

2011-01-01

Improvement of Complex Decision Making using System Dynamics and Zachman Framework techniques

Bharath Bhushan Dantu

University of Texas at El Paso, bharath1815@gmail.com

Follow this and additional works at: https://digitalcommons.utep.edu/open_etd



Part of the [Instructional Media Design Commons](#)

Recommended Citation

Dantu, Bharath Bhushan, "Improvement of Complex Decision Making using System Dynamics and Zachman Framework techniques" (2011). *Open Access Theses & Dissertations*. 2265.
https://digitalcommons.utep.edu/open_etd/2265

This is brought to you for free and open access by DigitalCommons@UTEP. It has been accepted for inclusion in Open Access Theses & Dissertations by an authorized administrator of DigitalCommons@UTEP. For more information, please contact lweber@utep.edu.

IMPROVEMENT OF COMPLEX SYSTEM DECISION MAKING USING SYSTEM
DYNAMICS & ZACHMAN FRAMEWORK TECHNIQUES

BHARATH BHUSHAN DANTU
Department of Industrial Engineering

APPROVED:

Eric D Smith, Ph.D., Chair

Tzu-Liang (Bill) Tseng, Ph.D., CMfgE

Jose Humberto Ablanedo Rosas, Ph.D.

Benjamin C. Flores, Ph.D.
Acting Dean of the Graduate School

Copyright ©

By

Bharath Bhushan Dantu

2011

IMPROVEMENT OF COMPLEX SYSTEM DECISION MAKING USING SYSTEM
DYNAMICS & ZACHMAN FRAMEWORK TECHNIQUES

By

BHARATH BHUSHAN DANTU, B. Tech.

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Industrial Engineering

THE UNIVERSITY OF TEXAS AT EL PASO

December 2011

Acknowledgements

I express my gratitude to all my family members and friends for their support without which this thesis would not have been possible. I would like to express my sincere thanks and gratitude to Dr. Eric Smith for his support from the beginning of my journey as a Graduate student at UTEP. I would like to thank Dr. Eric Smith, Dr. Bill Tseng and Dr. Ablanado Rosas for being my committee members and dedication of their valuable time with my research.

Abstract

Modeling representations of complex systems useful for the examination of changes that occur in any part or whole of these systems must allow for the performance of a requisite variety of essential analyses, in order to cover decision contingencies that have the potential to sway outcomes to extreme values. The purpose of sophisticated modeling techniques is to make the model as realistic a reflection of the real world as possible, considering all constraints of available data, analyst time availability, and computational resources needed to evaluate the model. The fundamental purpose of decision making methods is to create a quantitative representation of different choices which enable incorporation of uncertainties and different representations of the decision maker's preferences for various possible outcomes. Decision analysis can validate scenarios regarding decisions, help to compare choices quantitatively, allow for rapid assessment of the effects of variations in assumptions on the optimal choice, and provide a mechanism for evaluating a given decision along various outcome dimensions like survival expectancy, performance and costs.

Models for complex healthcare decisions must incorporate consideration for the usual multiplicity of important factors, interacting feedback loops among these factors, and the dynamic nature of the full diagnostic arena. A diagnoses modeling technique that has the requisite variety of relevant considerations is presented. The technique has the potential to overcome mandatory time criteria, while considering the competence and robustness of high importance diagnostic decisions. In this study, descriptive narratives dictated by examining physicians who were directly involved in the diagnosis and treatment of patients were examined in detail not only to extract key factors involved in medical decision making processes, but also to illustrate the wide ontological origin of key decision making factors. Important factors in the

narratives were identified and mapped with a new System Dynamics methodology that incorporates a Zachman Framework for establishing the overall scope and context of the full medical decision making context within the modern medical enterprise.

Table of Contents

Acknowledgements	iv
Abstract	v
Table of Contents	vii
List of Figures	ix
Chapter	
1. Introduction.....	1
2. Problem Statement	5
3. Soft System Methodology	7
4. System Dynamics.....	9
4.1 Research Center in Health Care System.....	13
4.2 Federal & State Funding and Insurance Coverage.....	15
4.3 Medical Billing Process.....	16
4.4 Request Management System.....	18
4.5 Generalized Health Care System with Flow rates	19
5. Review of the history of Soft Systems Modeling	20
6. Zachman Framework.....	32
6.1. Analysis Process.....	39
6.2. Advantages of the Zachman Framework.....	39
6.3. Disadvantages of the Zachman Framework.....	39
7. System Dynamics on Zachman Framework	41
7.1. Integration of System Dynamics on Health Care System	41
7.2. Comparative a narrative (oral or written) with visualization.....	42
7.3. System Dynamics on Diagnosis Interview	45
7.4. Zachman Framework.....	47
7.4.1. Rows or Perspectives	48
7.4.2. Columns or Interrogatives	48
8. Justifications (Comparision with other tools/techniques)	50
8.1. Swim lanes using Sequential Diagrams	50
8.1.1. Sequence Diagram on Black Hawk Incident	51

8.1.2. Sequence Diagram on UAV Mishap	53
8.2. Interpretive Structural Modeling Process	54
8.2.1. Procedure	54
8.3. Cyclomatic Complexity	58
8.3.1. Analyzing Diagnostic interview using Complexity measurement.....	60
9. Applications	62
9.1. Acedamics.....	62
9.1.1. System Diagrams Model	63
9.1.2. Zachman Framework.....	63
9.2. Defense	64
9.2.1. Black Hawk Incident	65
9.2.1.1. Analysis on the Incident using SD Technique	69
9.2.1.2. Analysis on the Incident using Zachman Framework	70
9.2.2. Unmanned Aerial Vehicle	71
9.2.2.1. Analysis on the Incident using SD Technique	71
9.2.2.2. Analysis on the Incident using Zachman Framework	72
10. Future Work.....	73
10.1. Vensim Modeling	73
10.2. Word Parser	73
10.3. Applying the concept on other techniques.....	74
11. References.....	75
12. Curriculum Vita	79

List of Figures

Figure 1.1: Framework of Apex System	4
Figure 2.1: Attribute Substitution illustrated with hierarchical decomposition	6
Figure 3.1: Basic Structure of Soft System Methodology.....	8
Figure 4.1: Generalized Health Care System	12
Figure 4.2: Role of Research Centers in Health Care System.....	13
Figure 4.3: Role of Federal and State Governments in the Insurance Coverage.....	15
Figure 4.4: Medical Billing Process.....	16
Figure 4.5: Request Management System.....	18
Figure 4.6: Generalized Health Care System including flow rates using Vensim	19
Figure 5.1: Complementary sides of a system.....	22
Figure 5.2: Encompassing abstraction levels	23
Figure 5.3: Levels with complementary aspects at different levels of abstraction	23
Figure 5.4: Effects of complementary levels	24
Figure 5.5: Idealized, perfect correspondence in mathematics	26
Figure 5.6: Semantics and Syntactics in an expressive system	28
Figure 5.7: Self Reference Diagram	29
Figure 6.1: General Enterprise Zachman Framework Model.....	37
Figure 7.1: System Dynamics depiction of influences noted during interview	45
Figure 7.2: Zachman Framework for the Medical Enterprise	47
Figure 7.3: Medical Interview shows with a SD overlaid on Zachman Framework	49
Figure 8.1: Sequence diagram illustrated the Black hawk incident.....	51
Figure 8.2: Sequence diagram illustrated the Mishap of Unmanned Aerial Vehicle	53
Figure 8.3: Interpretive Structural Modeling.....	54
Figure 8.4: Matrix model.....	56
Figure 8.5: Example for Cyclomatic Complexity.....	58
Figure 8.6: Diagnostic case with System Dynamics technique	60
Figure 8.7: Diagnostic Interview with Cyclomatic Complexity	61
Figure 9.1: System Dynamics Model on CCAM.....	63
Figure 9.2: CCAM with SD overlaid on Zachman Framework	63

Figure 9.3: Black Hawk incident with System Dynamics model	69
Figure 9.4: Black Hawk incident shows with SD overlaid on Zachman Framework.....	70
Figure 9.5: UAV Mishap with System Dynamics model.....	71
Figure 9.6: UAV Mishap shows with a SD overlaid on Zachman Framework.....	72
Figure 10.1: Model for Population count with Vensim Modeling Technique	73

Chapter 1: Introduction

Today, technology is what drives our life. New systems are needed all the time to stop competitors gaining some form of advantage but where do these systems come from? [1] Systems are based on computers and that system engineering is closely allied to software engineering. This view is frequently enhanced by the published papers that rarely attempt to qualify the definition of a system with adjectives to narrow the scope of the system being covered. There are many ways of looking at a system and attempting to define what it can be, but first let us examine standard texts typified by dictionaries and encyclopedias. These yield the following:

- System: A set of things considered as a connected whole or complex thing or parts.
- Complex: Consisting of parts, composite, complicated
- Thing: Whatever is or may be an object or thought
- Parts: Portion allotted
- Connected: Joined in sequence, coherent
- Object: Person or thing to which action or feeling is directed

Thus virtually any object can be thought of as being a system whether it has some action or not but it must be able to be conceived as a whole entity by an observer.

The system definition may be presented in different forms. Antonin Wild preferred the following wording:

"A system is a bounded physical entity which achieves in its environment a defined purpose through the interaction of its parts". [2]

A system must be a physical entity. If it is not a physical entity, sufficient assumptions have to be made to create a physical system which can be defined as required by the definition. Prediction

of reliability for a drawing or a schematic of a circuit is possible only after careful consideration of how this drawing or schematic will be converted through a series of technological processes into the final product, a physical entity. The prediction will afterwards refer to that hypothetical physical entity and not just to the drawing or schematic.

A system has a boundary. Both the functional and physical boundaries must be defined. The physical boundary is described by identification of the parts/departments which form the boundary. The definition of the functional boundary includes description of all loops, impacts other influences crossing the physical boundary. Outside the system boundary is considered for the purpose of this definition the environment of the system.

The design implementation and use of adequate performance measurement and management frameworks can play an important role if organizations are to succeed in an increasingly complex, interdependent and changing world. There are certain issues which require further focus in order to make measurement system much effective in supporting the decision making techniques. The environment within which most organizations operate is changing rapidly. Organizations failing to adapt and respond to the complexity of the new environment tend to experience, implementation and use of adequate performance measurement and management frameworks is one of the major challenged confronting organizations and can play an important role in their success. Several factors elevate the issues behind the dissatisfaction with traditional performance measurement systems and the new environment faced by most organizations. The key factors are like a better understanding of the interrelationships and the consideration of trade-offs between performance measures, the dynamism of organizations and the dynamism of measurement systems etc. [2]

Here is an example to discuss more about significance in analyzing a complex system. Modeling and Simulation have the capability to play an important role in designing and analyzing large-scale in a safe and cost-effective way. The large scale complex systems can be defined as system of systems consisting of number of components such as machines, technical systems, humans depending on the field we are working to reach the effective goals. One of such example is analyzing the Human Behavior in Complex Systems.

A human behavior in a complex system consists of several interactions with technical devices, other humans, physical and mental interactions with surroundings etc. The author in this experiment used a simulation tool called “Agent-based modeling and simulation” technique. But we are only focusing on the importance of analyzing a complex system and the importance of analyzing in order to reach the final goals. Below is the simulation model of the” Human behavior in a complex system”,

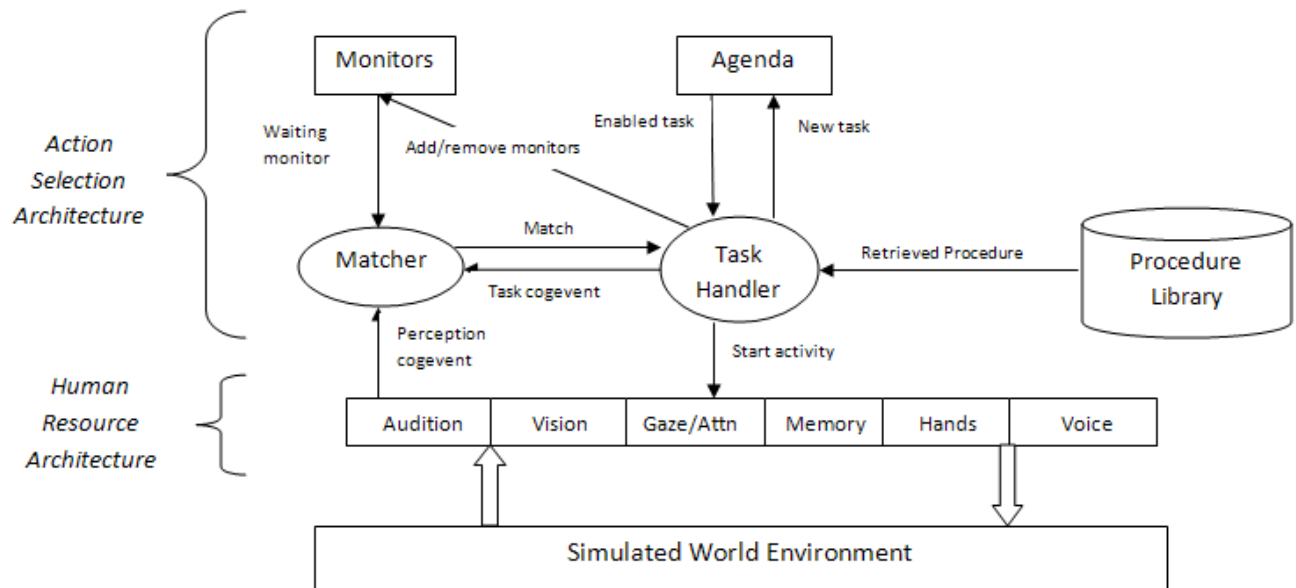


Figure 1.1: Framework of Apex System

In an enterprise with such a complex system, it is most essential to understand and have a clear vision on activities happening in each department of the system. In order to achieve that an abstract model is necessary that can illustrate the entire system process even including the internal loops, impacts and influences.

Chapter 2: Problem Statement

With a broader view on a complex system, it is very difficult to restrain all the errors in each department of the system. Taking this as an initiative, we started analyzing several existing Soft System Methodologies and found their advantages and disadvantages. For better results we have chosen one of the complex systems “Health Care”.

Several heuristics and biases, notably representativeness, anchoring, base-rate neglect and the conjunction fallacy, summarized by Smith [3], are now considered by Kahneman to be instances of a super-heuristic called attribute substitution [4]. Human judgment is mediated by this heuristic of attribute substitution when, without realizing that it is so,

“an individual assesses a specified target attribute of a judgment object by substituting another property of that object -- the heuristic attribute -- which comes more readily to mind. Many judgments are made by this process of attribute substitution.”

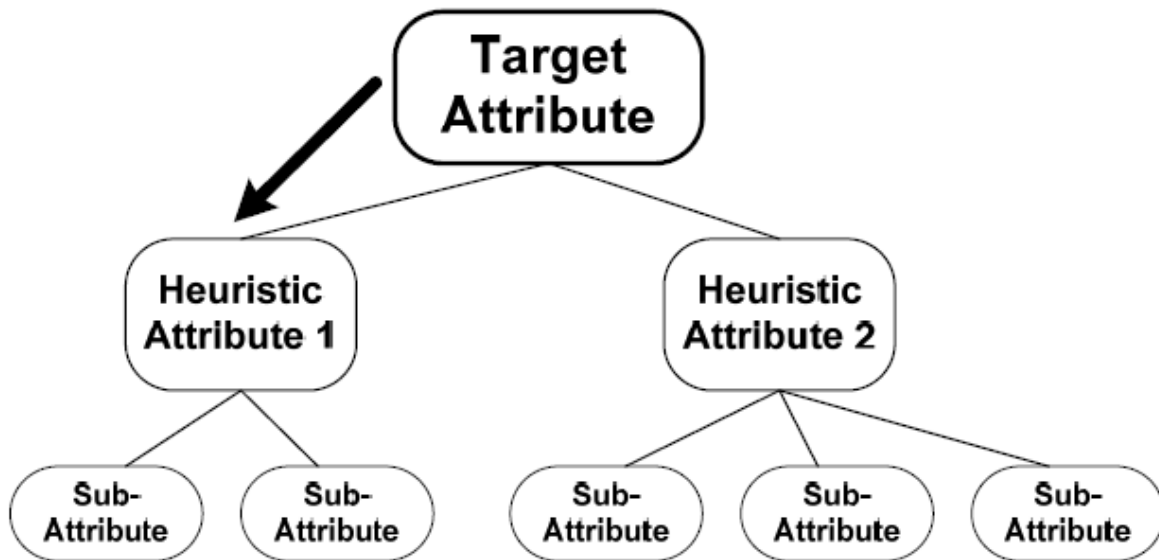


Figure 2.1: Attribute Substitution illustrated with hierarchical decomposition

Redelmeier and Shafir [34] described how a technique such as considering each alternative in relation to the status quo is more effective than considering all alternatives at once and only in relation to themselves. More biases that result from the consideration of alternatives exist, and are presented below.

Specific de-biasing techniques have been shown to be effective under specific circumstances. Principally, such techniques involve increasing the decision maker's awareness of possible cognitive biases, and then mandating a procedure that has been shown to reduce the particular bias. However, in practice, few professionals remember the presence of biases, and almost none implement proven de-biasing processes. What is needed is a single abstract model of biasing, and the ability to apply the model generally. [35]

Chapter 3: Soft System Methodology

Soft-system methodology (SSM) was first introduced by Checkland in 1981 in his book, *Systems Thinking, Systems* [6]. It is a methodology for analyzing and modeling hard-to-define and complex systems that integrate technology/hard systems and human/soft systems. Checkland proposes that the same methods used for engineering technology may not work as well for the more unpredictable and complex human side of the system. SSM addresses “fuzzy” problems that occur when objectives are unclear, multiple objectives exist, and where there may be several different perceptions of the problem. SSM recognizes that different individuals will have different perceptions of the situation and different preferable outcomes. It recognizes these differences, and explicitly attempts to take these into account from the outset to ensure that the results of the analysis are acceptable to all parties concerned [5]. Use of an SSM approach does not attempt to define a single right method of action, but through an iterative process, defines an acceptable improved path of action. SSM has been used to identify the value, impact, and barriers to information access and use, as related to quality of health care by a group of regional directors for a Mexican national health care organization [7].

