

# Advancing Problem Definition and Concept Generation for Improved Product Life Cycle Management

Eswaran Subrahmanian  
Carnegie Mellon University &  
National Institute of Standards and Technology, USA  
Yoram Reich  
Faculty of Engineering  
Tel-Aviv University, Israel

## Abstract

Large engineering projects have become the norm with new additional requirements arising from the point of view of life cycle management of the product. Traditional methods for generating requirements and concept generation often have not taken into account the range and diversity of disciplines that have to be part of the process. In this paper, we propose two novel processes: one for the problem definition phase and the second for the concept generation phase. These processes, by design, attempt to bring together a variety of experiences in the early stages of product development to ensure the identification of the set of goals from conception to disposal. These methods have been used in classroom settings and in industrial contexts and have been effective. However, more detailed studies of their effectiveness and tools to support these methods in large scale projects are part of future explorations with these methods.

## 1. Introduction

What if it is discovered that after using the most advanced management techniques and tools, in a large scale project, ready for delivery that some needs of the known stakeholders and those of unrecognized stakeholders were left out. Instead of a highly successful product that was developed in record time with less than estimated resources, we might have a total failure. What we must remember that in large-scale projects, the initial stages, or what has been termed, the fuzzy front-end, is ever more critical than in smaller products, where these stages control about 80% of the cost and quality of the final product (Pahl and Beitz, 1996; Ullman, 2003). What is unfortunate is that in such projects, exercising the initial stages is more complex due to the difficulty in even identifying who the stakeholders are and what their needs or requirements are.

Product Life cycle Management (PLM) has become an important topic due to the recognition that conceiving a product is not just about its design and production, but its recyclability and sustainability, and the ability to make it satisfy a variety of customers in the market (Stark, 2004; Subrahmanian, 2005). This broad definition of the design and realization of the product requires approaches during the early stages of design that incorporate a range of disciplines, stakeholders and market needs to create products that use less energy, are recyclable and composable for different contexts. There have been efforts that are directed at Design for X (where X could be assembly, recyclability, or manufacturability). A PLM approach would require the integration of design for all of the Xs. It is precisely the comprehensive perspective of PLM that requires a thorough and deliberate inclusion for all perspectives in the early stages of the design of the product. When projects or products fail to perform, we can often identify the failure to the lack of inclusivity of a certain perspective in the early stages of the design process. Examples of such failures in the area of Information Communication Technologies for Development have been enumerated in (Tongia and Subrahmanian, 2006).

In this paper, to address the problem of inclusiveness in the early stages of the design process, we combine two methods that were developed to deal with the problem of bringing all the perspectives together to ensure that their concerns are taken into account. These methods have been in development and in use independently in real life contexts. In this paper, we propose a model to integrate the two processes of a) Problem Definition and b) Concept Generation. Both of these methods are unique in their characteristics and are different from existing methods in the literature for either problem definition or concept generation.

The first process is directed at developing the definition of the problem and the second process aids in the development of alternative viable conceptual solutions to the problem. Both processes are based on bringing multiple disciplines and perspectives together to achieve the best results possible. In this model, the approach is to first create alternative problem definitions and for the chosen problem formulations to develop alternative problem solutions concepts. The principle of the approach is to create a well-defined and rigorous definition of the problem that can serve as a record of

understanding of the participants in the formulation process. Once the definition is created, the second part of the method is to create alternative conceptual solutions again in a rigorous manner that can be evaluated against the formulation of the problem and to detail the potential conceptual solutions.

The following two sections will describe the two-part approach and they are called:

- a) Wicked problem approach
- b) Subjective Objective System for solutions concepts

The last two sections present an outline for combining the above two processes and future directions.

## **2. Wicked Problem Approach: A definition and process to developing problem formulation**

Rittel and Weber (1973) first coined the term, Wicked problem, in the context of urban and policy planning problems. The fundamental aspect of any planning problem is that it is not definable easily and that the problem definition evolves from a process of proposing a potential definition and its potential solutions. In light of the solution principle, we need to re-examine the problem definition again. As the solution principles are defined, the problem often transforms into a different problem from the one that was originally envisioned.

In the case of wicked problems, it is the ill-structured nature of the problem being solved that does not lend itself to assuming that the problem has been defined in terms of perceived needs for which there might exist a straight forward method of problem solving. Other characteristics of wicked problems include (Rittel and Weber, 1973):<sup>1</sup>

- *Wicked problems cannot be formulated definitively*
- *Wicked problems have no stopping rules*
- *Solutions are not true-or-false but good-or-bad<sup>2</sup>*
- *Wicked problems cannot be tested definitively or immediately*
- *Wicked problems do not have a well-described set of potential solutions*
- *Every wicked problem is unique*
- *Every wicked problem can be considered a symptom of another problem*
- *Wicked problems interlock and overlap and change over time*
- *The causes of a wicked problem can be explained in numerous ways*
- *The planner, or designer, has no right to be wrong*

Large scale projects such as a new aircraft series, infrastructure design (transportation systems, power systems, etc.) belong to the class of planning and execution of novel products and systems. We will now turn to the process of taming a wicked problem.

