vol. 15, no. 4 (2009)

[Hefke, (04)] Hefke, Mark: A Framework for the Successful Introduction of KM Using CBR and Semantic Web Technologies. Journal of Universal Computer Science, vol. 10, no. 6 (2004)

[Miranda and Xexéo, (05)] Miranda, Mutaleci G. and Xexéo, Geraldo: A complex adaptive system approach for agent-based peer-to-peer collaborative applications. Workshop de Teses e Dissertações em Banco de Dados (2005)

[Miranda, Xexéo and Souza, (06)] Miranda, Mutaleci G., Xexéo, Geraldo and Souza, Jano M.: Building Tools for Emergent Design with COPPEER. Conference on Computer Supported Cooperative Work (2006)

[Marakas, (99)] Marakas , George M.: Decision Support Systems in the 21st Century. Prentice Hall (1999)

[Muller, (02)] Muller, M.J.: Participatory Design: The Third Space in HCI. IBM Technical Report # 01-04 (2002)

[Rodriguez, Steinbock, Watkins, Gershenson, Bollen, Grey and deGraf, (06)] Rodriguez , Marko A., Steinbock, Daniel J., Watkins, Jennifer H., Gershenson, Carlos, Bollen, Johan, Grey, Victor, and deGraf, Brad: Smartocracy: Social Networks for Collective Decision Making. Hawaii International Conference on System Sciences (2006)

[Surowiecki, (06)] Surowiecki, James: A Sabedoria das Multidões. Record (2006) (Portuguese version of “The Wisdom of Crowds”)

[Turban and Aronson, (01)] Turban , Efraim and Aronson , Jay E.: Decision Support Systems and Intelligent Systems. Prentice Hall (2001)

[Turoff and Hiltz, (96)] Turoff , Murray and Hiltz, Starr Roxanne: Computer Based Delphi Processes (1996)