SSM articulates a learning process which takes the form of an enquiry process in a situation that people are concerned. This process leads to action in a never ending learning cycle: once the action is taken, a new situation with new characteristics arises and the learning process starts again. The basic structure of SSM rest on the idea that in order to tackle real world situations, we need to make sure that the real world is separated from the systems thinking world. This distinction is crucial for SSM because that assure that we will not see systems ‘out there’; that is in the real world. We perceive and evaluate, take action which itself becomes part of this flux which lead to next perceptions and evaluations and to more actions and so on. It follows that

SSM assumes that different actors of the situation will evaluate and perceive this flux differently creating issues that the manager must cope. Here, SSM offers to managers the systems ideas as a helpful weapon to tackle problematic situations arising from the issues. [8]

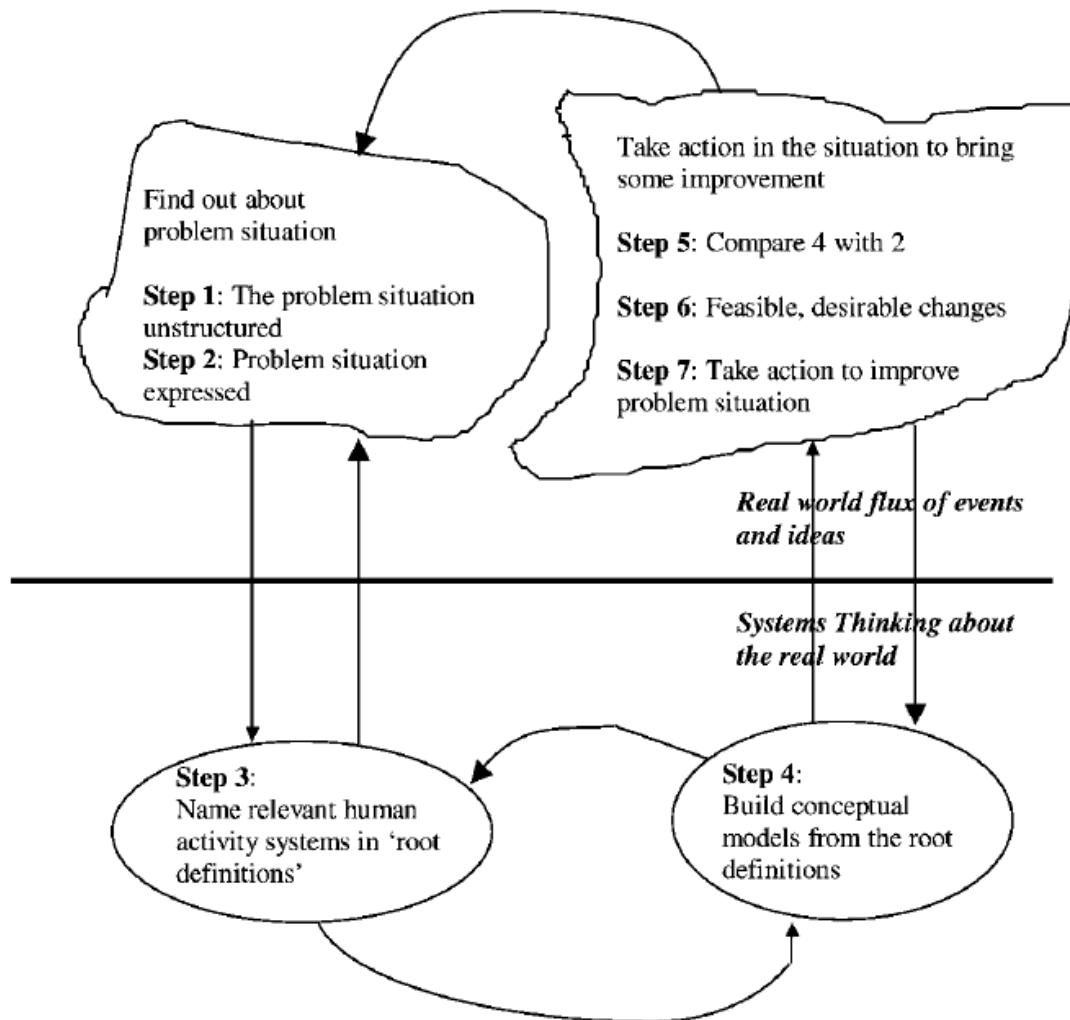


Figure 3.1: Basic Structure of Soft System Methodology

Chapter 4: System Dynamics

The System Dynamics can provide a useful framework in which to explore some of these issues and finally help in improving the organizational performance.

In reality, these issues which we highlighted or focused helped in developing of several frameworks of different characteristics and of varying complexity. There is a general belief that the measurement systems should be designed, implemented and used so that they enable continuous performance improvement rather than simply control or monitor. The systems or procedures that currently exist do not really provide detailed information for the decision makers to effectively manage the performance. In reality, diverse reasons may be highlighted to justify the reasons behind the failure to meet the requirements in order to improve the effectiveness of the performance which can lead to great success. One of the major reasons to fail might be related with the large and complex amount of information they provide, conjointly with the absence of approaches to assist decision makers understand, organize and use such information to manage organizational performance. Due to the limited information processing capabilities of the human brain, we believe that the use of approaches like System Dynamics can be very valuable to assist the decision makers to understand and organize this information in order to develop the performance of the system or organization. In past, people used to utilize the cognitive maps in order to identify the factors affecting performance and their relationships. But due to the limitations like cognitive maps like interconnections between factors, the existence of non-linear interactions between different elements, delays and feedback loops, this process failed to provide sufficient information for the participants. To deal with the dynamic complexity in the social systems and to infer dynamic behavior, quantitative simulation is required. From all the above points that are mentioned, we believe that the translation of qualitative diagrams into a

simulation model using the System Dynamics approach can enrich the analysis and provide very useful insights for the design of measurement systems. Our actions in order to improve the performance measurements sometimes may end up with decline in other performance measurements due to several reasons as mentioned. Due to time delays in feedback, some actions may produce long-run affects in performance different from their short-run effects. One other reason could be the way of emphasizing the need of measurement systems by other frameworks like Balanced Scoreboard, Results and Determinants Framework and Performance Pyramid to make explicit the trade-offs between the various performance measures, but are vague in how to deal with these trade-offs. One other importance of knowing the performance measurements is to evaluate the effectiveness of different policy alternatives or courses of actions to improve performance. The long term delays and systemic effects of actions are the main reasons in differentiating the performance between good and bad. Usage of System Dynamics can play an important role in achieving this objective in the most effective way. System Dynamics enables a greater understanding of the effects of actions already implemented but also of the effects of alternative actions to be considered for implementation. It is widely recognized that organizational performance measurement systems should be dynamic, evolving over time; most organizations have only static performance measurement systems. We believe that the System Dynamics approaches allow decision makers to review and update systematically the measurement system, taking into consideration these changes. System Dynamic models can help decision makers gain insights of system's behavior over time which may reveal very valuable to review. System Dynamics have individually proved their potential to inform and support decision making, working as a vehicle to reach consensus, ownership and commitment among

decision makers, we believe that their effective use in the context of performance measurement can facilitate the implementation phase.

The design and control of isolated measures is a valuable exercise in the sense that it informs decision makers about how the organization is performing against goals, and it assists in identifying an organization's strengths and weaknesses. System Dynamic models are frequently developed and used to represent, analyze and explain the dynamics of complex systems. The dynamic nature of the system is defined by its structure and the interactions of its parts. The main goal of System Dynamics is to understand how this behavior is produced and use this analysis to predict the consequences over time of policy changes on the system. System Dynamics can also help in providing a very useful insight when supporting the performance measurement and management process. In order to understand much more about this concept of improving the process measurement we implemented this technique on complex Health Care System.

Below we can observe how system dynamics helps in understanding about a scenario or organization in closer way. The loops and influences among the different entities/departments are clearly indicated and analyzing each department individually is also so easy with the help of System Dynamics.

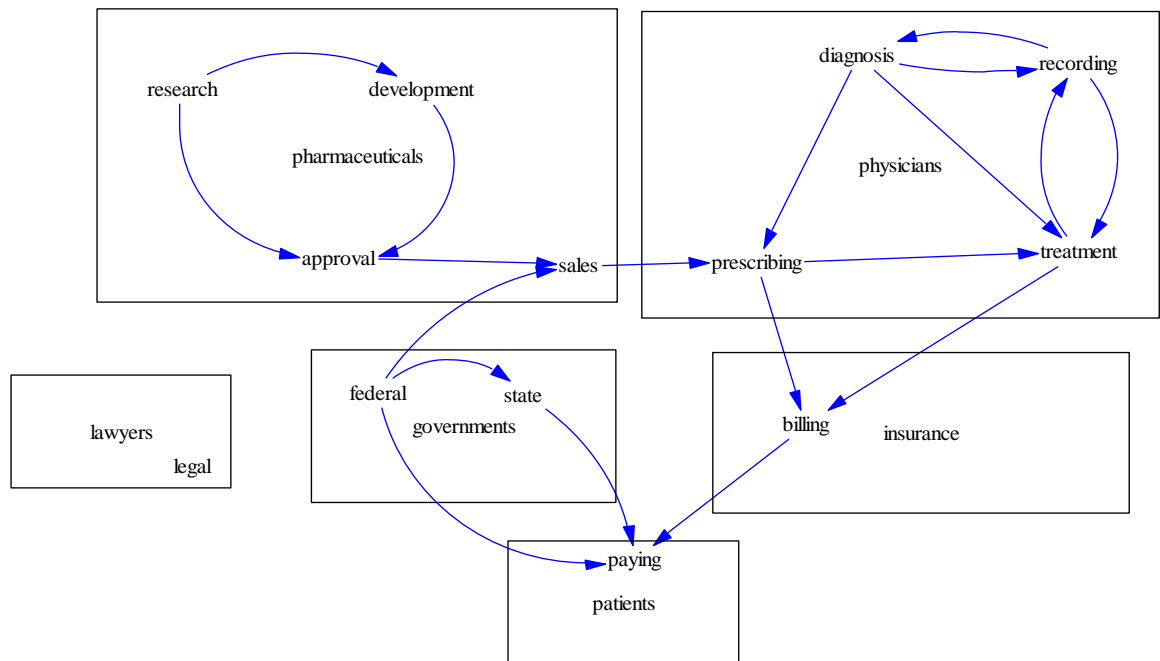


Figure 4.1: Generalized Health Care System

The above diagram shows us the generalized health care system in a broad way. From the diagram it's very clear that to form a good health care system, mutual understanding and effective work from all inter departments are essential. Here in this diagram, different departments like Pharmaceuticals, Physicians, Insurance and State and Federal government have their involvement in making a health care system more effective. In pharmaceuticals departments, teams perform research work and they regularly send new drugs to the development and get approval from the inspection team. Once the drugs are approved then they get released to the market for sales.

Based on the prescription given to the patients, drugs are changed and newly released; powerful and effective drugs are used. In other way, patients are recommended to change the prescription depending on the new drugs in market. Once the patient is treated and is eligible to

claim a medical insurance, the billing department takes care of this issue. The Insurance agency reviews the medical claims given by patients in different levels and once everything is set they provide medical benefits to the patient. Federal and state government provides funds to the people who do not have medical insurance and has eligibility to get this facility provided by government. In this paper, we have a clear explanation on the role of Research center in the health care system, Importance of Federal Government in the Health care department and Medical Billing process. Emphasizing on these areas really helps in understanding how effective a health care system works.

4.1 Research Center in Health Care System

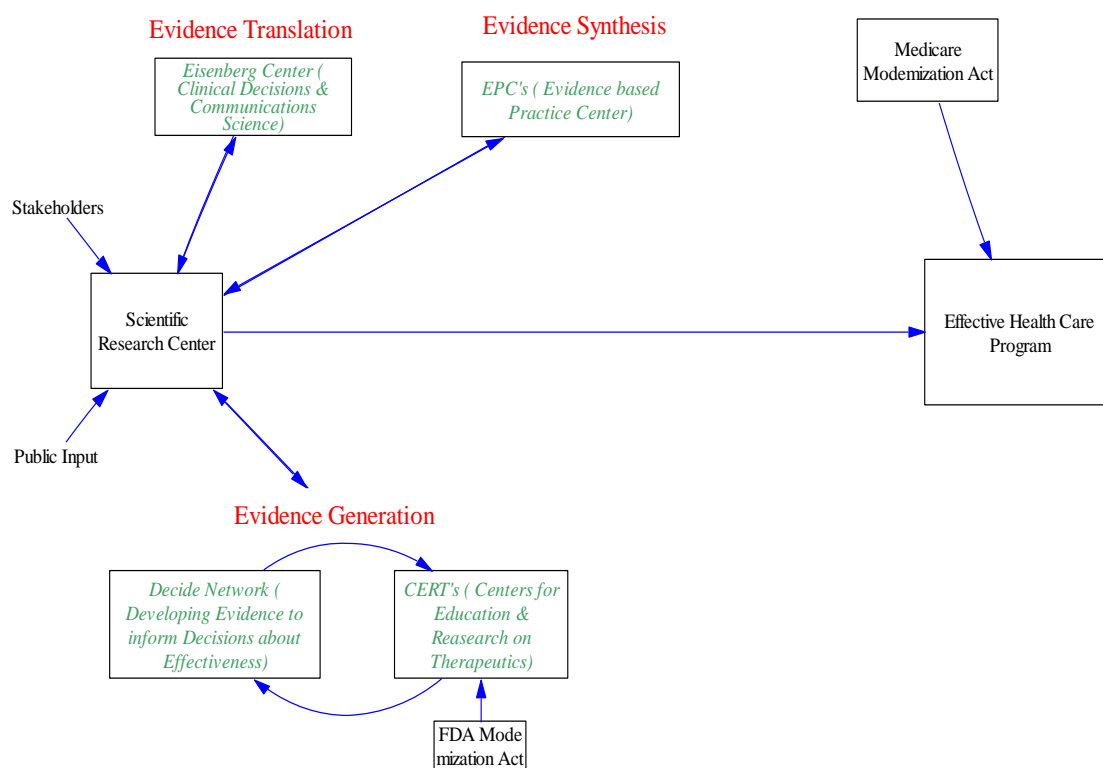


Figure 4.2: Role of Research Centers in Health Care System

The above diagram [9] explains how scientific research influences the effectiveness of the health care system. The Effective Health care program was created in response to the Medicare Modernization Act, and the scientific research center acts via stakeholders and public input. Scientific Research Center is also influenced by other 3 departments: Evidence Generation, Evidence Translation, and Evidence Synthesis [10, 11].

Evidence Generation: It follows a network called DEcIDE Network (Developing Evidence to Inform Decisions about Effectiveness) Network. Its main purpose is to develop valid scientific evidence about the outcomes, clinical effectiveness, comparative effectiveness, safety, and appropriateness of health care items and services.

Evidence Translation: Its purpose is to perform a systematic and comprehensive literature search, uniform extraction and tabulation of data from studies, qualitative and quantitative synthesis of data, and critical appraisal of studies to identify factors that may lead to biased results.

4.2 Federal & State Funding, and Insurance Coverage

The general division of the burden of paying for healthcare costs is shown in Figure 3, where the responsibility of paying for healthcare is either fully borne by the federal government, employers, or uninsured individuals, or is partially borne by the government through the new subsidized public option.

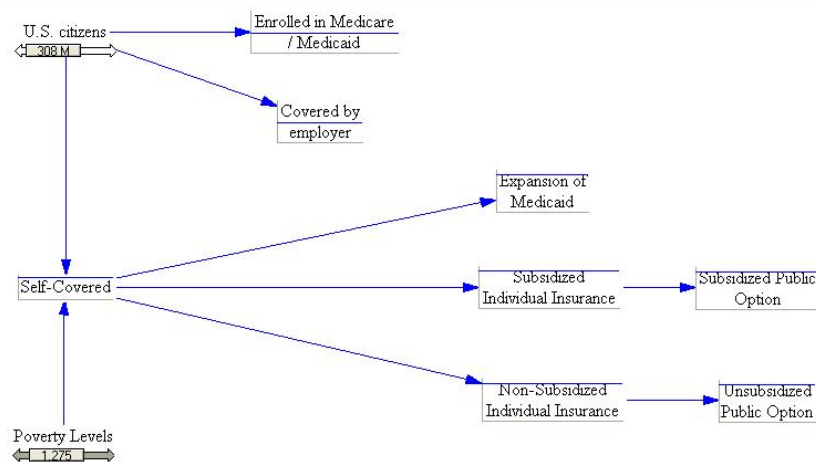


Figure 4.3: Role of Federal and State Governments in the Insurance Coverage

4.3 Medical Billing Process

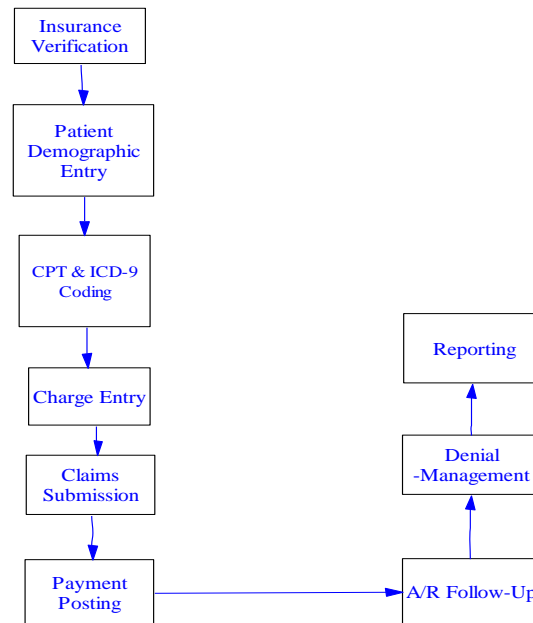


Figure 4.4: Medical Billing Process

Medical billing is another important department of Health care system. Medical records and updates of patients are maintained in this department. The payment is done by patient to the hospital by a third party called “Insurance agency”. The medical claim is checked thoroughly by the agent. The entire process is shown above in 9 different steps.