### *2.1. A process to extract the wisdom of teams*

The Wicked problem approach is a process that creates an environment to elicit quickly and reliably the wisdom of carefully constructed independent diverse groups. The process determines the fundamental problem to be solved along with the important elements and metrics of its solution.

Due to the inherent nature of Wicked problems, the primary assumption of the approach presented here is that the more comprehensive the set of perspectives that can be brought together in the definition process the better the formulation would be. The evidence for this assumption, about achieving better results by employing multiple and diverse perspectives, has been termed “Wisdom of Crowds” by Sureweiki (2004). Sureweiki argues with numerous examples that the wisdom of crowds often works better than purely expert-based approaches. We will use the term wisdom of teams<sup>3</sup> to imply wide participation of perspectives especially in the context of design problems of all kinds (Reich *et al.*, 1996). The observations have also been substantiated by studies and observations in the world of computational agents that the multiple distributed asynchronous agents working together and sharing partial solutions lead to faster and better solutions (Talukdar and de Souza, 1992). The participation of large number of people with diverse backgrounds and opinions ensures decentralization of idea generation and the provision of mechanisms for aggregation will lead to eliminating most of the possible things that can go wrong. It is in this sense that eliminating the possibility of going wrong - an important condition for a Wicked problem - can be achieved. Hence, the process proposed uses the underlying principles of extracting the wisdom of teams and the replication of the efficiency of arriving at formulations as in the case of asynchronous (decentralized) agent based problem solving systems. Towards that end, the following steps describe the process.

#### *Team formation*

Creating the team that needs to be tapped for the problem formulation process is critical, as it requires ensuring diversity. There are three dimensions to the formation of diverse groups: a) experience of the people; b) their problem solving inclination measured by the Myers Briggs (1985) test; and c) training. Besides these, we also take into account age, gender, and other relevant characteristics in defining a diverse team. The approach ensures the first condition in

---

<sup>1</sup> Taken from: Lesley Seebeck (<http://www.itee.uq.edu.au/~lesley/Complex%20Adaptive%20Systems.htm>)

<sup>2</sup> We add that there are also shades of grey, with trade-offs.

<sup>3</sup> The teams are made from a combination of experts, non-experts and others whose participation is encouraged because of the diversity of perspectives.

extracting the wisdom of crowds – Diversity of opinion.

*Idea creation*

This step is critical in creating an environment for independence and decentralization in the Wicked problem approach. To create an environment of independence, several small teams (4-5 persons each) of diverse compositions are formed; in each group there is diversity as well as the ability to present ideas by all individuals. Multiple teams (5-7) lead to the decentralization that ensures that local knowledge of the participants in the group is embodied in the creation of ideas. The process for idea creation has two steps. They are: a) expansion and b) closure. The expansion step consists of idea generation and collaboration.

The expansion step takes a starting concept and each group elaborates the same concept to identify the target concept. The collaborative step leads to sharing the target concepts across the different groups by the generation of a set of targets. These steps embody both the idea of independence and decentralization that is critical to extracting the wisdom of teams. Figure 1 illustrates this process.

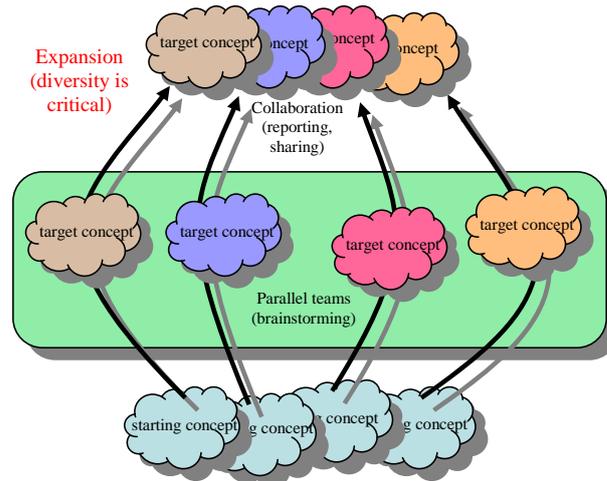


Figure 1: Expansion and collaboration Step

*Closure*

This step is where all the groups involved will share their starting concepts and analyze them to arrive at a set of target concepts that are commonly agreed upon by collaborative discussion. This step is critical from the perspective of ownership of the target concepts by the participants. The choice of the target concepts as accepted by the group will embody the trade-off inherent in the choices. The critical analysis in this step provides for the refinement of the target concepts as well as the discarding of those that will lead to the wrong approach. This step corresponds to the aggregation aspect of extracting the wisdom of teams. Figure 2 illustrates this process.

In the context of a complex process that involves a multiple inter-related concept development process, the above two steps of idea creation and closure are repeated multiple times until all the concepts are explored and their inter-relationships thoroughly examined.