Insurance verification: Once the hospital gets the medical claims from the patient, they forward that to the medical billing team through a scanned document or through courier. Then the billing department uploads the scanned document into secure FTP server (File Transfer Protocol). It’s a standard network protocol used to manipulate the files, so that it makes the experts to access files easily.

Checking medical claims: Once the files are uploaded in FTP server, the billing team starts checking the files and notifies to the hospital authorities. If any document is missing and asks them to re-scan the entire document including the missing files.

Medical Coding: This is the important step in the billing process. The medical billing teams use a standard coding technique named “CPT” (Current Procedural Terminology) and “ICD-9” (International Classification of Diseases, Ninth edition) to fix the procedure and to diagnoses codes to each patient. The “Level of service” determines the associated 5-digit ‘procedure code’ and the diagnoses code is based on the medical diagnosis made by the doctor.

Charge Creation: Once the coding part is verified, the billing team creates a medical claim based on billing rules and then it is forwarded to audit team where a thorough checking is performed in different levels. Recognition of any incorrect or missing documents at this level is rejected.

Medical Claims Transmission: The final Medical claims are filled up before they are electronically sent to the claims transmission department with all required documents and information.

Claims Submission to Insurance Agencies: The audit team finally prints the medical claims and sends them to the concerned insurance agency or governmental agencies for the ultimate settlement.

Follow-up and Settlement: This is the final stage where the experts in medical billing department follow up consistently with the insurers until the final settlement is done.

4.4 Request Management System

RMS is an IT request management system built for Healthcare organizations. It helps in manage and improving the Business processes. RMS integrates the business process involved in managing the IT requests at a hospital. RMS uses standard XML based process definition in order to handle different types of requests using custom request templates.

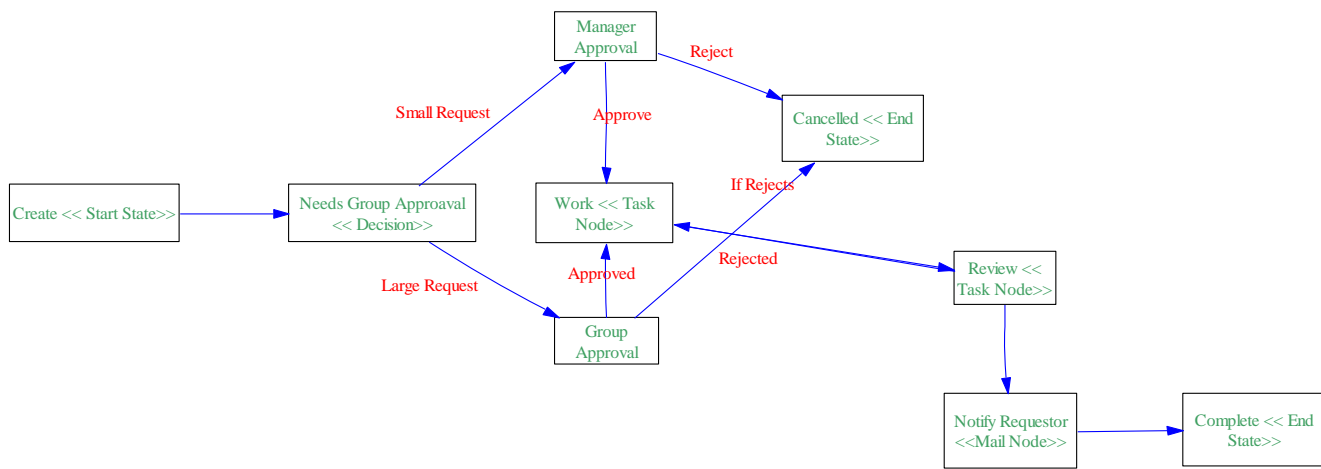


Figure 4.5: Request Management System

The process flows are represented in flowcharts and states the process steps in detail. It also allows the managers, workers and requestors to track the current status of the process flow. RMS collects all data from IT request and provides detailed reports for monitoring and managing requests. RMS is an excellent tool which helps in collaboration between Healthcare IT department and other operations within Healthcare organization.

4.5 Generalized Health Care System with Flow rates

The model of the generalized healthcare system was made quantitative by reformulating the model with flow and flow rates.

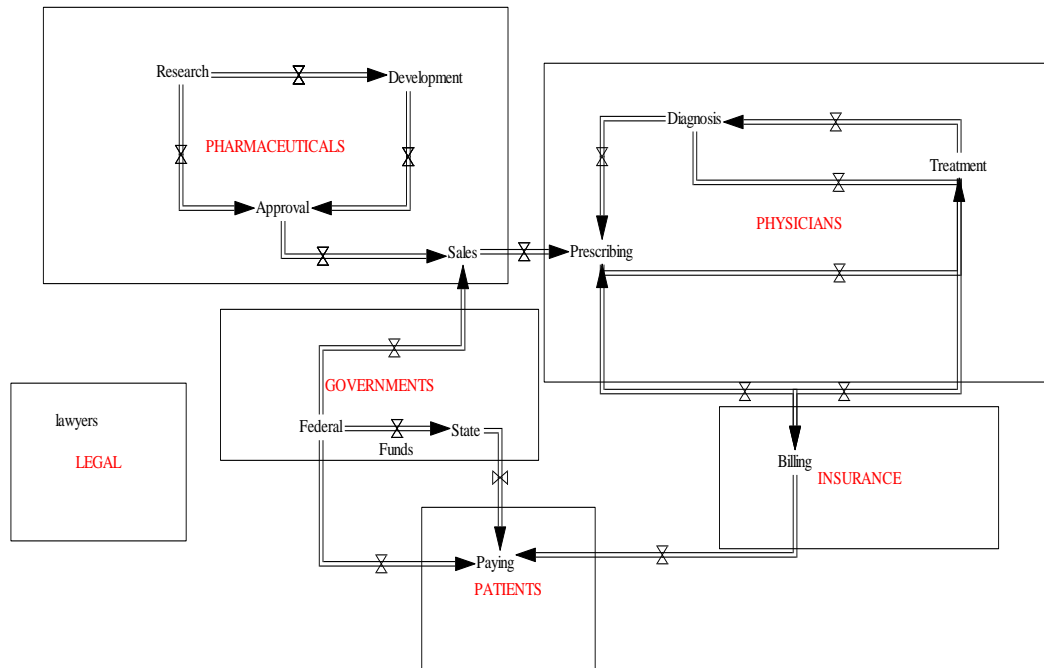


Figure 4.6: Generalized Health Care System including flow rates using Vensim

This model actually joins sectors of the healthcare system that are characterized by different measures. For example, the Pharmaceutical sector is best quantified by Number of Drugs in the Research, Development, or Sales stage. The Physicians sector is probably best quantified by Number of Patients, either being Diagnosed, Treated, or for whom drugs are being Prescribed. The Government, Insurance and Patient sectors are best quantified in terms of Dollars being spent at the Federal or State government level, being Billed by the Insurance companies, or being paid by the Patients. Standing aside is the Legal sector, which has not been incorporated into the model as yet.

Chapter 5: Review of the History of Soft Systems Modeling

The question “What is a system?” can be asked and answered differently, but the fact that the question *refers* to a *whole* -- called a system -- remains. While formal engineering design and modeling languages describe system parts, the practice of systems engineering results when there is reference to holistic systems, often via *self-reference*. Self-reference creates the possibility of circular, paradoxical reasoning where multiple outcomes can occur. Conceptual structuring by abstraction levels with complementarity clarifies paradoxes without resort to strict hierarchical decomposition that nullifies complexity. Gödel’s Incompleteness and Inconsistency Theorems prove truths about formal languages that have the ability of *self-reference*, elucidating analogous relations among: informal natural language statements about systems, systems, and formal languages that describe systems. The goal of this work is to foster cognizance in system descriptions.

Complicated systems have a great number of mechanical or deterministic parts, which despite the possibly great effort needed for their deciphering, are nonetheless fully understandable by formal means, such as formal logic or deterministic mathematical formulations. The configurations of a complicated system are enumerable, even if not all enumerations are available with current computational abilities. For example, a large collection of ideal pre-arranged billiard balls may be struck by a cue ball, invoking the question: “In what direction and at what speed will the billiard balls propagate?” or, “What is the resultant vector of the billiard ball placed at the very back?” Any collective ‘emergent’ behavior that complicated systems evince can in fact be predicted by accounting for the behavior of the constituent parts. The most useful and widely applied tools of Systems Engineering arguably remain methods that

effectively decompose and mechanize complex systems so as to render their models as merely complicated.

Complexity in mathematical formalizations can be perceived through the presence of intriguing members of mathematics, including: 1, Random numbers, 2, Transcendental numbers, and 3, Imaginary numbers. Random numbers cannot be produced by established algorithms, and are thus only truly available via non-determinable generators, such as the quantum nature of reality. Transcendental numbers, like irrational numbers, cannot be described succinctly as the quotient of two integers, and never exhibit patterns. The transcendental number π is essential to the description of the completeness of a circle, and the transcendental number e is the only consistent base for a natural logarithm. The fundamental nature of imaginary numbers remains a mystery, despite the laconic definition, $i = \sqrt{-1}$. Imaginary numbers are constructively employed in complex numbers, and provide a bridge between phases in wave mechanics and real objects. In so far as any system requires characterization by these three types of numbers, the system can be classified as complex. As a practical matter, systems in this universe, with the examples of humans and other phenomena in the natural world, are complex systems.

Emergence is a fascinating and vexing feature of complex systems, giving rise to system properties that are not present in their constituent parts [Warfield, 2002]. Emergence is approached in this paper via concept structuring.

Complementarity, a dualistic principle that is a touchstone of complexity and found in the description and mathematics of quantum mechanics, describes the relation of emergent

qualitative attributes [Smith and Bahill, 2010], as contrasted with incommensurable concrete and logical parts of a system. The principle of complementarity structures natural language descriptions of systems and clarifies discussions in systems engineering and architecting [Smith, 2008]. Complementarity diagrams show qualitative attributes as distinct, yet coexisting with, logical elements, as shown in Figure 1.

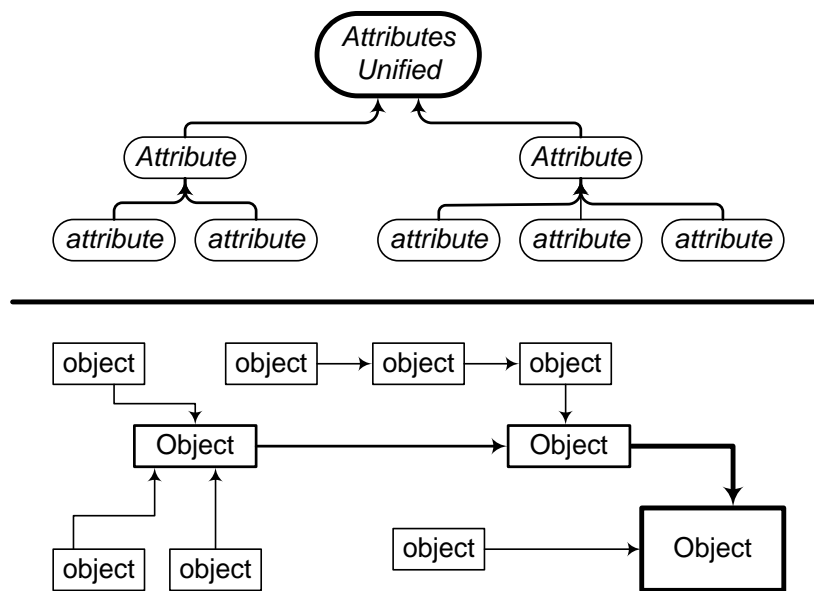


Figure 5.1: Complementary sides of a system

Complementarity in nature gives rise to an infinite interplay between irreconcilably different aspects of reality. Complementarity diagrams reduce the aspects of naturally complementary systems to a perceivable and distinct two-sided description.

Levels of abstraction [Bahill, Szidarovszky, Botta and Smith, 2008] are a primary construct for the descriptions of systems that exhibit encompassing layers. Figure 2 illustrates encompassing levels of abstraction.

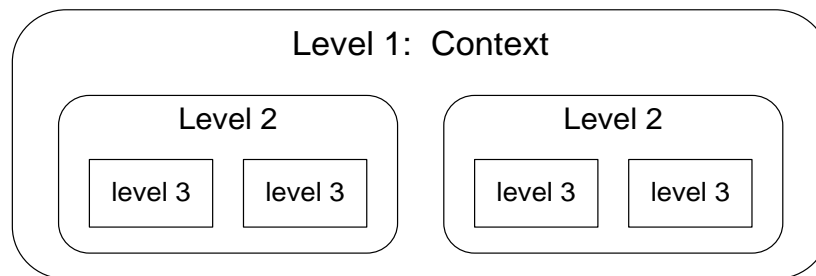


Figure 5.2: Encompassing abstraction levels

Note that the encompassing abstraction is shown at higher levels, but to be fair, the numerous details observable at lower levels could alternatively be shown as encompassing the more vacuous abstract levels. Alternatively, if the upper levels have fuller and greater detail, they are not abstract. Discussions in this paper are facilitated by the depiction of complementarity at different levels of abstraction as shown in Figure 3.

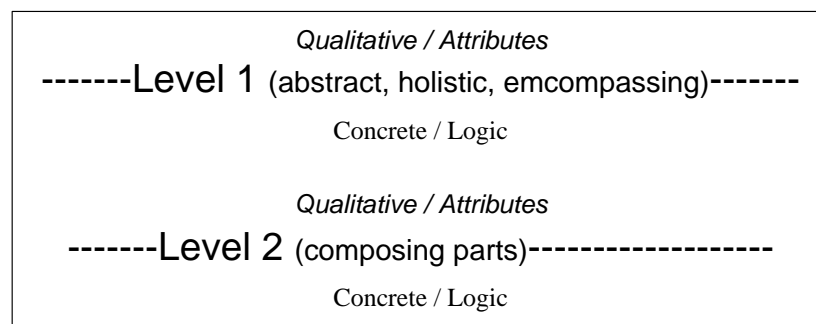


Figure 5.3: Levels with complementary aspects at different levels of abstraction

At any particular level, the attribute side of the complementarity dual is characterized by the qualities apparent at that level, while the logical side is the collection of concrete logical elements and their interfaces. The influences and effects of qualitative or logical elements on other qualitative or logical elements identified on other levels are diagrammed in Figure 4.

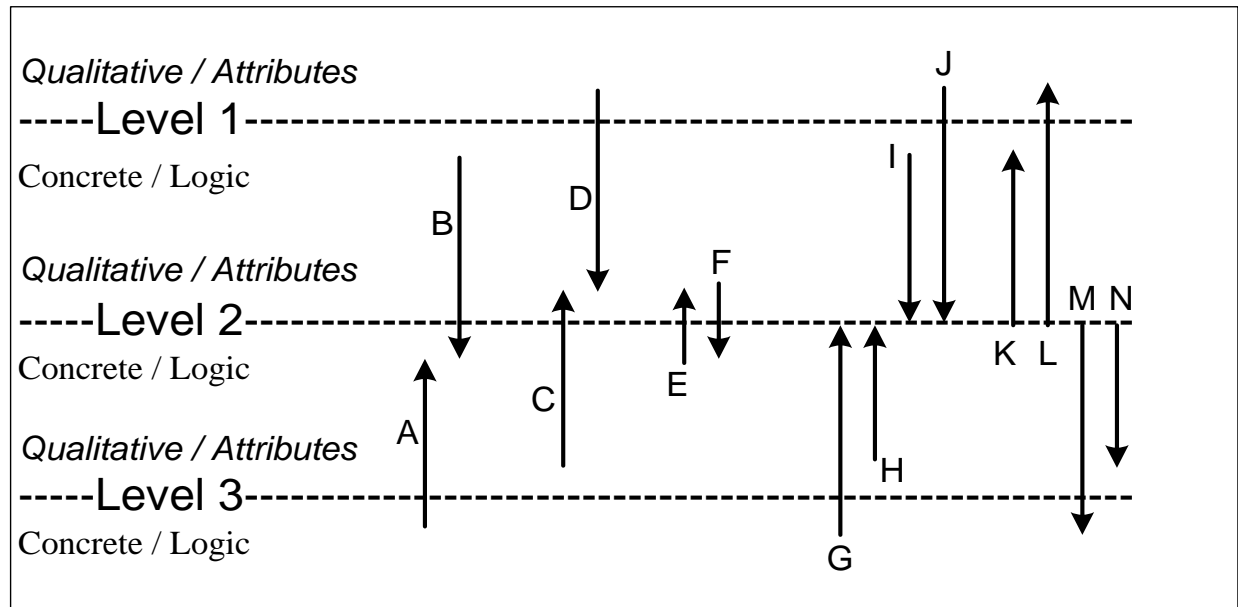


Figure 5.4: Effects of complementary levels

The effects, by letter label, can be described as follows:

Concrete / Logic relations between adjacent levels:

A, Lower-level concrete/logic elements composing upper-level concrete/logic elements

B, Upper-level concrete/logic elements decomposed into lower-level concrete/logic elements

Qualitative/Attribute relations between adjacent levels:

C, Qualitative attributes are holistically and abstractly combined at the next higher level.

D, Holistically qualitative attribute provides the context (scope) for attribute decomposition.

Complementary relations on same level:

E, Logical elements create holistic attributes at the same level. (Example: Reliability calculated)

F, Attributes set global scope of possibilities and imbue logical elements at same level.
(Example: Reliability as a mandated quality)

Additional relations available:

G, Lower-level logic contributing to whole upper level

H, Lower-level attributes contributing to whole upper level

I, Upper-level logic encompassing whole lower level

J, Upper-level attributes encompassing whole lower level

K, Whole level influencing upper-level logic

L, Whole level influencing upper-level attributes

M, Whole level encompassing lower-level logic

N, Whole level encompassing lower-level attributes

Extensive use of this framework has not yet been demonstrated.