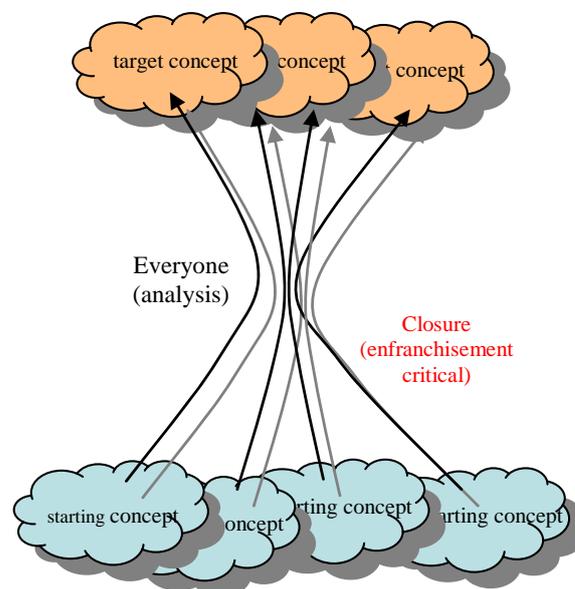


Figure 2: Closure step leading to enfranchisement and agreement on concept variety

### 2.2. Applying the process to problem formulation of Wicked problems

All design problems, wicked or otherwise, can be characterized as having the following components:

- a) Context
- b) Stakeholders
- c) Goals
- d) Design Variables and Design Space
- e) Metrics
- f) Tests

The primary goal of the process is to ensure that the basic components of the problem formulation components are being explored and get as well defined as possible. Applying the process of extraction of wisdom of teams we will use the steps of expansion and closure.

The expansion step allows each group to describe the problem elements incrementally by iterative use of ideas generation and collaboration steps. For example, in the first stage of the process, the expansion step would identify the initial components of Stakeholders, Goals and Context. In the collaboration step all groups would share the three components that often lead to each group modifying their components to reflect the input from other teams. Figure 3 is an illustration of applying the process for the goals generation component.

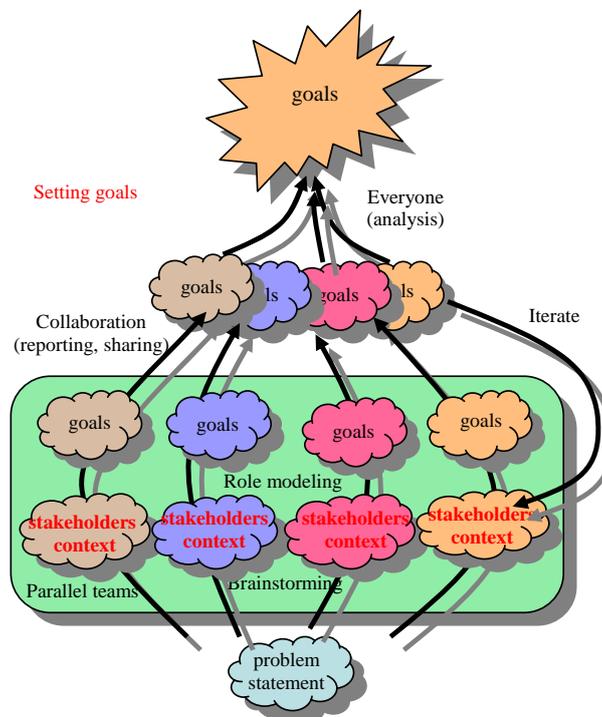


Figure 3: Extraction of wisdom of teams for the goals of Wicked problems

After the collaboration step, individual groups repeat the idea generation step to identify decision variables and the design space as envisioned by each of the teams. This step then leads to the identification of a base case solution. At this stage, each team will use their base case and their definition of the problem up to that stage and identify the metrics associated with decision variables and the goals. Having done so, each group will share the formulations for the last round of learning about how each of the teams has formulated the problem. Finally, each of the teams would revise the formulation if they deem necessary, and create the tests associated with each of the goals and the constraints. The tests can be of any kind from being evaluated subjectively by a committee or computational simulation or physical testing, as long as they are a justifiable means for testing. This final step would complete the formulation of the problem with the number of formulations equal to the number of teams. The problem formulation is then shared among the teams for discussions. In the closure step, the formulations are justified by each team, leading to the trade offs implicit in the potential choices of the problem formulations.

It is at this point that all of the concepts generated can be reformulated into a mathematical notation to the extent possible by the identification of the goals into objectives and constraints using the decision variables. A record of the process<sup>4</sup> provides the rationale and the perspectives that went into each of the formulations. The choice of the

<sup>4</sup> Currently, this is done on flip charts and paper-based templates.

formulation itself can be viewed as a step that involves closure in terms of how the exploration of conceptual solutions to the problem is undertaken.

### 2.3. Characterizing the end product of Wicked problem formulation Process

The record of the process would result in the following:

#### *Stakeholders*

The context of the problem determines the stakeholders and the assumptions associated with describing the context. Based on their identification, the following are determined to describe the problem that has been bounded by the wicked problem process.

#### *Goals*

Goals are divided into two parts; they are Objectives and Constraints. Objectives are goals that aim to achieve extremum values; otherwise, the goals are represented in the form of constraints where a certain threshold is specified. Both objectives and constraints are represented using a linear or non-linear combination of the decision variables and other parameters.

#### *Decision variables and Design space*

Decision variables,  $D_i$ , are the independent variables that characterize the problem. Design space is represented by the Cartesian product of the set of decision variables that characterize the problem as generated by the Wicked problem process.