Mathematical logic, in its own idealized world, could limit the number of attributes to only two: True and False, which are absolute attributes arising from logic. Such a view leads to the Mathematician's Credo [Hofstadter, 2007, pp. 120-122]:

- (1) X is True *because* there is a proof of X. – consistency of logical system
- (2) X is True *and so* there is a proof of X. – completeness of logical system

The first statement speaks to the consistency of the logical system – because an inconsistent logical system could contain both the proof and counterproof of X. A related statement is: X is False *and so* there is no proof of X.

The second statement speaks to the completeness of a logical system; that is, the logical system does contain a proof for all true Xs, and no proof for false Xs. The second statement can be re-phrased as: X is False *because* there is no proof of X.

This perfect alignment of strict bi-directional relations creates tightly-bound dyads between truth and logic, and is illustrated in Figure 5.

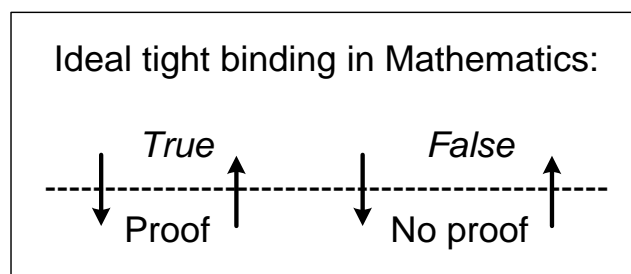


Figure 5.5: Idealized, perfect correspondence in mathematics

Historically, the effort to uncover this perfect alignment between truth and presence of proof, and between falsity and the absence of proof, was memorialized in the movement to axiomatize all of mathematics, beginning with the axiomatization of arithmetic. The climax of this movement was the appearance of **Principia Mathematica**, published 1910-1913 as the *magnum opus* of Bertrand Russell and Alfred North Whitehead. Principia Mathematica sought to implement this perfect alignment between truth and logic, with seed axioms producing all true theorems, and of course, no untruths [Hofstadter, 2007, p. 129].

Kurt Gödel (1906-1978), Austrian logician, mathematician and philosopher, ultimately proved that such a tight binding is not possible. Gödel's theorem utilizes the conceptual framework of complementary mathematical languages at different levels of abstraction, as will be illustrated. "The utterly shocking import of Gödel's theorem ... is that the mighty edifice of mathematics is ultimately built on sand, because the nexus between proof and truth is demonstrably shaky. The problem that Gödel uncovered is that in mathematics, and in fact in almost all formal systems of reasoning, statements can be true yet unprovable – not just unproved, but unprovable, even in principle" [Davies, 2007, p. vi]. A seemingly tight binding between qualitative attributes and logical proofs in mathematics is made more complex by reference in mathematics to many more qualities and attributes besides True and False, for example, strength, soundness, adequacy, and well-formedness. Logical mathematics cannot advance without sophisticated perception of a plethora of qualitative attributes, as memorialized by Leibniz:

"Sans les mathématiques on ne pénètre point au fond de la philosophie.

Sans la philosophie on ne pénètre point au fond des mathématiques.

Sans les deux on ne pénètre au fond de rien." — Leibniz

(Without mathematics we cannot penetrate deeply into philosophy.

Without philosophy we cannot penetrate deeply into mathematics.

Without both we cannot penetrate deeply into anything.)

1686 *Discours de Métaphysique* [Montgomery, 1962]

Expressive systems employ complementary *semantic* and syntactic sides.

Specifically, the system must have quality-expressive *semantics*, and must be logically expressive in syntactic terms as illustrated in Figure 6.

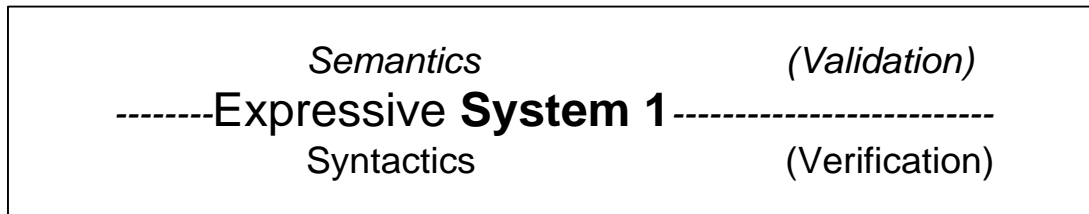


Figure 5.6: Semantics and Syntactics in an expressive system

A parallel can be drawn to the *validation* of a system – in that the system holistically satisfies the totality of customer needs – and the verification of specific logical requirements.

Self-Reference can only occur where a higher level system encompasses a lower-level system. Self-Reference is possible when syntactic terms in a lower-level expressive system typographically refer to syntactic and semantic terms that only properly exist in a more abstract, encompassing and higher-level expressive system. Reference to any holistic quality of a lower-level system, from within the lower level system, can only truly occur with a reference to the holistic total quality emergent and fully sensed only at a higher level, as illustrated in Figure 7.

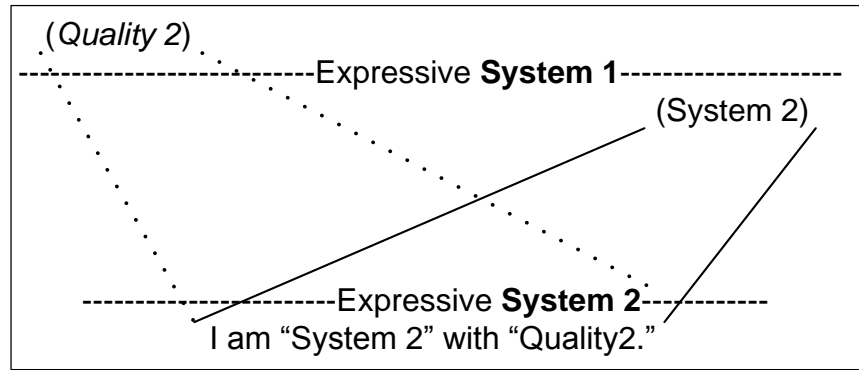


Figure 5.7: Self-Reference

Some examples of self-reference within a systems engineering enterprise include: 1, A requirements database for an industry program contains the requirement: “This program shall remain within schedule.”, and, 2, A Systems Modeling Language (SysML) context block within a diagram referring to the “entire design process.”

Self-Reference is produced often, effortlessly and almost without notice in the human mind, and can be easily written into systems engineering documents. Cognizance of the occurrence of self-reference is vital to the production of properly organized systems engineering design materials. For example, unnoticed self-reference in a systemic decomposition can quickly and erroneously insert, in lower levels of the decomposition, elements of the design that simply do not exist at lower-levels of the decomposition – for example, high-level attributes. Such errors often result because the human mind – even when supposedly focused only on lower decomposition levels, has easy access to the total system, and quickly generates terms that refer to the total system.

Self-referring expressions imply the integration of an entire system. Systems engineering vaunts the practice of integrating systems; consequently, self-reference to the totality of a system is typical within the many languages of systems engineering. As an example: Systems

engineering processes that shape entire systems are often referenced within systems engineering documents. Therefore, this question can be asked: How can integrative efforts be improved by recognition of the concept and practice of self-reference within natural and system-theoretic languages?

Complexity exists wherever a self-reference has been made. In addition to the previously noted three (3) succinct mathematical indices of complexity, self-reference also indicates the presence of a complex situation. Note the concept of self-reference was only reachable in this introductory section after developing two concepts which are complex – 1, complementarity, and 2, levels of abstraction which imply emergence.

Self-reference gives rise to the possibility of infinite self-reference in a series of loops. “In short, there are surprising new structures that looping [self-reference] gives rise to that constitute a new level of reality that could *in principle* be deduced from the basic loop and its detailed properties, but that *in practice* have a different kind of “life of their own” and that demand – at least when it comes to extremely finite, simplicity-seeking, pattern-loving creatures like us – a new vocabulary and a new level of description that transcend the basic level of out of which they emerge” [Hofstadter, 2007, p. 71].

Self-reference is arguably the beginning of self-awareness. In lieu of a definition and discussion of self-awareness, the description of a Universal Turing Machine, which can observe and model itself, can be examined:

“Inspired by Gödel’s mapping of *PM* [Principia Mathematica] into itself, Alan Turing realized that the critical threshold for this kind of computational universality comes at exactly that point where a machine is flexible enough to read and correctly interpret a set of data that describe its own structure. At this

crucial juncture, a machine can, in principle, explicitly watch how it does any particular task, step by step. Turing realized that a machine that has this critical level of flexibility can imitate any other machine, no matter how complex the latter is. Universality is as far as you can go!” [Hofstadter, 2007, p. 242].

Fractals, vivid illustrations of mathematical complexity, are generated by self-reference. For example, the Mandelbrot Set, is generated by the iterative application of the mathematical feedback loop:

$$z_{n+1} = z_n^2 + c$$

A complex number, c , is in the Mandelbrot set if, when starting with $z_0 = 0$ and applying the iteration repeatedly, the [absolute value](#) of z_n never exceeds a certain number that depends on c . When computed and graphed on a complex plane, the Mandelbrot set is seen to have an elaborate boundary which does not simplify at any given magnification. This qualifies the boundary as a fractal – a touchstone of complexity.

As a prelude to outlining Gödel’s Theorems, this paper now turns to the explanation of paradoxes via the application of the previously illustrated concepts of complementarity, levels, and self-reference. The insights gained are then applied to the current taxonomy of systems engineering methods.

Chapter 6: Zachman Framework

The subject of information systems architecture is beginning to receive considerable attention. The increased scope of design and levels of complexity of information systems implementation are forcing the use of some logical construct for defining and controlling the interfaces and the integration of all of the components of the system. Current technology is rapidly removing both conceptual and financial constraints. It is not hard to speculate about, if not realize, very large, very complex systems implementations, extending in scope and complexity to encompass an entire enterprise. One can readily delineate the merits of the large, complex enterprise-oriented approaches. Such systems allow flexibility in managing business changes and coherency in the management of business resources. However there also is merit in the more traditional, smaller, suboptimal systems design approach. Such systems are relatively economical, quickly implemented, and easier to design and manage.

On the assumption that an understanding of information systems architecture is important to the development of a disciplined approach, the question that naturally arises is “What, in fact, is information systems architecture?” Among the proponent of information systems architecture, there seems to be little consistency in concepts or in specifications of “architecture”, to the extent that the words “information systems architecture” is already losing their meaning. The commitment associated with vested interests almost demands a neutral, unbiased, independent source as a prerequisite for any acceptable work in this area. In any event, it will be necessary to develop some kind of framework for rationalizing the various architectural concepts and specifications in order to provide for clarity of professional communication, to allow for improving and integrating development methodologies and tools, and to establish credibility and confidence in the investment of systems

resources. Information systems architecture is related to strategy, both information strategy and business strategy, this paper deliberately limits itself to architecture and should not be constructed as presenting a strategic planning methodology. The development of a business strategy and its linkage to information systems strategies, which ultimately manifest themselves in architectural expression, is an important subject to pursue, but it is independent of the subject of this work, which is defining a framework for information systems architecture.

In order to understand more about the importance of a frame work for an Information system, let's choose a, example of constructing a building. Earlier to build a design, people used a different basic means of architectural concept called "Bubble Chart". The first architectural deliverable created by the architect is a conceptual representation which describes the size, shape, spatial relationships and basic intent of the final structure. To understand better, here is an example of a conversation between the owner and architect

"I'd like to build a building"

"What kind of building do you have in mind?"

"Do you plan to sleep in it? Eat in it? Work in it?"

"Well. I'd like to sleep in it."

"Oh, you want to build a house?"

"Yes, I'd like a house."

"How large a house do you have in mind?"

"Well, my lot size is 100 feet by 300 feet."

“Then you want a house about 50 feet by 100 feet?”

“Yes, that’s about right.”

“How many bedrooms do you need?”

“Well, I have two children, so I’d like three bedrooms.”

Looking at the above conversation, it is very clearly stated that the architect from his/her way of putting questions trying to understand the whole plan from owner’s perspective. From the way the owner explaining his idea of the building to get constructed, architect draws the building in his perspective and then tries to match with the owner’s perspective. Then, once the architect shows the building drawing to the owner and gets approval for the further financial expenditure on building the house, architect starts drawing the building design in a technical way with all specifications and details. Then the architect approaches the Contractor and draws his plans to produce the contractor’s plans representing the builder’s perspective. Such plans are prepared because of complex engineering involved in making the components required for the construction. The Contractor considers all constraints like technology constraints, natural and environmental constraints etc. [12]

Shop plans which are described as short of the final structure itself, and are prepared by the subcontractors. These subcontractors may not come into picture when we plan for constructing a building but they are equally important people in reaching the final goal. Seven thousand years of human history would establish that the key to complexity and change is Architecture. If it gets so complex that you can’t remember everything all at the same time, you have to write it down. Once the Contractor begins to manufacture the components, the inner or

sub components are taken care by the shop plans who actually fabricate and assembly the sub sections of the components. [12]

Once the components are assembled and Contractor finishes the work as they planned accordingly from the architect's design, then the final representation of the physical building itself. In summary, there is a set of "architectural" representations that are produced during the process of constructing a building.

Representation	Nature/Purpose
Bubble Charts	Basic concepts for building Gross sizing, shape, spatial relationships Architectural/owner mutual understanding Initiate project
Architect's drawings	Final building as seen by the owner Floor plans, cutaways, pictures Architect/owner agreement on building Establish contract
Architect's plans	Final building as seen by the designer Translation of owner's view into a product Detailed drawings – 16 categories Basis for negotiation with general contractor
Contractor's plans	Final building as seen by the builder Architect's plans constrained by laws of nature and available technology

	“How to build it” description Directs construction activities.
Shop Plans	Subcontractor’s design of a part/section Detailed stand-alone model Specification of what is to be constructed Pattern
Building	Physical building

Once we have listed the perspectives of the owner, architect and the contractor, then descriptions

Zachman Framework is an Enterprise Architecture introduced in 1987 by John Zachman and extended by Sowa in 1992. This Framework helps in modifying an enterprise into a logical structure for classifying and organizing the descriptive representations of an enterprise that are significant to the management and as well as the development of the enterprise’s systems. The units of the framework can also be understood as organization scheme for all kinds of systems and have therefore become widely recognized during the last years. Since this Framework is independent from tools or methodologies, any methodology can be mapped against it to understand about the system. [13]

The Zachman Framework is a powerful answer to these questions: by providing a global view of the multiple aspects of enterprise architecture, it offers a navigation tool that acts both as starter and a compass for enterprise modelers. It provides a context in which Business and IT architects can build a flexible, consistent information system, according to the strategy of their enterprise. [14]

Zachman Framework appears as a matrix with 30 cells, each of them focusing on particular dimension and perspective of the enterprise.

	Data (What)	Function (How)	Network (Where)	People (Who)	Time (When)	Motiva- tion (Why)
Objectives / Scope	List of things important to the enterprise	List of processes the enterprise performs	List of locations where the enterprise operates	List of organizational units	List of business events / cycles	List of business goals / strategies
Business Owner's View	Entity relationship diagram (including m:m, n-ary, attributed relationships)	Business process model (physical data flow diagram)	Logistics network (nodes and links)	Organization chart, with roles; skill sets; security issues.	Business master schedule	Business rules
Architect's View	Data model (converged entities, fully normalized)	Essential Data flow diagram; application architecture	Distributed system architecture	Human interaction architecture (roles, data, access); Security requirements	Dependency diagram, entity life history (process structure)	Business rule model
Technology Designer's View	Data architecture (tables and columns); map to legacy data	System design: structure chart, pseudo- code	System architecture (hardware, software types)	User interface (how the system will behave); security design	"Control flow" diagram (control structure)	Business rule design
Builder's View	Data design (denormal- ized), physical storage design	Detailed Program Design	Network architecture	Screens, security architecture (who can see what?)	Timing definitions	Rule specification in program logic
Function- ing system	(Working systems)					
	Converted data	Executable programs	Communica- tions facilities	Trained people, using the system	Business events	Enforced rules

Figure 6.1: General Enterprise Zachman Framework Model [14]

The rows represent the points of view of different players in the systems development process, while columns represent different aspects of the process.

- a. Scope: Definition of the enterprise's direction and business purpose. This is necessary to establish the context for any system development effort.
- b. Owner's view: This defines the nature of the business, including its structure, functions, organization, and so forth.
- c. Architect's view: This defines the business owner's view in more rigorous information terms. It describes those things about which the organization wishes to collect and maintain information, and begins to describe that information.
- d. Designer's view: This describes how technology may be used to address the information processing needs identified in the previous rows. Here all kinds of languages are selected and program structures are defined, user interfaces are described, and so forth.
- e. Builder's view: Here a particular language is chosen, and the program listings, database specifications, networks, and so forth are all produced.
- f. Functioning system: Finally, a system is implemented and made part of an organization.

The columns in the Zachman framework represent different areas of interest for each perspective. The columns describe the dimensions of the systems development effort.

- a. Data: Each of the rows in this column address understanding of and dealing with an enterprise's data.
- b. Function: The rows in the function column describe the process of translating the mission of the enterprise into successively more detailed definitions of its operations.
- c. Network: This column is concerned with the geographical distribution of the enterprise's activities. At the strategic level, this is simply a listing of the places where the enterprise does business.

- d. People: The fourth column describes who is involved in the business and in the introduction of new technology.
- e. Time: The fifth column describes the effects of time on the enterprise.
- f. Motivation: This is concerned with the translation of business goals and strategies into specific ends and means. [14]

6.1 Analysis Process

Where other methods look at analysis as a single process, the Zachman Framework makes an important distinction. As analysts who view the world in terms of information, it is hard sometimes to realize that not everyone sees things that way. It is illuminating to be forced to recognize that the terms of reference for the user community are not the same as ours.

6.2 Advantages of the Zachman Framework

- Improving professional communications within the information systems community.
- Understanding the reasons for and risks of not developing any one architectural representation.
- Placing a wide variety of tools and/or methodologies in relation to one another.
- Developing improved approaches (including methodologies and tools) to produce each of the architectural representations, as well as possibly rethinking the nature of the classic “application development process” as we know it today

6.3 Disadvantages of the Zachman Framework

- It can lead to a documentation-heavy approach (although this does not have to be the case). There are 36 cells in Figure 1, each of which could be supported by one or more models.