#### *Representation of objectives and Constraints in terms of decision variables*

Objectives are represented as a collection of dependent variable  $P_i$ , through  $P_n$  such as:

$$\max P_i \text{ or } \min P_i, P_i = \sum C_i \cdot D_i$$

where each  $D_i$  is a decision variable that participates in the expression and each  $C_i$  is a parameters associated with each  $D_i$  for a given expression of  $P_i$ . All wicked problems are multi-objective problems and hence are not characterized by a single objective function, resulting in different choices for the set of objectives and constraints; hence different formulations of the problem. This is where the diversity among the teams is displayed.

The constraints are expressed as  $P_i \propto T_i$  where  $\propto$  could be one of  $<, >, =$ ,  $P_i$  are similarly expressed as before, and  $T_i$  is a threshold associated with a given constraint expressed for a given  $P_i$ .

Metrics are associated with dependent variables and decision variables. In each of these cases, the measures associated with goals and constraints are matched to have the same units in the expression. Hence each  $D_i$  has associated with it a dimensional measure or a dimensionless measure. The measure may be qualitative or quantitative depending on the nature of the dependent variables that characterize the goals or constraints.

#### *Base case solutions*

Base case solution is a characterization of a solution that corresponds to the problem being formulated. The base case is used as a means to ensure that all the variables dependent and independent are well defined and the measure corresponding to them are well determined. Base case solutions are composition of pre-existing solutions to partially or fully overlapping formulated problems with respect to the current problem at hand.

#### *Tests*

For each goal (objectives and constraints), a test is associated with it that may be qualitative or quantitative. The test measures the conformance of a given dependent variable that is expressed by an objective or constraint for a given solution in the design space. The tests are of different kinds that vary from consensus based evaluation, to simulations, and to physical tests. The tests provide a cross checking mechanism ensuring that goals can be met as per agreement of the stakeholders of the process.

For each such formulation there will exist solutions with the dimension implied by the number of objectives chosen in the formulation. Hence for each such formulation a Pareto set of solutions can be generated as alternatives.

### 2.4 Examples of the final problem formulation

#### *Example 1*

The first example illustrates the differences between two groups that are assigned to identify the goals of design of petrochemical plants. The groups' formulations show the divergence between their thinking (Herder, 1999). The goals identified by the two groups are given below.

#### Team 1

- Safety during operation
- Operability of the plant
- Acceptability for environment
- Safe startup and shutdown
- Fit for purpose

- Efficient use of raw materials
- Design should meet location-specific demands
- Control of product quality and quantity
- Maintenance
- Total life span aspects

The second team looked at the same problem but arrived at the following set of goals.

Team 2

- Early establishment of safety and environment requirements
- Reliability
- Sustainability
- Operability
- High feedstock efficiency
- Clear project and communication structure
- Environmental constraints
- Safety for employees and environment
- Adequate use of knowledge and experience of designers, operators and maintenance services

The above example illustrates the generation of different goals by two different groups in the expansion step. In the collaboration step the teams will often look at their different goals and arrive at the goals for which they are comfortable but not necessarily arrive at the same set of goals. Nevertheless, describing the entire process would be very difficult in this context. Choice of these two different sets of goals will lead to different formulations of the problem. We will now take a simple problem and represent the final result of the formulation process for a single formulation among several that were generated.

#### Example 2

The problem is to schedule crunchy and creamy peanut butter production.

The end result of the process of going through the Wicked problem approach for a team is shown below. For each team, a formulation similar to the one below is generated with different set of goals chosen by a team.

#### Context

In order to formulate this problem, the following assumptions about the context were made:

1. We are provided (Figure 4) with the demand profile for product A and B (Crunchy = type A, Creamy = type B) as a function of time.

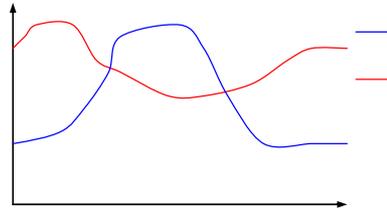


Figure 4: Demand profiles for product A and B

2. We have cut the profile into  $n$  pieces for  $n$  periods throughout the year, and assumed that all the demand for each period occurs instantaneously on the last instant of the period (i.e., trucks leave on the last day to deliver the demand). The demand for each period and product is summed and termed  $D_j(i)$  (for product  $j$ , and period  $i$ ), but this demand is actually due at the last instant of period  $(i-1)$ . As a matter of simplicity, we have shifted the demand curve back one period to allow us to provide the demand for the  $(i+1)^{th}$  period at the end of the  $i^{th}$  period. Accordingly, we basically renamed the demand for period  $D_j(i+1) = D_j(i)$ .

$$D_j(i) = \int_i^{i+1} D_j(t) dt$$

3. The production of butter is a single stage process.
4. The demand for the whole year for both butters can be met by the capacity of our plant.
5. After production of each batch (a defined quantity), cleaning is mandatory, even if the next batch will be the same product (i.e., the maximum amount produced before cleaning is at most one batch of either product).
6. We assume that partial batch production is economically infeasible due to excessive wastes, and furthermore, that demand per period (or production cycle) will always be at least one batch's worth. Also, for economical reasons, we assume Single Production Campaign (SPC) for production of A and B (as we produce all the demand for product A and then the demand for product B, or visa versa); i.e., only one switch from product to product is allowed per period.
7. Inventory costs for products A and B are given and are equal.

8. Times of no production do not hurt profits, i.e., ignoring holding costs, slowly producing over the course of the period/year is equivalent to quickly producing right at the end.

### Problem formulation

#### Defined Terms

Product A - A

Product B - B

$j$  - Used throughout this paper to signify A or B

$i$  - Refers to the period

$n$  - 1 year/ $k$  number of periods in the year. – Integer [#]

$k$  - 1 year/ $n$  period size – 1/Integer [#]

$P_{rj}$  - Production rate of product  $j$  (A or B) – Constant and known [#units/time]

$D_j(i)$  - Demand of product  $j$  (A or B) at end of period  $i$  [#units]

$t_{is}(i)$  - Starting time for production of product  $j$  in period  $i$ . [time]

$t_{ie}(i)$  - Ending time for production of product  $j$  in period  $i$ . [time]

$t_k$  - Time at end of period  $n$  (essentially  $t_n = k$ ) [time]

$I_{jo}$  - Initial inventory of product  $j$  (A or B)– Constant and known [#units]

$I_{js}(i)$  - Inventory of product  $j$  (A or B) at the start of period  $i$ . [#units]

$I_{je}(i)$  - Inventory of product  $j$  (A or B) at the end of period  $i$ . [#units]

$I_{jarea}(i)$  - Inventory\*time for product  $j$  in period  $i$ . [#units\*time]

$I_{areatotal}$  - Inventory\*time for both products over all periods. [#units\*time]

$CT_j$  - Clean-up time for product  $j$ . [time]

$B_j$  - Batch size of product  $j$  (A or B) – Constant [#units]

$B_{tj}$  - Time to make a batch of product  $j$ .

$$B_{tj} = \frac{B_j}{P_{rj}} + CT_j \text{ [time]}$$

$x_j(i)$  - Number of Batches of product  $j$  made in period  $i$  – Integer [#]

#### Goals

The Objectives and Constraints make up the Goals.

#### Objectives

1. Minimize the inventory costs during the year (defined as inventory\*time held).
2. Holding costs decrease our profit margin.
3. Minimize the Clean-up time required  $CT_j$ .
4. Labor involved in clean-up time decreases our profit margin.
5. Maximize period length  $K$ .
6. From a manufacturing standpoint, smaller the number of periods (i.e. longer time period) in a year, simpler the production schedule. Furthermore, our company would like to send out large quantities of goods at once (we do not want trucks leaving every week half full).

#### Design Space

The design space will be all the possible schedules – constructed using the decision variables  $x_j(i)$ .

#### Constraints

The constraints are:

1. Satisfy demand of products throughout the year.
2. Since we are running a single campaign production cycle, time of production of A plus B must be less than the period time  $k$ .
3. The intersection of the vectors of start and stop times for products A and B must equal zero because there is only one machine.
4. We must produce an integer multiplier of a batch (i.e., we cannot produce half a batch; only integer value batches are allowed).
5. Defines the values of period time allowed, max = 1 year, min = shortest time to produce 2 batches.

#### Decision Variables

- $[t_{is}(i), t_{je}(i)]$  for all  $i$  and  $j$

- $k$

### *Test for Objectives*

From the problem, the design should include a schedule that minimizes product inventory and satisfies demand for each type of product in each period and minimizes the cleaning time. The test for these objectives and other constraints can be done analytically or through simulation. The teams decide which methods will be used for testing the solution. Similarly for each of the constraints, the associated tests have to be identified.

The example has been made brief for providing an overview of the results and not to present a complete mathematical representation of the same. As was mentioned, mathematical notation is used for clarity and not necessarily to fit the use of a solution method.

### *2.5 Context of application of the method*

The above process has been applied in the classroom for over 6 years to teach students in the art of formulating problems (Subrahmanian, 2003). The students were divided into groups and the process was used to train them from small to sufficiently large undefined problems. The approach has been applied in industry for a number of problems and has resulted in some very substantial changes to the original conception of the problem. We have used this approach with 10 to 35 persons. We have taught and used this process at Boeing and Air Products to specific problems faced at their respective firms. The framework has also been used to identify deficiencies in problem definition with existing and past designs (Tongia and Subrahmanian, 2006).

In the next section, we describe the process of solution concepts generation that again uses multiple perspectives. Consequently, it can use the results of the formulation process to provide a better overall problem solution.

## **3. SOS: Subjective Objective System for generating robust solution concepts**

SOS is a method for generating robust product concepts (Reich and Ziv-Av, 2005; Ziv-Av and Reich, 2005). It creates the best concept that maximizes the satisfaction of a set of goals and adheres to specified constraints.

### *3.1 Definitions*

We use the following definitions in explaining the SOS method:

*Customer requirements* are product properties that are specified by the customer or the product users.

*Implementation requirements* are product properties derived from the context of the manufacturing organization including its capabilities, for example, a capability to mass-produce a product, which determines the product production cost.

*Engineering environment* is the setting in which the design takes place. In particular, it determines the product *building blocks*: product descriptors such as components, parameters, or technologies, used by designers to describe the design solution.

*Constraints* are dependencies and limitations placed on the use of various combinations of building blocks when creating the product concept.

*An optimal objective product concept* is the best product concept that can be found to maximize attaining the customer requirements but without taking into account any resource, organization, or issues such as investment, risk, knowledge, etc.