- It can lead to a process-heavy approach to development – you can instantly see the opportunity to define a collection of rigorous processes to support the Zachman Framework.
- The Zachman Framework isn't well accepted within the development community and few developers even seem to have even heard about it.
- The Zachman Framework seems to promote a top-down approach to development. When people first read about the Zachman Framework, they tend to think that it implies a top-down approach where you start with the models in row 1, then work on row 2 models, and so on. This doesn't have to be the case, you can in fact start in any cell and then iterate from there.
- The Zachman Framework appears to be biased towards traditional, data-centric techniques (thus explaining its popularity within the data community). [15]

Chapter 7: System Dynamics on Zachman Framework

7.1 Integration of System Dynamics on Health Care System

Hospital trusts are complex systems in which several parts interact over time. Because of lots of ongoing activities in several departments internally and tremendous interconnectedness between them and the influences arising from the external environment, measuring and managing hospitals performance is especially challenging. System's performance is completely depends on the interactions and co-ordinations among the parts of the system. Complex Health Care system usually has lot of sub systems which can really affect the whole system with their actions. In a complex area like Health Care there could also be affects from the elements outside the boundary of hospitals. So, it's very important to take into consideration all these elements in order to maintain the whole system. To measure and manage the design of a system performance in Hospital Trusts, we need to take into account the interests of stakeholders like patients, health care professionals, administrative and managerial staff, purchasers and central government. Each stakeholder has their own perspective in the way to judge the performance of the Hospital Trust. So, considering all different measures on judging the performance of Hospital Trusts helps to find the best outcome in achieving the best performance.

Financial and Non-financial performance measures are usually poorly integrated. The interconnections between the performance measures across performance dimensions are very difficult to establish. Cause and establish relationships, specific targets or standards against goals attainment can be measured are frequently are absent. To overcome all these weaknesses in the system we need to obtain a tool that can analyze the whole system. It is widely accepted that the performance measurements of a system helps the decision-makers to understand about the system and to achieve the objective of the organization. Initiating and developing a performance

measurement and management system helps to identify the strategic objectives or orientation of the organization and the factors that are critical to its success and also to promote effective delivery of high quality services. However, such objective is too broad for managers to evaluate how well it is being achieved by the Hospital Trusts. Generating the proper set of performance measures to analyze the entire system is not an easy task. So, it is always recommended to a repetitive thorough check on the system elements in fixed time period. The use of both qualitative and quantitative System Dynamic models helps in understanding the system

7.2 Compare a narrative (oral or written) with Visualization

As a part of our analysis on the complex system decision making techniques, diagnosing patients and doctors interview gives us a better scope in emphasizing on the internal loops, impacts and on influences among different entities. During the interview we can identify certain terms which has a great impact on the entire issue. It's important to understand the scenario not only by identifying the entities or terms but also by finding the influences among entities. This experiment helped us to understand the scenario in a much better way where we also identified the sequence of entities to certain extent.

Doctors interface with patients through three principal means, interviews, observation and testing. While testing based on scientific principles is highly useful, first hand information based on the patient's perspective is often available on through interviews with the patient, and, for this reason, medical interviewing has been described and developed by Coulehan and Block [16], Enelow, Forde and Brummel-Smith [17], Newell [18], Berstein and Berstein [19], and Froelich and Bishop [20]. Interviewing and observation is of course balance with quantitative

evidence-based medical practice [21] [22]. Complementary balance in medical decision making is described by Seedhouse [23] in “*Values-Based Decision-Making for the Caring Professions.*”

An abridged and edited example based on a diagnostic interview from Newell [18, pp. 89-98] follows:

Doctor: Please tell me in general terms the issues that brought you here today.

Patient: Well, it's about my inhaler. They say I use it far too much.

Doctor: What do you think yourself?

Patient: I know the amount that I'm using it has increased just of late.

Doctor: It sounds like controlling its use is a problem.

Patient: Definitely. Every time I go anywhere, I have to have it with me just in case.

Doctor: What symptoms do you get before attacks?

Patient: Tightness in my chest, wheezing, and my heart starts thumping ...

Doctor: And that makes you say certain things to yourself?

Patient: Yes, it's very frightening.

Doctor: And then what do you do?

Patient: I take maybe three puffs, and wait and see if the breathlessness will pass. Doctor:
When does this happen?

Patient: Particularly when I am driving and the kids are in the car.

Doctor: You say you have had asthma for 20 years ...

Patient: Yes.

Doctor: Our approach will be for you to relax.

Patient: Ok ...

Doctor: ... and when the attacks come on, delay reaching for the inhaler for just a few seconds, to give you a better idea if you can cope with the anxiety that accompanies an attack.

Patient: Ok.

Doctor: This goes along with helping you reach some of the goals you are setting for yourself in life.

Patient: Yes, I want to get to where I enjoy life.

Doctor: Gradually, you will lessen the frequency of puffs, and the need for prescriptions.

Patient: Sounds like a good plan.

7.3 System Dynamics on Diagnosis Interview

Systems Dynamics (SD) is a methodology use to define the influences and relations among many factors that contribute to directed effects and feedback loops. System Dynamics was developed by Forrester [24-25], and has been adopted widely as a visualization of complex systems. SD drawings can be used to visualize the many attributes pertinent to a medical diagnosis, as obtained in a medical interview. Additionally, the directed arcs of SD show attributes drew the attention of the interviewing doctor, and which causal attributes were deemed to contribute, either positively or negatively, to affect attributes.

The diagnostic interview quoted above can be shown with an SD diagram. As a first option, the SD diagram could show the temporal order of attributes as they arose in the interview, but this would produce a rather tangled diagram.

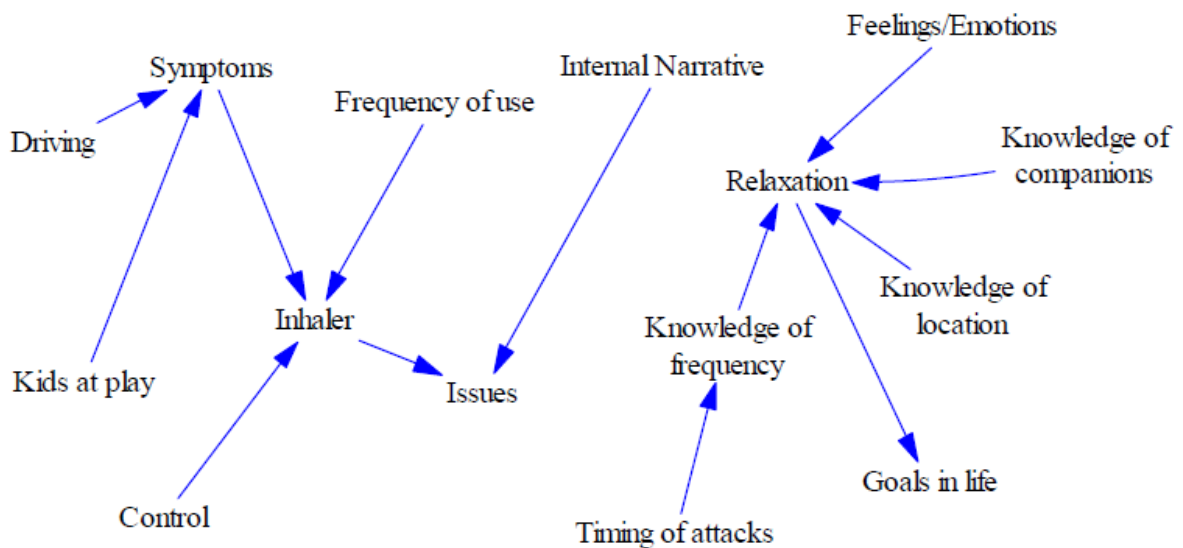


Figure 7.1: System Dynamics depiction of influences noted during interview

From the above System Dynamics model we can identify the important entities and their influences among each other. From the above SD diagram, physician can identify the reasons behind her present health condition. Her mental and physical relaxation has effects from the knowledge of frequency, location, and companions and from her feeling and emotions. By analyzing this SD diagram helps the physician to decide the way to treat the patient and may get succeeded in the diagnosis. But System Dynamic elements, within pure SD diagrams, have no guarantee of being arranged in an intuitive fashion that shows the classification of attributes according to their levels and aspects. The use of a Zachman Framework with an overlaying SD diagram accomplishes the purpose of clarifying the inherent differentiations and relative scales of all factors used within an SD diagram.

7.4 Zachman Framework

The Zachman Framework [26] was originally an enterprise modeling tool. It is essentially a 6x6 matrix which defines 6 levels relevant to any enterprise, as well as 6 aspects. The structuring provided by the Zachman Framework provides that attention is placed on all the relevant scales, as well as on all relevant aspects, of any situation under consideration. Any Zachman Framework should be calibrated so that all relevant scales occur within its boundaries. The circles in this particular depiction of the show the enterprise areas that were involved in the diagnosis of Anne Dodge, a patient whose particularly difficult story of finding health is narrated by Groopman. The number and variety of enterprise areas involved in one diagnosis is surprising, and indicates that medical professionals need an expanded awareness of the entire medical enterprise in order to serve patients.













		What	How	Where	Who	When	Why	
Wellbeing	Scope							Strategists
Community	Medical practice							Leaders
Business	System							Architects
Knowledge	Diagnosis							Doctor
Domains	Specialties							Specialists
Tests	Physiology							Technicians
Attendants	Operations							Nurses
		Examined	Process	Network	Patient	Timing	Motivation	

Figure 7.2: Zachman Framework for the Medical Enterprise

As we discussed about the rows and columns in the Zachman Framework section, the rows are the perspectives from each individual view of the context.

7.4.1 Rows or Perspectives

Wellbeing: It can also be called as Planner and illustrates the goals or objectives of the issue and the entities in this row portrait the perspectives of that view.

Cognitive: Looking at the current issue we are working with, this row gives a better understanding on the cognitive work that undergoes from different entities like patient, physician or surrounding of the patient.

Behavioral: This row represents the surrounding nature of the main entity which is patient. This row helps in illustrating the causes behind the entire scenario.

Physical: The title itself gives us the meaning and this row represents the physical reasons and timings of the sickness symptoms.

Treatment Aids: This section of rows gives a detailed view on kith and kin entities in the issue.

7.4.2 Columns or Interrogatives

Each perspective focuses attention on the same fundamental questions, then answers those questions from that viewpoint, creating different descriptive representations (i.e., models), which translate from higher to lower perspectives. The six categories of enterprise architecture components, and the underlying interrogatives that they answer, form the columns of the Zachman Framework and these are

- The data description — What
- The function description — How
- The Network description — Where
- The people description — Who

- The time description — When
- The motivation description — Why

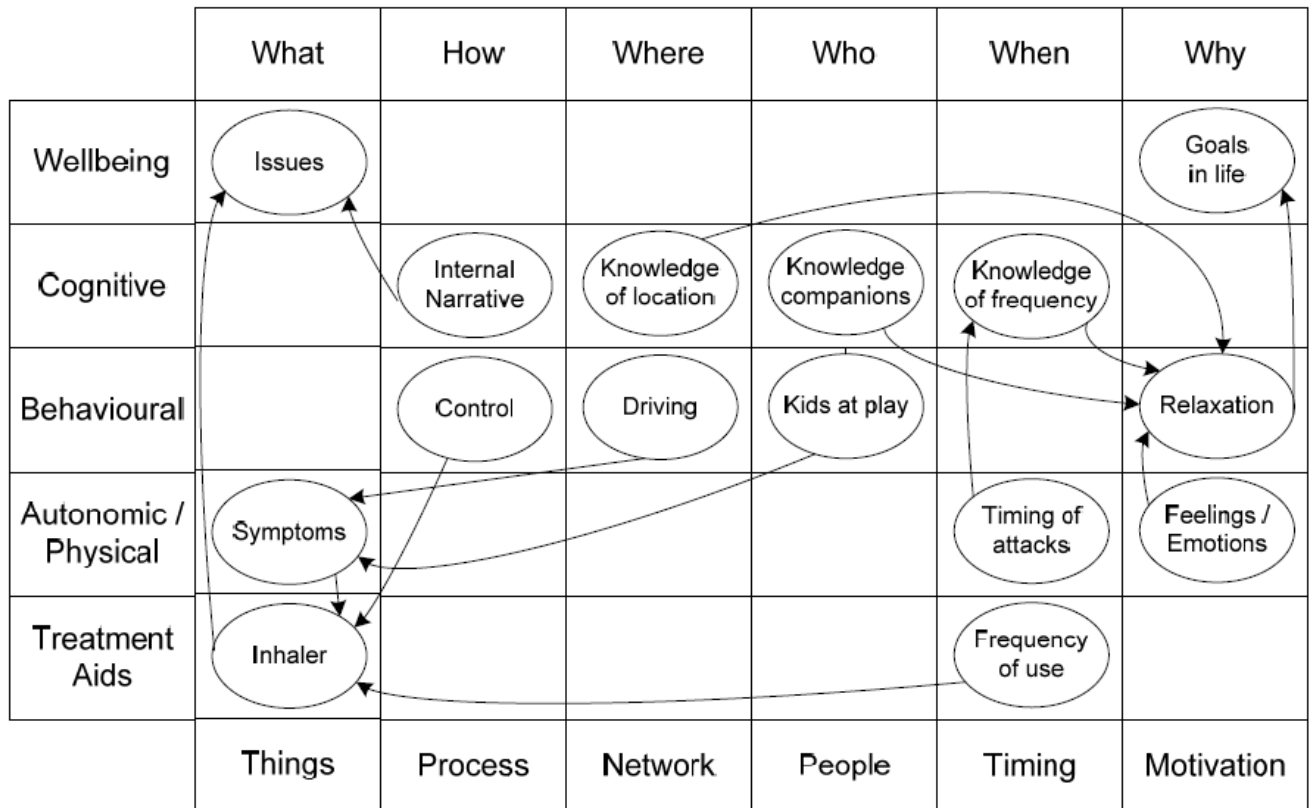


Figure7.3: Medical Interview shows with a System Dynamics overlaid on a Zachman Framework

The above diagram gives a structured and formal view on the scenario and from this it's easy to place the entities from system dynamics diagram into the cells in Zachman Framework and also we can indicate the influences among the entities. This way of representing the enterprise gives a better view and helps in avoiding the mistakes we do unknowingly. It has been a goal to avoid uncertain Attribute Substitution and from this concept of overlaying the SD model in Zachman Framework we can avoid that mistake to a large extent.

Chapter 8: Justifications (Comparison with other tools/techniques)

8.1 Swim lanes using Sequential Diagrams

A Sequential diagram is an interaction diagram that helps in plotting the sequence of a process from one operator or machine to another one. To know the sequential order of any kind of process, sequential diagram provides the platform we can also include the feedback loops, time intervals and also focus on system behavior.

Sequence diagrams are used to present the dynamic behavior of system design while class diagrams are system static structure. As one of two kinds of UML interaction diagrams, a sequence diagram shows interactions between objects arranged in a time sequence. [27]

While constructing a sequence diagram certain basic diagrammatic rules must be followed. The parallel lines which are also called as lifelines represent the different processes or objects existing in the entire process. The horizontal arrows show the exchanged messages in the order they occur during the process. In a sequence diagram the lifeline is an object and leaving the instance blank represents the anonymous and unnamed instances. In order to represent the type of message transmitting, we can write the name of message on top of the arrow. Solid arrows with full heads are synchronous calls, solid arrows with stick heads are asynchronous calls and dashed arrows with stick heads are the return calls. Activation boxes or method call boxes are the rectangle boxes drawn on top of lifelines to represent that processes are being performed in response to the message. Objects calling methods on themselves use messages and add new activation boxes on top to indicate a further level of processing. When an object is destroyed in the process, that can be represented with a “X” mark and few dotted lines are drawn below the object. If any message transmitted from outside, then it can be represented by a filled-in-circle or from a border of sequence.

8.1.1 Sequence Diagram on Black Hawk Incident

Here is an example that shows how to map an incident like Black Hawk (Iraq, 1994). The top most rectangle boxes are the important entities in the incident. As we explained above, the boxes on the vertical lines or lifelines represent the process existing for certain amount of time during the incident from an object.

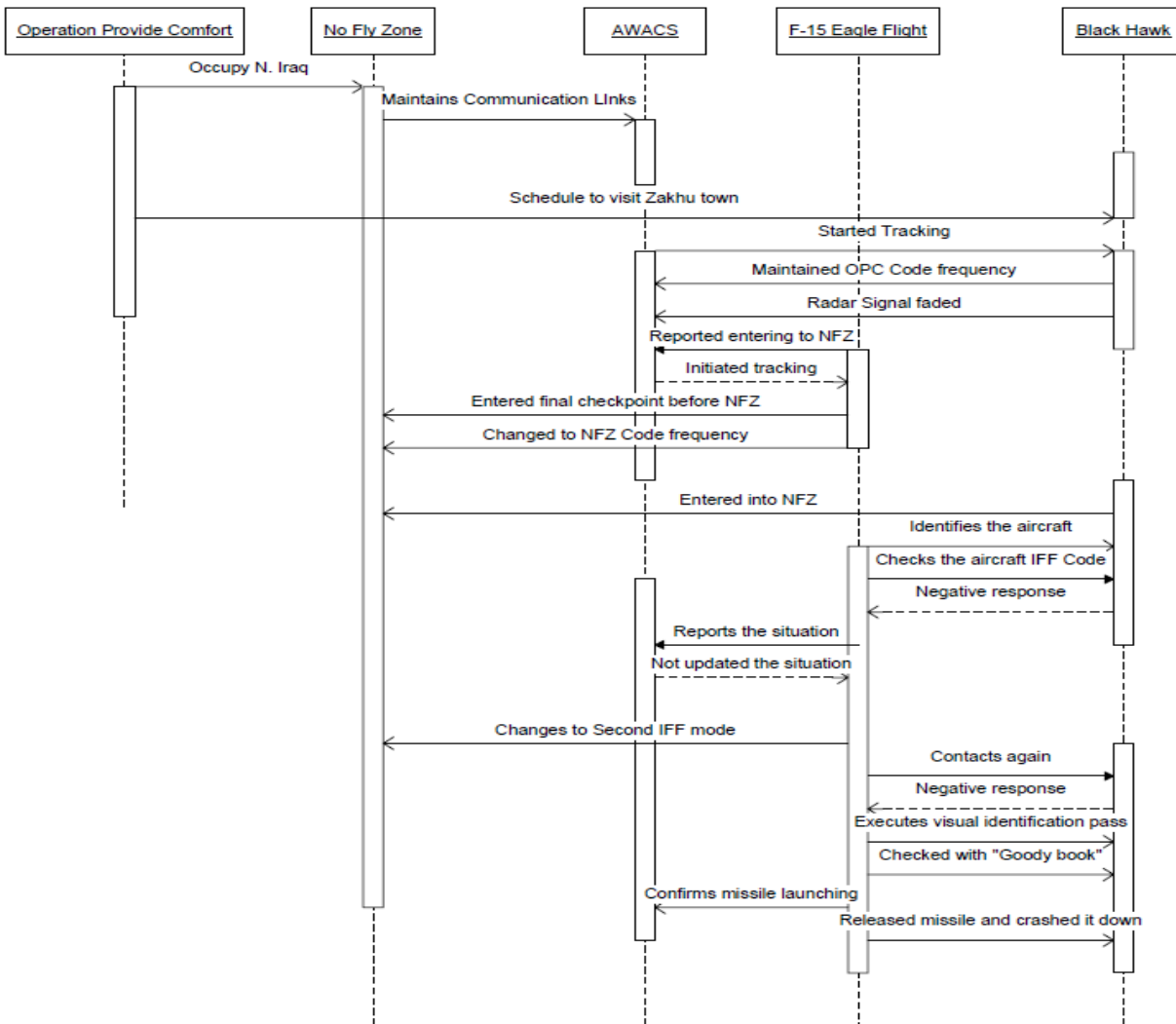


Figure 8.1: Sequence diagram illustrated the Black Hawk incident

As we discussed, this above example has lines of both kinds (dotted and solid with different shapes of arrows) that represent the step-by-step sequential activities in the incident where some of them are returned messages from objects.