*An optimal subjective product concept* is the best product concept that is independent of functionality, but addresses all implementation characteristics such as manufacturability, simplicity, cost, risk, investment, know-how, etc.

*A decision layer* is a part of the objective or subjective concept formulation that organizes the relevant information in relation to the layer topic (e.g., product simplicity). The layer topic is either a customer or implementation characteristic.

*A decision layer weight* is the relative importance placed by the customer and/or designer on each of the decision layers.

*An optimal concept* is formed by combining the formulations of the objective and subjective product concepts and solving it.

SOS is formulated as a maximization problem, where a function that includes all contributions, is expressed and maximized subject to constraints.

### *3.2 The layers organization*

Figure 6 shows the general arrangement of  $m$  layers. There is a layer for each customer or implementation requirements. The vector  $D$  of length  $n$  denotes the engineering environment components and is the same for all layers. We assume the availability of a database of existing components or parameters, or that the design team that uses the method can provide building blocks for potential inclusion into the product concept. In many areas, these building blocks are well

identified (e.g., aircraft design); nonetheless, conceiving a product concept is a challenging task (e.g., defining the concept of a new aircraft).

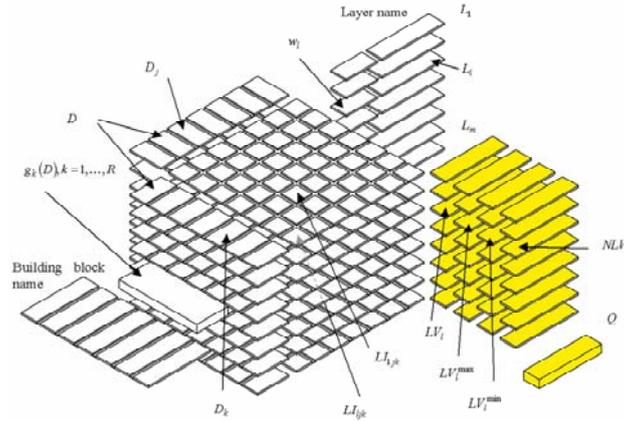


Figure 6: The arrangement of SOS decision layers

The elements of  $D$  take 1 or 0 values depending on whether they are incorporated or not in the design concept. There are constraints between the candidate design building blocks. They are modeled by:

$g_k(D_1, \dots, D_n) \propto b_k, k=1, \dots, R$ , where  $\propto$  is defined as before. In SOS, the constraints  $g_k$  are linear functions of the independent variables  $D$ . This modeling can account for diverse constraints such as:

- *Mutual exclusiveness*: If three components  $D_1, D_2, D_3$  compete to be incorporated in the product and only one could be selected then the constraint  $D_1 + D_2 + D_3 = 1, D_j = 0,1, j=1,2,3$ , makes sure that only one would be selected for the design concept.
- *Functional necessity*: When component  $D_1$  must be selected if component  $D_2$  is selected we get  $D_1 - D_2 \geq 0$ . This works if  $D_2$  is set to 1,  $D_1$  must be set to 1 also to satisfy the equation. If  $D_2$  is set to 0 (not selected),  $D_1$  can assume any value to satisfy the equation.

The matrix  $LI_l$  denotes the influence of the engineering environment components  $D$  on attaining the customer or implementation characteristic  $L_l$ . Each entry  $LI_{ljk}, LI_{ljj} \in \{-1,0,1\}$ , in the matrix specifies how much the incorporation of the two design building blocks  $D_j$  and  $D_k$  is assisting in attaining the overall value of the layer. The diagonal elements  $LI_{ljj}$  simply specify the contribution of  $D_j$  towards  $LV_l$ .

$$LV_l = \sum_{j=1}^n D_j \cdot \sum_{k=1}^n LI_{ljk} \cdot D_k$$

If we solve the following optimization problem:

$$\begin{aligned} g_k(D_1, \dots, D_n) &\propto b_k, k=1, \dots, R \\ D_j &= 0,1, j=1, \dots, n \end{aligned}$$

We would get the best combination of  $D$  that maximizes the value of the layer and satisfies the constraints. We denote this value by  $LV_l^{\max}$ . Similarly, we denote by  $LV_l^{\min}$  the minimal value of the layer obtained by the worst combination of  $D$ .

The normalized layer value is given by:

$$NLV_l = \frac{LV_l - LV_l^{\min}}{LV_l^{\max} - LV_l^{\min}}$$

The optimal solution takes into account the contribution of all layers. In the formulation, each characteristic  $l$  (whether customer or implementation) is assigned a weight  $w_l$ . Consequently, the problem becomes:

$$\begin{aligned} \max Q &= \max \sum_{l=1}^m w_l \cdot NLV_l \\ \text{subject to:} \\ g_k(D_1, \dots, D_n) &\propto b_k, k=1, \dots, R \\ \sum_{l=1}^m w_l &= 1, \quad 0 \leq w_l \end{aligned}$$

$$D_j = 0,1, j = 1, \dots, n$$

Note that  $Q \in [0,1]$ . This formulation is an integer quadratic programming with linear constraints, which is easily solved by a variety of numerical techniques.