This is one way of mapping the scenarios or processes undergoing in an organization to understand the activities going on in an enterprise and also helps in analyzing the organization. This sequence diagram gives as a clear vision on step-by-step sequential order of process but it's hard to find the influences and impacts among the objects because of certain changes or results occurs as the process continues.

8.1.2 Sequence Diagram on Unmanned Aerial Vehicle Mishap

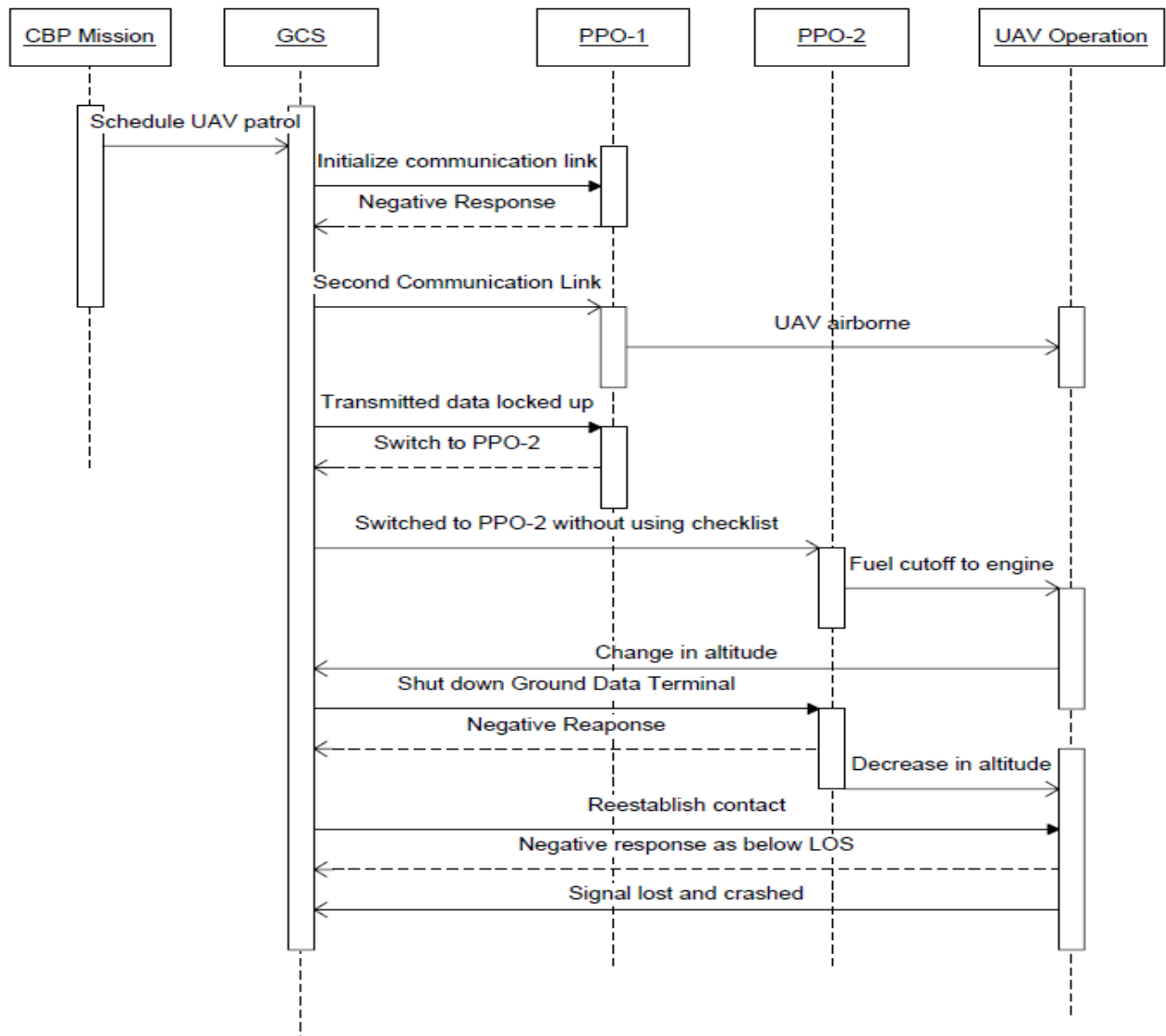


Figure 8.2: Sequence diagram illustrating the Mishap of Unmanned Aerial Vehicle

The above diagram explains what happened in an UAV mishap by representing the step-by-step processes or activities undergone at the time of incident.

8.2 Interpretive Structural Modeling Process

Interpretive Structural Modeling was first proposed by J. Warfield in 1973 to analyze the complex socioeconomic systems. ISM is a computer-assisted learning process that enables individuals or groups to develop a map of the complex relationships between the many elements involved in a complex situation. Its basic idea is to use experts' practical experience and knowledge to decompose a complicated system into several sub-systems (elements) and construct a multilevel structural model. ISM is often used to provide fundamental understanding of complex situations, as well as to put together a course of action for solving a problem.

8.2.1 Procedure

To construct an ISM structure to analyze a complex issue we need a team with group of skilled people in different areas like,

- a. Specialists: Having knowledge in several aspects of the issue.
- b. Stake Holder: People who are affected in some way by the outcome of the investigation.
- c. Modelers: People who work with the participants in structuring the issue.
- d. Facilitator: Person who takes the participants through the steps of formal processes.

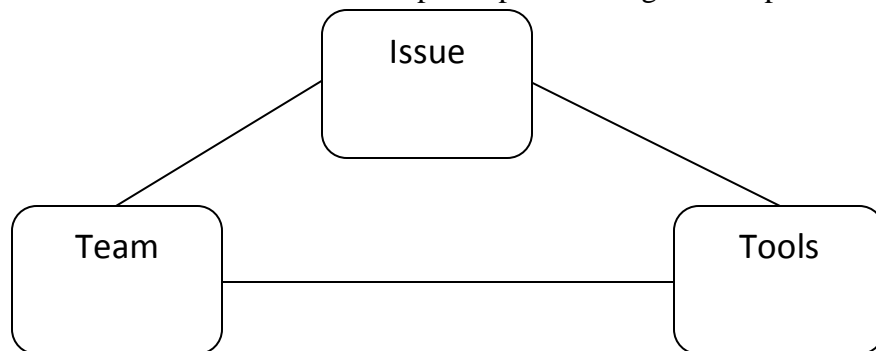


Figure 8.3: Interpretive Structural Modeling

Issues are identified and are thoroughly studied to find the elements on which the further process is implied. Once the elements are selected, then the decision on type of ISM comes in sequence. We have five different types of ISM structures which are considered depending on the following criteria,

- a. Intent Structure: Interrelations between set of objectives. Helps in clarifying thinking, explaining the future accomplishments of an organization or project, providing basis for taking action.
- b. Priority Structure: This structure is constructed when there is a requirement of ranking the number of elements in the order of priority.
- c. Attribute Enhancement Structure: Interrelations between set of factors, problems or opportunities.
- d. Process Structures: Involves sequencing of a set of activities.
- e. Mathematical Dependence Structures: This structure may be used to map the interrelations between a set of quantifiable elements.

Group size: This is another essential parameter in order to limit the debate among the participants. Usually it is better to have a group of 6-8 members in a team. This includes members from all categories that we considered above. Increase in group size leads to downfall in the quality of debate among the members and finally turns to waste of time by discussing unnecessary topics. The number of possible outcomes in a communication among “n” people is! $(n-1)$.

In general, set of elements to be structured are selected by members by analyzing the issue. But in other cases the elements are predetermined, like they may be a set of county highway

schemes, which have to be prioritized because the financial, resources to carry them all out are not available.

Nominal Group Technique is a process that has been found to work particularly well in conjunction with ISM. It is an efficient method for generating ideas in groups, for clarifying the generated ideas, for editing the generated ideas, and for developing a preliminary ranking of the set of ideas.

- I. Clarification of the trigger question
- II. Participants writing their ideas
- III. Recording the ideas on flipchart
- IV. Discussion
- V. Ranking the ideas in terms of importance

Matrix of element interactions: At this stage of process, we use the ISM software which helps in constructing a matrix with the selected elements. The software in the computer starts asking questions and the answers from the group are limited to “Yes” or “No”. For “Yes” it takes “1” and for “No” it takes “0”. An example for such matrix is shown below,

	E1	E2	E3	E4
E1	1	1	0	1
E2	0	1	0	0
E3	1	1	1	1
E4	1	1	0	1

Figure 8.4: Matrix model

E1, E2, E3, E4 are the elements and 1 = “Yes”, 2 = “No”

Once the matrix is constructed, the computer converts the matrix into a multi-level digraph with cycles. The facilitator then helps to explain the digraph to the group and lets them understand the content in the structure. Then the group discusses and changes will be made in the structure if there is a necessity. But, proper reasons must be noted down to make any changes in the structure since ideas get changed as we construct the model with more elements.

This technique has certain disadvantages and those are pointed below,

- a. No way to check the missing entities and there impacts
- b. Takes more time to rectify if any error occurs since everything is verbally noted
- c. Cannot recheck in the middle of the process and have to wait till the end
- d. No way to identify the influences among the entities
- e. Time taking

8.3 Cyclomatic Complexity

Current burning issue in the corporate world is the modularization of the software system which provides the ability to test and maintain the final result modules. The reason behind the facts that testability and maintainability are really important came from the ancient truth that a company spends half of the development time in testing and can spend most of the dollars in maintaining the systems. It's essential to have a mathematical technique that provides a quantitative basis for modularization and allow us to identify software modules that will be difficult to test and maintain. General motive for any company is to reduce the program size which looks easy but in a real time scenario it's not that easy because of several distinct control paths. In order to avoid these complications we need a mathematical tool like Cyclomatic Complexity.

The complexity measure approach allows to measure and control the number of paths through a program. But the raising question is, in a program with backward branch potentially has an infinite number of paths. Although it is possible to define a set of algebraic expressions that gives the total number of paths through the program, using total number of paths is impractical. Because of this, the complexity measure developed is defined in terms of basic paths that can generate all the possible paths.

The Cyclomatic number $V(G)$ of a graph G with n vertices, e edges and p connected components is

$$V(G) = e - n + p.$$

To utilize this mathematical tool, the program must associate with a directed graph that has unique entry and exit nodes. Each node in the graph represents a block of code in the program, flow represents the sequence among the blocks of code and the arcs represent the branches taken in the program. This graph is classically known as the program control graph and it is assumed that each node enters and exits from the succeeding and preceding nodes.

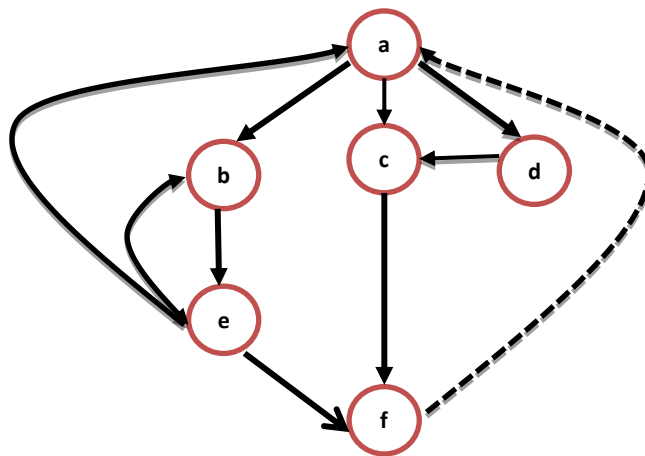
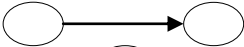
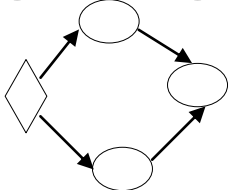


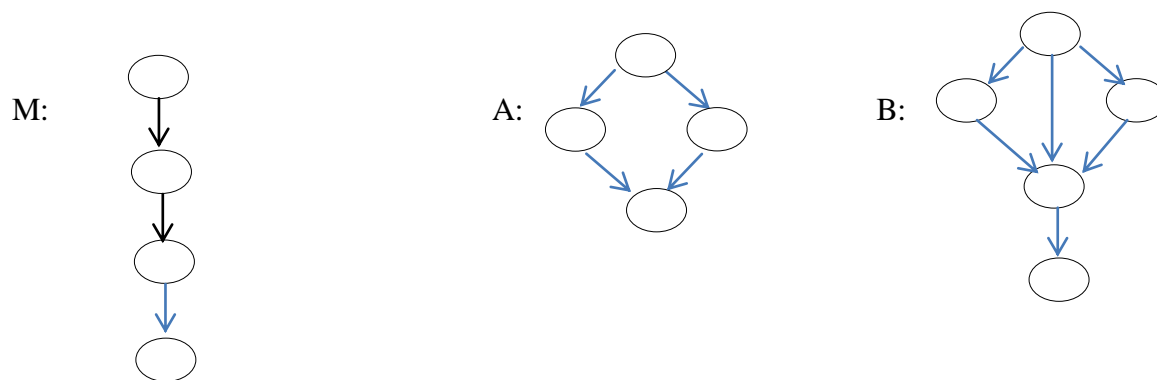
Figure 8.5: Example for Cyclomatic Complexity

The strategy behind the whole study is to measure the complexity of a program by computing the number of linearly independent paths $V(G)$. Below diagrams provide a better understanding about the control graphs.

CONTROL	STRUCTURE	CYCLOMATIC COMPLEXITY
Sequence		$v = 1 - 2 + 2 = 1$
If Then Else		$v = 4 - 4 + 2 = 2$

From the above complexity equation the term “P” is the number of connected components. A defined program control graph which has a unique entry and exit nodes, all nodes reachable from the entry and the exit reachable from all nodes would result in all control graphs having only one connected component.

To emphasize more on the program control graphs, let’s imagine a main program M and two subroutines A and B having a control structure.



Looking at the above diagrams,

$$v(M \cup A \cup B) = e - n + 2p = 13 - 13 + 2 \times 3 = 6$$

This method with $p \neq 1$ can be used to calculate the complexity of a collection of programs, particularly a hierarchical nest of subroutines. This expression can also be written as,

$$v(M \cup A \cup B) = v(M) + v(A) + v(B) = 6$$

In general, the complexity of the entire control graphs “C” with “k” connected components is equal to the summation of their complexities.

$$v(C) = e - n + 2p = \sum e_i - \sum n_i + 2k$$

$$= \sum (e_i - n_i + 2) = \sum v(C_i).$$

8.3.1 Analyzing Diagnostic interview using Complexity measurement

By applying this case on Cyclomatic complexity technique we can find the number of defects we have in the diagnosis process. To find the defects we have to convert the System Dynamics model into a Cyclomatic complexity diagram, where we consider the same entities that we choose to make the SD model.

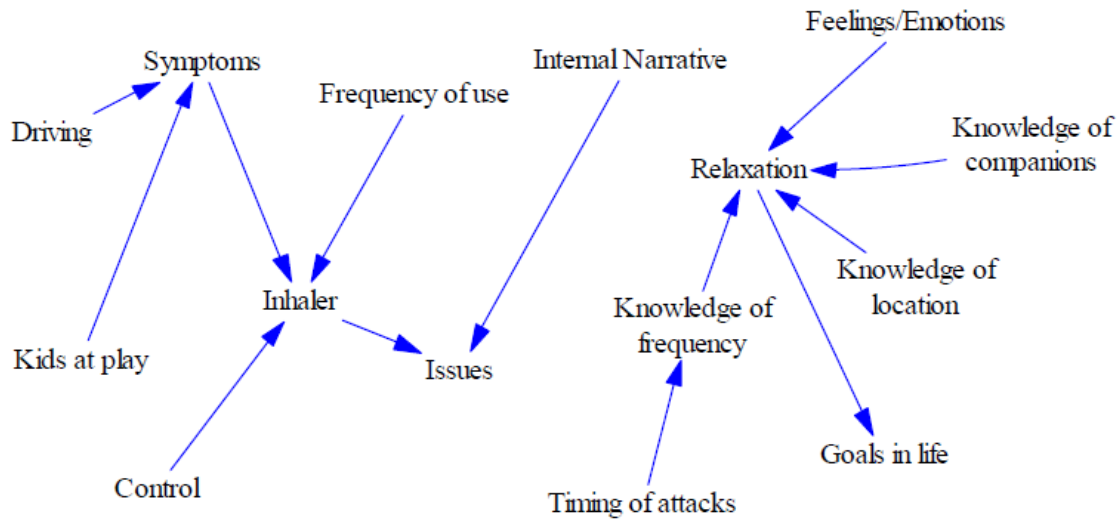


Figure 8.6: Diagnostic case with System Dynamics technique



Figure 8.7: Diagnostic Interview with Cyclomatic Complexity

$$V(G) = e - n + p.$$

$$V(G) = 14 - 15 + 10 = \underline{11}$$

By using this mathematical tool, it is clear that we have one defect in this diagnosis process. But this approach is correct mathematically, practically when we use this it doesn't give a clarity on the impacts and influences among the entities.