The objective layers represent the contribution of the customer characteristics or requirements and the subjective represent the implementation characteristics. Therefore, the optimal objective solution is derived by only considering the objective layers and the subjective by taking into account the subjective layers. The objective solution becomes the target for attainment since it best addresses the customer requirements without constraining the solution by any context related aspect. Examples of using SOS and its potential could be found in (Reich and Ziv-Av, 2003, 2005; Ziv-Av and Reich, 2003, 2005).

#### **4. Interleaving SOS and wicked problem approach**

In this section, we show how SOS will be integrated with the wicked problem approach (see Figure 7). The following hypothetical scenarios illustrate this integration:

##### *Checking the impact of goals of a new project*

Once problem formulation ends, the team could check how the different goals impact the potential concept. By examining the resulting solutions, the contribution of different goals could be assessed. At the end, the engineers focus the discussion on those goals that matter and leave the rest for future discussions.

##### *Focusing on customer preferences for further study*

The marketing department provides the project team data on customer preferences. The team uses SOS to find concepts that match these preferences. In the process, the team explores local variations of the preferences. Those preferences whose variations do not change the concept are considered satisfactory. In contrast, customer preferences whose variations cause major concept changes are marked. Following the meeting, the project team asks marketing to conduct a detailed survey to make sure that those preferences do reflect future customers' desires. Marketing can now refine the survey by asking consumers who might be interested in the concept that emerges from SOS.

##### *Relaxing ad-hoc assumptions*

After the problem formulation, SOS is used to create a concept or several concepts. The concept is robust but its quality is rather poor. The team checks the constraints that are active and tries to examine whether each constraint is indeed a constraint or whether it embeds a preference that could be relaxed into an objective. As an objective it could be met to a degree and not be precise. The team transforms each such constraint into an objective, one at a time, and checks its impact on the concept. One such transformation, results in a much better concept. The team focuses on this constraint and agrees that there is no inherent reason to treat it as a constraint. The resulting concept is subsequently adopted.

##### *Solution robustness*

The design team has agreed upon a problem formulation and has found a design concept. It could now conduct a sensitivity analysis to estimate the sensitivity of the concept to the various assumptions it made in the problem formulation. The team varies the customer preferences, the market definition, the different evaluations on the layers, and any other problem definition parameter. These variations correspond to exercising various "what-if" scenarios. The team records whether the solution changes or not. While exercising most changes to the assumptions, the solution does not change; however, one change causes the solution to change. The team examines this change and its cause and determines that the change in the concept is minimal and if this part of the design is deferred a little, enough information about the cause would be accumulated, allowing to make the right decision at that point in time. The sequence of design activities needed to support this conclusion is recorded.

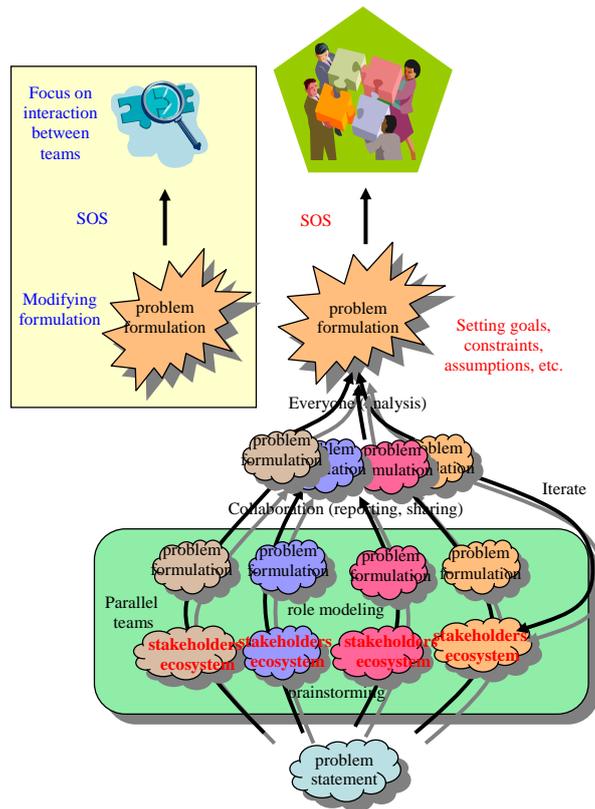


Figure 7: Extraction of wisdom of teams for the goals of Wicked problems and conceptual solutions

## 5. Discussions and Future work

Most problem definition methods such as brainstorming and De-Bono's six hats are based on creating ideas that attempt facilitating a broad variety of perspectives to provide input to both the problem definition and solution process. While these methods have been useful anecdotally, their effectiveness is questionable (Paulus *et al.*, 1993). In our approach, we have attempted to arrive at a process that is more structured than the prior methods and is directed at designing the problem definition process explicitly to ensure focused and broad participation of the stakeholders. Our approach addresses the issue of the inherent wicked nature of most design problems and scale in terms of participation. The evaluation of our approach for its effectiveness is part of the future work.

In a similar vein, we have developed a concept generation process that acknowledges the process of negotiation of product attributes and multiple-objectives inherent in the product life cycle process. Hence, not only does this process generate solutions while other methods merely select between previously generated solutions; even if we used it only for selection, it extends and modifies the prior concept selection processes such as Pugh's method and Quality function deployment method. The richness of SOS allows for incorporating platform-based design for product variety. This method has been tried in several classroom projects and industrial contexts. The main effort now will be in integrating these two methods in the context of real world projects to experiment and evaluate their effectiveness.