Chapter 9: Applications

9.1 Academics

Teaching quality is critical throughout the modern educational enterprise. Despite the guidance of many teaching approaches, teacher's narratives tend to remain within a defined space of educational enterprise elements. Restriction to habitual areas of discourse can be a major mistake, reducing the variety of classroom discussion topics. This phenomenon is well modeled by the abstract mistake of attribute substitution. Teaching narratives modeled with system dynamics shows that teachers are susceptible to a collapse of attention in which they focus on only a few attributes and the relations among them. In fact, excessive focus among a few attributes, and the relations among them, can lead to amnesia as to the full spectrum of relevant attributes, i.e., discussion topics. The principal amelioration for attribute substitution is a widening of attention to where awareness of the entire educational enterprise is maintained. The Zachman Framework can form the backdrop for a teaching narrative evaluation process, ensuring that no mistaken collapse of teacher and classroom attention occurs.

The CCAM is a general research approach that provides guidelines for the systematic generation of theory from data regardless of whether it is qualitative or quantitative. As essential characteristic of the approach is the continuous cycle of collecting and analyzing data. It follows the following four distinct stages,

- a) Comparing incidents applicable to each category
- b) Integrating categories and their properties
- c) Delimiting the theory
- d) Writing the theory

9.1.1 System Dynamics Model

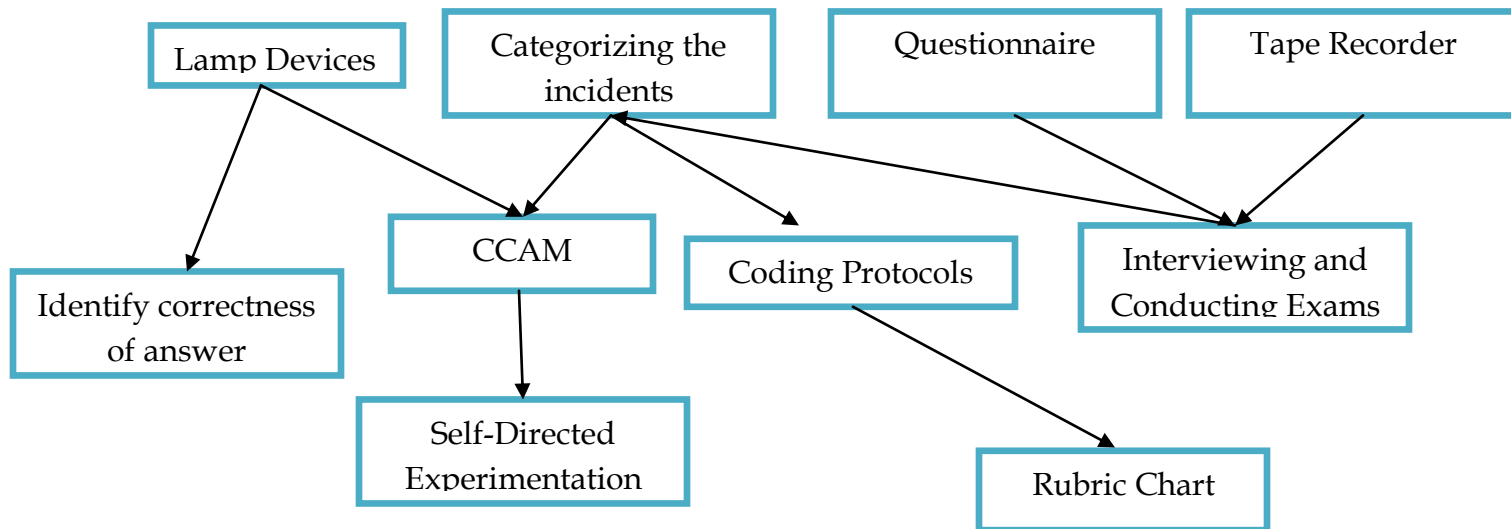


Figure 9.1: System Dynamics Model on CCAM

9.1.2 Zachman Framework

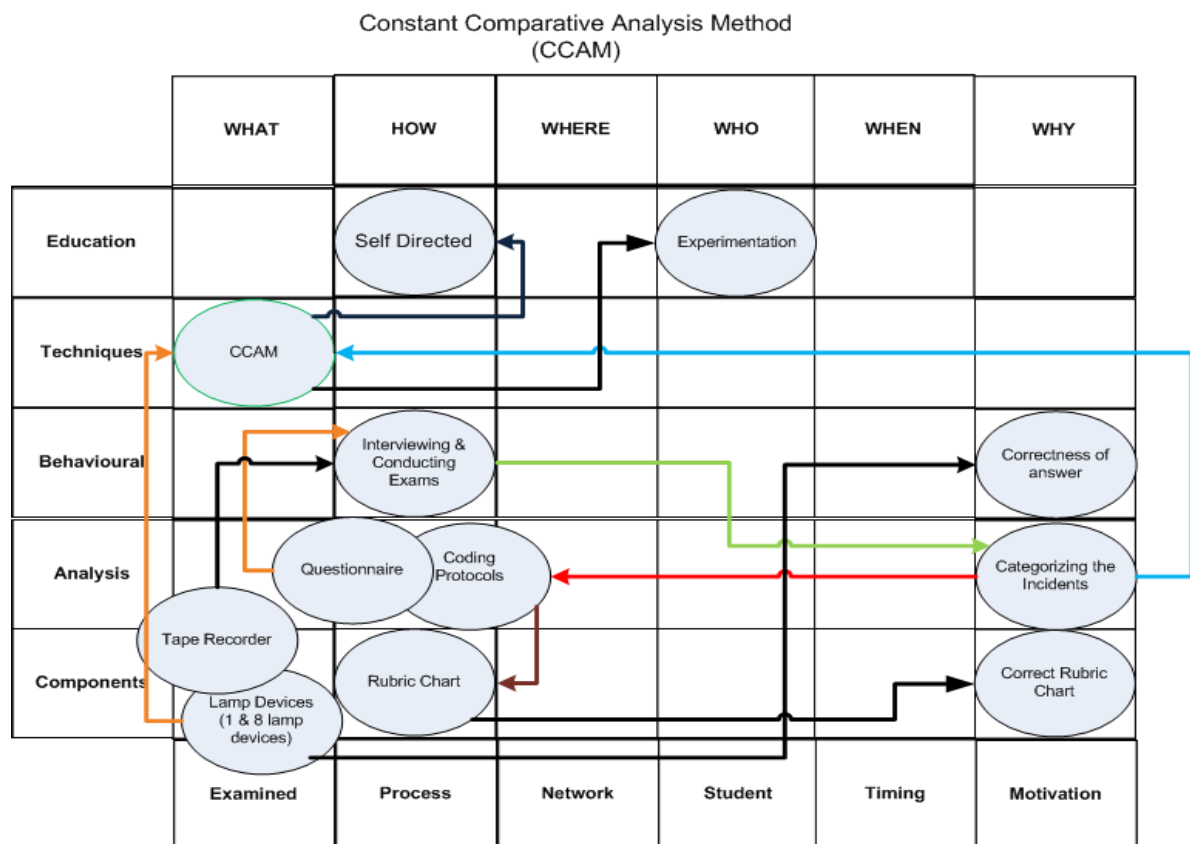


Figure 9.2: CCAM with System Dynamics overlaid on Zachman Framework

9.2 Defense

Decision making by Unmanned Aerial Vehicles (UAVs) is perhaps the most complex and crucial decision within the future air battle-space. Models for UAV decisions – whether by remote operators or by artificial intelligence -- must incorporate consideration for the usual multiplicity of important factors, interacting feedback loops among these factors, and the dynamic nature of the full diagnostic arena. A modeling technique that has the requisite variety of relevant considerations is presented. The technique has the potential to overcome mandatory time criteria, while considering the competence and robustness of high importance decisions.

Looking at the importance of decision making techniques in a system like this, we need to study the plausible narratives for UAV battle space decisions in detail not only to extract key factors involved in decision making processes, but also to illustrate the wide ontological origin of key decision making factors. Important factors in the narratives must be identified and mapped with a new System Dynamics methodology that incorporates a Zachman Framework that establishes the overall scope and context of the full decision making arena within the modern military enterprise. While traditional System Dynamics is able to link factors according to their influence on other factors, and in fact determine multiple interacting feedback loops in a rigorous and even exact mathematical fashion, the debility which was addressed in this research was the determination of the definite location of the usually unanchored factors into the matrix of the modern defense enterprise. This methodological technique set allows the ready identification of possibly disparate ontological origins of decision making factors. In the case where widely diverse factors must be taken into consideration, the technique provides for a graphical representation that summarizes important dimensions of choices.

Modeling representations of complex systems useful for the examination of changes that occur in any part or whole of these systems must allow for the performance of a requisite variety of essential analyses, in order to cover decision contingencies that have the potential to sway outcomes to extreme values. The fundamental purpose of decision making methods is to create a quantitative representation of different choices which enable incorporation of uncertainties and different representations of the decision maker's preferences for various possible outcomes. Decision analysis can validate scenarios regarding decisions, help to compare choices quantitatively, allow for rapid assessment of the effects of variations in assumptions on the optimal choice, and provide a mechanism for evaluating a given decision along various outcome dimensions like survival expectancy, performance and costs. The purpose of sophisticated modeling techniques is to make the model as realistic a reflection of the real world as possible, considering all constraints of available data, analyst time availability, and computational resources needed to evaluate the model.

9.2.1 Black Hawk Incident

To understand more about the technical aspects in the operation of a unmanned aerial vehicle, it's better to start analyzing a manned aerial vehicle mishap. In this study, we have chosen the Black hawk down case happened in 1994 after the Persian war in Iraq.

After the Persian Gulf War, Operation Provide Comfort (OPC) was created as a multinational humanitarian effort to relieve the suffering of hundreds of thousands of Kurdish refugees who fled into the hills of northern Iraq during the war. The goal of OPC was to ensure the security of relief workers assisting Kurdish refugees and to provide a safe haven for the

resettlement of the refugees. In addition to the operations on the ground, a major portion of OPC's mission was to occupy the airspace over northern Iraq.

To accomplish this task, a no-fly-zone (NFZ) was established that included all airspace within Iraq north of the 36th parallel. The coalition also established a security zone for the Kurds inside the NFZ, into which no Iraqi military could enter.

On April 14, 1994 two U.S. Air Force F-15's patrolling the NFZ shot down two U.S. Army Black Hawk Helicopters carrying 26 people including 15 U.S. citizens and 11 others (British, French, and Turkish Military officers as well as Kurdish citizens). Everyone was killed in one of the worst air-to-air friendly fire accidents involving U.S. aircraft in military history. Both flights were flying under the control of an AWACS (Airborne Warning and Control Systems) aircraft, the most advanced system of its type in the world. The weather was clear, all the sophisticated electronic and technical systems appeared to be operational, and the people involved were all highly trained and experienced. After two years and hundreds of hours of extensive investigation by accident boards, autonomous Army and Air Force teams, investigative reporters, lawyers, and congressional committees and their staff, no single cause was identified. According to an Air Combat Command official who was familiar with the official investigations, over 130 different mistakes were identified as being involved in the shoot down. Several analyses of the accident have been provided beyond the official investigation board report, most notably books by Scott Snook and Laura Piper, the mother of one of the Army officers killed. The GAO also wrote a report that evaluated the official accident reports. Each of these sources gives different explanations of the accident, in some aspects significantly different, due to a focus on different factors involved. In this paper, we use a control-based accident model to try to separate fact from interpretations of those facts and to provide a more complete and independent

analysis of the accident process. The goal of the model is not to determine blame but instead to more completely understand all the factors involved, particularly those that can be changed to prevent future accidents. In the next section, we provide a general description of the proximate events involved in the loss. Then a control-based model explaining these events is provided.

The Black Hawks (Eagle Flight) entered the NFZ through Gate 1, checked in with the AWACS controllers and flew to Zakhu, a town just inside the northeast corner of the security zone and forward headquarters for Army OPC ground operations. The AWACS surveillance officer labeled the flight on the radarscope track. The Black Hawk pilots did not change their IFF (Identity Friend of Foe) code from 42 (the code for all friendly fixed-wing aircraft flying in Turkey on that day) to 52 (the code for the NFZ). They also remained on the enroute radio frequency instead of changing to the frequency to be used in the NFZ. When the helicopters landed at Zakhu, their radar and IFF returns on the AWACS radarscopes faded. Thirty minutes later, Eagle Flight reported their departure from Zakhu to the AWACS and said they were enroute to Irbil (a town deep in the NFZ). The enroute controller reinitiated tracking of the helicopters. Two F-15s were tasked that day to be the first aircraft in the NFZ and to ‘sanitize’ it (check for hostile aircraft) before other coalition aircraft entered the area. The F-15s reached their final checkpoint before entering the NFZ approximately an hour after the helicopters had entered. They turned on all combat systems, switched the IFF Mode I code from 42 to 52, and switched to the NFZ radio frequency. They reported their entry into the NFZ to the AWACS. At this point, the Black Hawks’ radar and IFF contacts faded as the helicopters entered mountainous terrain. The computer continued to move the helicopter tracks on the radar display at the last known speed and direction, but the identifying *H* symbol (for helicopter) on the track was no longer displayed. Two minutes after entering the NFZ, the lead F-15 picked up hits on its

instruments indicating that it was getting radar returns from a low and slow-flying aircraft. The lead F-15 pilot alerted his wingman and then locked onto the contact and used the F-15's air-to-air interrogator to query the target's IFF code. If it was a coalition aircraft, it should have been squawking Mode I, code 52. The scope showed it was not. He reported the radar hits to the controllers in the AWACS, and he was told they had no radar contacts in that location. The lead F-15 pilot then switched the interrogation to a second IFF mode (Mode IV) that all coalition aircraft should be squawking. For the first second, it showed the right symbol but for the rest of the interrogation (4 to 5 seconds) it said the target was not squawking Mode IV. The lead F-15 pilot then made a second contact call over the main radio, repeating the location, altitude, and heading of his target. The wing F-15 pilot replied that his equipment showed the target. This time the AWACS enroute controller responded that he had radar returns on this scope at the spot but did not indicate that this might be a friendly aircraft. After making a second check of Modes I and IV and again receiving no response, the F-15 executed a visual identification pass to confirm that the target was hostile. He saw what he thought was an Iraqi helicopter. He pulled out his "goody book" with aircraft pictures in it, checked the silhouettes, and identified the helicopters as Hinds, a type of Russian helicopter flown by the Iraqis. The F-15 wing pilot also reported seeing two helicopters, but never confirmed that he had identified them as Iraqi aircraft. The F-15 lead pilot called the AWACS and said they were preparing to engage enemy aircraft, cleared his wingman to shoot, and armed his missiles. He then did one final Mode I check, received a negative response, and pressed the button that released the missiles. The wingman fired at the other helicopter and both were destroyed. [28][29][30]

9.2.1.1 Analysis on the Incident using System Dynamic Techniques

It is an approach or a tool that helps in understanding the behavior of any complex system. The model consists of several internal loops which show the complete process flow in an unstructured and informal way.

In this paper we are discussing about the mishaps of "Manned Aerial vehicles" and "Unmanned Aerial vehicles". The diagram shown below is a system dynamics model of "Black Hawk incident 1994" in Iraq.

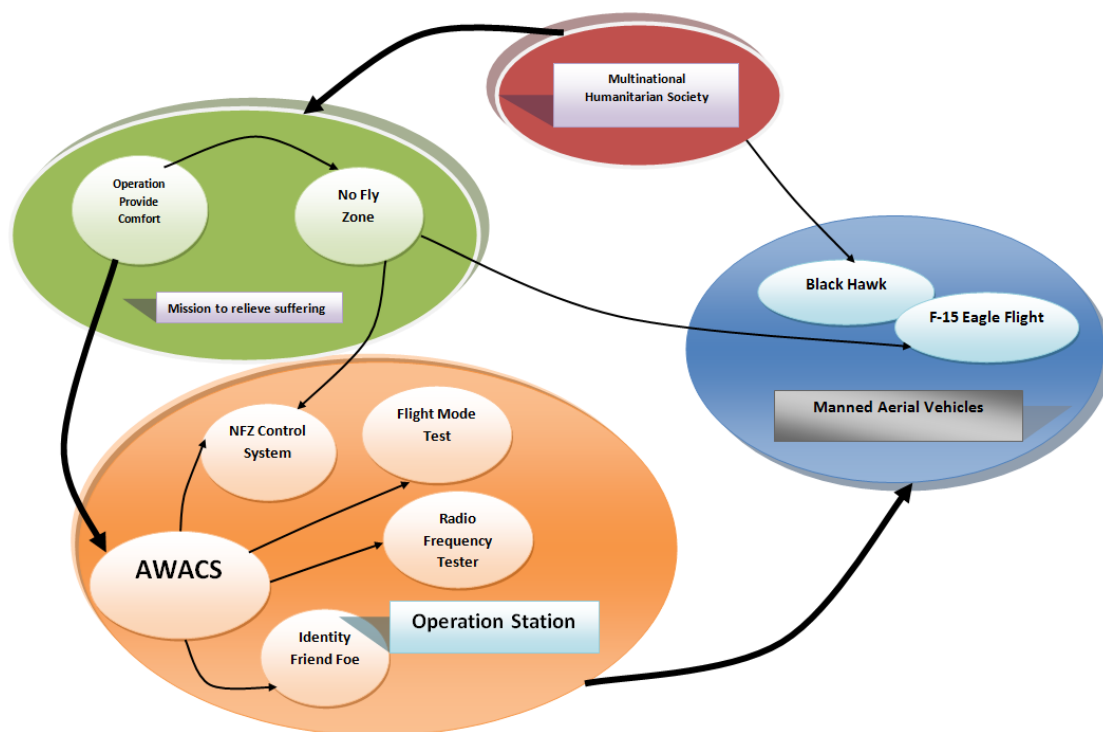


Figure 9.3: Black Hawk incident with System Dynamics Model

In the above system dynamics model, there are four important entities that play a major role. Some of them contain internal entities which have effect on each other. This approach gives a better picture on how entities in a system influence each other and perform their duties for the

final objective of the mission. But still a system dynamics model is not giving a well structured idea about the situation. So, we have applied the same scenario with a Zachman Framework model.