Any project that will use our approach will require several iterations of the two processes depending on the complexity and size of the problem. For example, if one were to design a Rural-Airlink system, the first task would be to derive the broad problem definition by including in the scope of the problem the available infrastructure, the scope of the air-traffic problem, scope of the aircraft being designed for the purpose, scope of the support and maintenance skill base, and several other aspects. Consequently, the project plan would have to accommodate the recursive and multi-part nature of the Rural-Airlink project. The project plan will start with the definition of the overall problem just as in the case of the design of the chemical process. The overall problem is then decomposed into several sub problems, each of which is a Wicked problem in itself, and for each of the sub problems, the conceptual solutions have to be derived. The right compositions of the conceptual solutions to each problem will have to be composed to address the overall problem.

A fundamental characteristic of both the processes is the interleaving of problem formulation and solution concept steps. The approach allows for continual refinement of both. It accommodates multiple perspectives at different levels of detail of the definition and solution. The formulation part can involve very diverse participants in the early definition and the SOS method can employ specific classes of experts. The two methods provide the flexibility to define the scope and scale of elicitation of the wisdom of teams.

The approaches described in this paper have been conducted with paper and pencil or simple computational means but it would be best done using dedicated computational support tools that do not presently exist. One of the future objectives of this work is the development of tools for aiding the above processes. The development of support tools

would be useful and would contribute to the early stages of conceptual design where there are very few tools available today.

#### *Acknowledgment*

The authors would like to thank Steve Stadelmeir, Sarosh Talukdar and Arthur Westerberg who are key participants in the development of the problem formulation course.

#### **References**

1. I. Myers Briggs and M.H. McCaulley, *Manual: A Guide to the Development and Use of the Myers-Briggs Type Indicator*, Palo Alto, CA, Consulting Psychologists' Press, 2<sup>nd</sup> Edition, 1985.
2. E. De Bono. *Six Thinking Hats*, Penguin books, 2000.
3. P.M. Herder, *Process Design in a Changing Environment - Identification of Quality Demands Governing the Design Process*, Ph.D. thesis, Delft University Press, ISBN 90-407-1818-0, The Netherlands, 1999.
4. G. Pahl and W. Beitz, *Engineering Design - A Systematic Approach*, Springer Verlag, London, 1996.
5. B. P. Paulus, T. M. Dzindolet, G. Poletes, and L. C. Mabel, Perception of performance in group brainstorming: The illusion of group productivity, *personality and social Psychology Bulletin*, 19:78-89, 1993.
6. Y. Reich, Y., S.L. Konda, S. N. Levy, I. A. Monarch, and E. Subrahmanian, Varieties and issues of participation and design, *Design Studies*, 17(2):165–180, 1996.
7. Y. Reich and A. Ziv-Av, A comprehensive optimal product concept generation framework, in *Proceedings of the 15th International Conference on Design Theory and Methodology (DTM)*, (New York, NY), ASME, 2003.
8. Y. Reich and A. Ziv-Av, Robust product concept generation, in *CDROM Proceedings of the 15<sup>th</sup> International Conference on Engineering Design (ICED)*, The Design Society, 2005.
9. H. W. J. Rittel and M. M. Weber, Dilemmas in a general theory of planning. *Policy Sciences*, 4: 155-69, 1973.
10. J.J. Stark, *Product Lifecycle Management: Paradigm for 21st century Product Realisation*, Springer Verlag, 2004.
11. E. Subrahmanian, R. Sudarsan, S. Fenves, S. Fofou, S. and R. D. Sriram, Product Lifecycle Management Support: A Challenge in Supporting Product Design and Manufacturing in a Networked Economy, *International Journal of Product Life Cycle Management*, 1(1),4–25, 2005.
12. E. Subrahmanian. A.W. Westerberg, S.N. Talukdar, J. Garrett, A. Jacobson, C. Paredis and C. Amon, Integrating Social Aspects and Group Work Aspects in Engineering Design Education, *International Journal of Engineering Education*, 19(1):75-80, 2003.
13. Surowiecki J., *The Wisdom of Crowds*, Random House, 2004.
14. S.N. Talukdar and P.S. de Souza., Scale Efficient Organizations. In *Proceedings of the 1992 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1458-1463, Chicago, Illinois, Oct.18-21, 1992.
15. R. Tongia and E. Subrahmanian, ICT for Sustainable Development (A Design Challenge), *Proceedings of the ICT for Development Conference*, Berkeley, May 2006.
16. D. G. Ullman, *The Mechanical Design Process*, 3<sup>rd</sup> ed, McGraw-Hill, New York, NY, 2003.
17. A. Ziv-Av and Y. Reich, SOS – Subjective Objective System for generating optimal product concepts, in *CD-ROM Proceedings of the 14th International Conference on Engineering Design (ICED)*, The Design Society, 2003.
18. A. Ziv-Av and Y. Reich, SOS – Subjective objective system for generating optimal product concepts. *Design Studies*, 26(5):509-533, 2005.