9.2.1.2 Analysis on the Incident using Zachman Framework

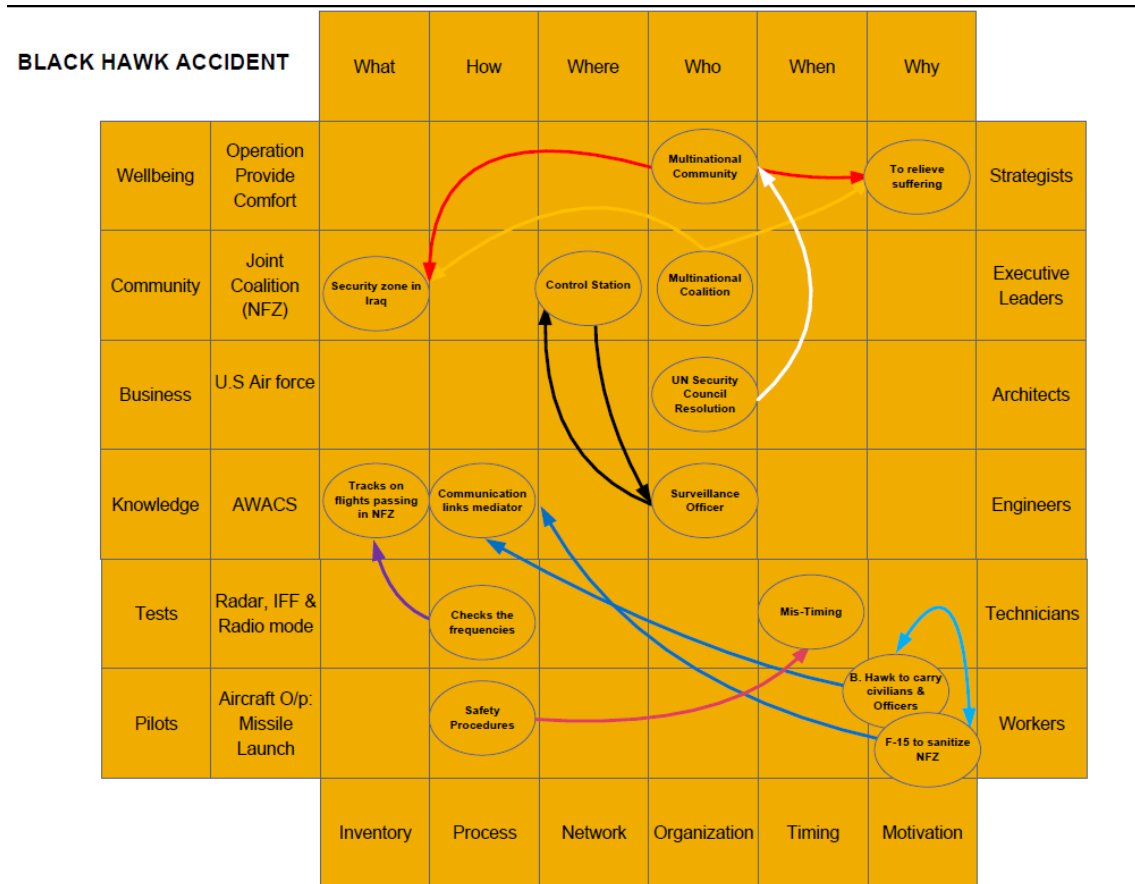


Figure 9.4: Black Hawk incident shows with the System Dynamics overlaid on Zachman Framework

Using this model, it's easy to identify the influence of different activities on each other. This model helps in understanding the drawbacks in a scenario and supports a better decision making.

9.2.2 Unmanned Aerial Vehicles

In April 2006, an unmanned aircraft (UA) collided with the terrain due to loss of engine power while patrolling the southern U.S. border on a Customs and Border Protection (CPB) mission. Pilot Payload Operator (PPO-1) was used for initial power-up and to control the fuel valve and Camera Control console (PPO-2) was to adjust the camera. The flight was being flown from the Ground Control System (GCS). The pilot was ordered to check the instruction manual before changing the controller from PPO-1 to PPO-2. The lever position should get matched in both before shifting the lever. On that day, operator failed to check and that resulted in complete fuel cut off and shut down of the engine. Within few minutes UAV lost its amplitude and communication with GCS and crashed [31].

9.2.2.1 Analysis on the Incident using System Dynamic Techniques

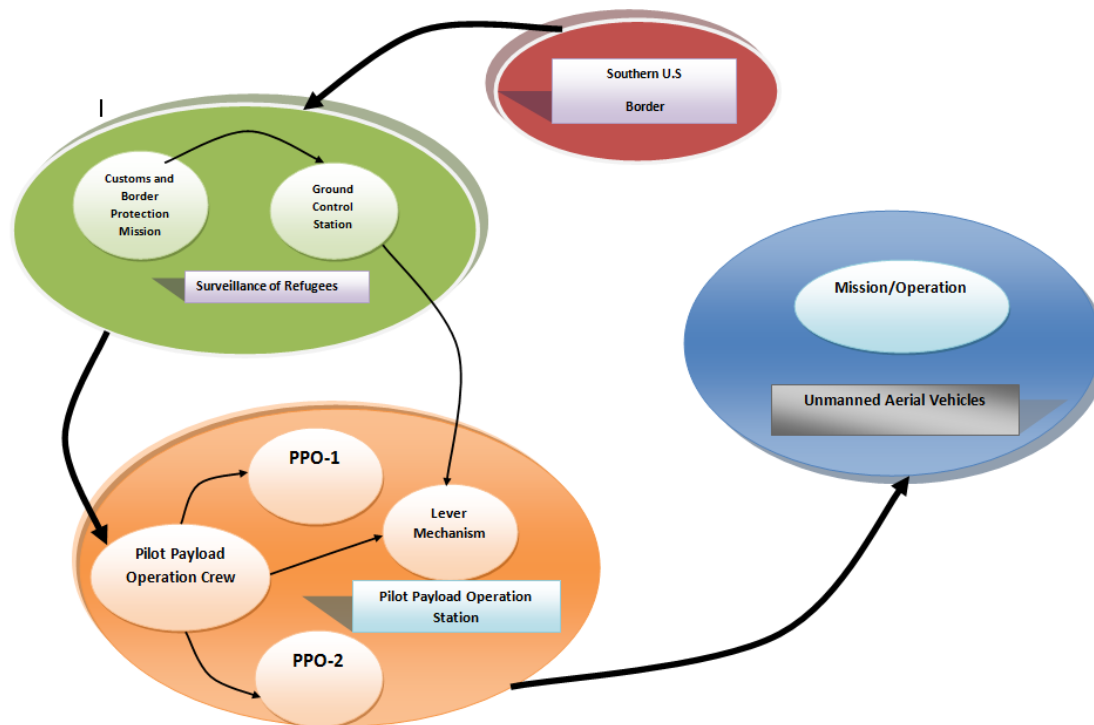


Figure 9.5: UAV Mishap with System Dynamics Model

In the above system dynamics model, factors involved in the crash of UAV are identified with internal loops between them. But still it's important to identify the influences on each factor so that to detect the reasons behind the mishap of UAV and also helps in implementing decision making techniques. Zachman frame work is a tool that provides required detailed information to obtain a decision making strategy.

9.2.2.2 Zachman Frame work Model

In this model, the factors that influenced in causing the final mishap of UAV are identified. This model even helps in implementing a better technique to avoid such failures in future. The model below shows the influence diagram which explains the effect of entities on each other. This gives a better imagination on the situation than a System Dynamics model.

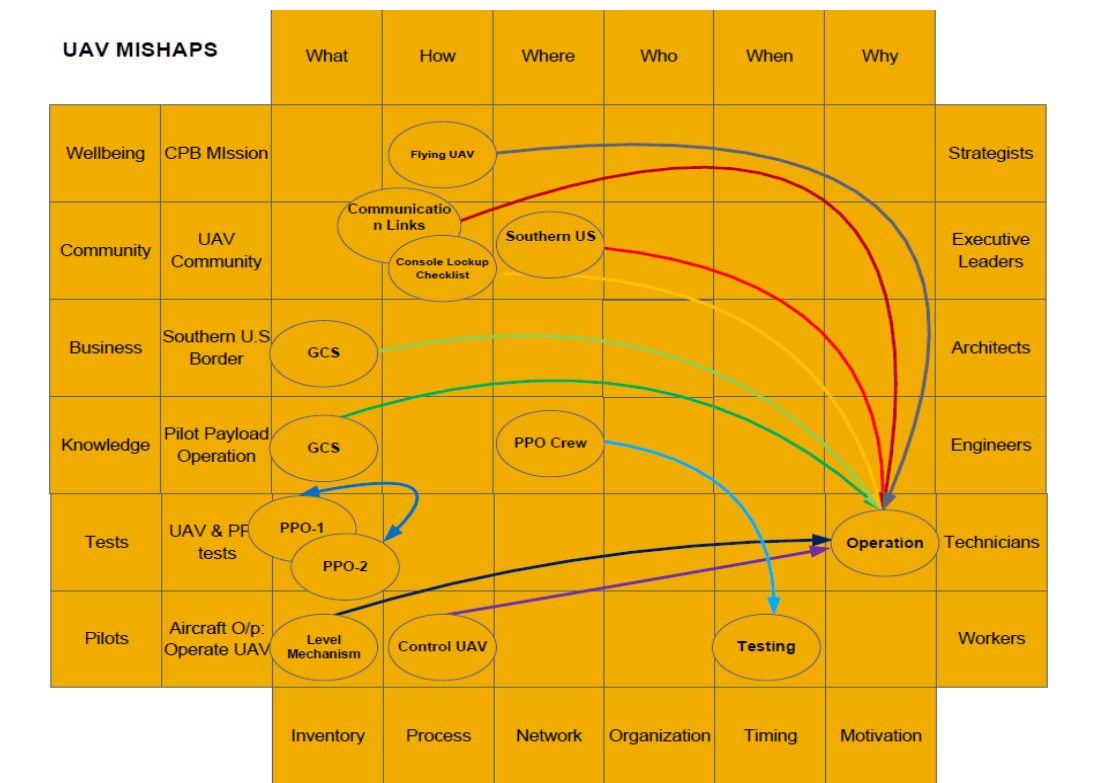


Figure 9.6: UAV Mishap shows with a System Dynamics overlaid on Zachman Framework

Chapter 10: Future Work

10.1 Vensim Modeling

As a future work, to obtain a better abstract model for analyzing any type of complex area it is important to research on the existing techniques and Vensim Modeling is one among them.

Vensim is simulation software made by Ventana Systems. Its purpose is to help companies to find an optimal solution for various situations that need analysis and where it's necessary to find out all possible results of future implementation or decision. Vensim is able to simulate dynamic behavior of systems to analyze without appropriate simulation software, because they are unpredictable due to many influences, feedback loops etc.

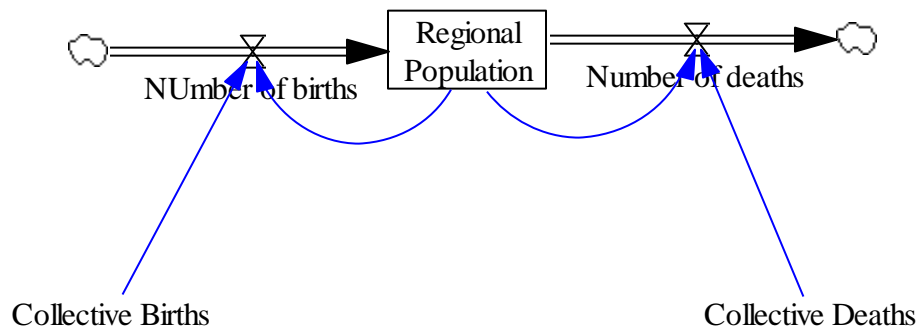


Figure 10.1: Model for Population count with Vensim Modeling Technique

10.2 Word Parser

By knowing the frequency of words used in a series of interviews in any field, we can identify the important entities in the scenario which has notable effect on the other entities or the entire system. By finding the high frequency words, it would be easy to concentrate on certain entities in the system which saves time and helps in analyzing the core entities instead of analyzing the entire system. We have several word counting tools like Word Parser, N-Vivo etc.

10.3 Applying the concept on other techniques

As a future work it is useful to apply this concept on other techniques in Soft System Methodologies and also other tools that can interact with the internal loops, feedback loops etc.

Chapter 11: References

1. “Systems Engineering, is it a new discipline” by Eur.Ing.A.G.Stoddart
2. “What is a System Approach to Prediction?” Antonin Wild; Management Consultant; Ottawa.
3. Smith, E. D. (2006). Tradeoff studies and cognitive biases. University of Arizona, Tucson, AZ.
4. Kahneman, D. (2003). A perspective on judgment and choice: Mapping bounded rationality. *American Psychologist*, 58(9), 697-720.
5. “A Framework for Modeling and Simulating Human Behavior in Complex Systems”, Seung Man Lee, Roger W. Remington, Ujwala Ravinder, NASA Ames Research Center, MS 262-4, Moffett Field, CA 94035.
6. “A Soft-Systems Methodology Approach for Product and Process Innovation” by Adrien Presley, Joseph Sarkins, and Donald H. Liles, *IEEE Transactions on Engineering Management* Vol 47, No. 3, August 2000.
7. A. Platt and S. Warwick, “Review of soft systems methodology,” *Ind. Manage. Data Syst.*, vol. 95, no. 4, pp. 19–21, 1995.
8. P. Checkland, *Systems Thinking, Systems Practice*, London: Wiley, 1981.
9. C. A. Macias-Chapula, “Development of a soft systems model to identify information values, impact and barriers in a health care information system,” *J. Inform. Sci.*, vol. 21, no. 4, pp. 283–288, 1995.

10. Soft System Dynamics Methodology (SSDM): Combining Soft Systems Methodology (SSM) and System Dynamics (SD), Ricardo Rodriguez-Ulloa¹ and Alberto Paucar-Caceres, *Systemic Practice and Action Research*, Vol. 18, No. 3, June 2005 (2005)
11. <http://www.ahrq.gov/about/annualmtg07/0927slides/smith/Smith-9.html>
12. Wurman, R. S., Understanding Healthcare, United Health Group, Markle Foundation.
13. Jonas, S., Goldsteen, R., and Goldsteen, K., An Introduction to the U.S. Health Care System, 6th Edition, 2007 Springer Publishing Company, LLC.
14. A Framework for Information systems Architecture, J. A. Zachman, IBM Systems Journal, Vol 38, NOS 2&3, 1999
15. A framework for information systems architecture, J. A. Zachman, reprinted from IBM Systems Journal, VOL26, NO 3, 1987; © 1987, 1999
16. A Different Kind of Life Cycle: The Zachman Framework, David C. Hay, *Copyright © 2000, Essential Strategies, Inc.*
17. <http://ccollins.wordpress.com/2008/02/16/introduction-to-the-zachman-framework>
18. Coulehan, J. L., & Block, M. R. (2006). The medical interview: Mastering skills for clinical practice. Philadelphia: F. A. Davis Company.
19. Enelow, A. J., Forde, D. L., & Brummel-Smith, K. (1996). Interviewing & patient care. New York: Oxford University Press.
20. Newell, R. (1994). Interviewing skills for nurses and other health care professionals: A structured approach. London: Routledge.
21. Bernstein, L., & Bernstein, R. S. (1985). Interviewing: A guide for health professionals. Norwalk, CT: Appleton-Century-Crofts.

22. Froelich, R. E., & Bishop, F. M. (1969). Medical interviewing: A programmatic manual.
Saint Louis: The C. V. Mosby Company.
23. Aveyard, H., & Sharp, P. (2009). A beginner' guide to evidence based practice in health and
social care. Berkshire, England: Open University Press, McGraw Hill.
24. Petitti, D. B. (1994). Meta-analysis, decision analysis, and cost-effectiveness analysis:
Methods for quantitative synthesis in medicine. New York: Oxford University Press.
25. Seedhouse, D. (2005). Values-based decision-making for the caring professions. New York:
John Wiley & Sons, Ltd.
26. Forrester, J. W. (1971). Counterintuitive behavior of social systems. *Technology Review*,
73(3), 52-68.
27. Forrester, J. W. (2009). System dynamics: The classroom experience: Quotations from K-12
teachers.Unpublished manuscript, Massachusetts Institute of Technology (MIT), Cambridge,
MA.
28. Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems
Journal*, 26(3).
29. “A Formal Semantics of UML Sequence Diagram”, Xiaoshan Li, Zhiming Liu and He
Jifeng, Faculty of Science and Technology, University of Macau, Macau. *International
Institute for Software Technology, United Nations University, Macau, Department of
Mathematics and Computer Science, University of Leicester, UK.*
30. Scott A. Snook. *Friendly Fire: The Accidental Shoot down of U.S. Black Hawks over
Northern Iraq* , Princeton Paperback.
31. Piper, Joan L. (2001). *Chain of Events: The Government Cover-up of the Black Hawk
Incident and the Friendly-fire Death of Lt. Laura Piper*. Brassey's.

32. Arana-Barradas, Louis A. (1996). ["Black hawk incident "tragic series of errors"'](#). Retrieved 2007-02-16.
33. <http://www.nts.gov/>.
34. Redelmeier, D., & Shafir, E. (1995). Medical decision making in situations that offer multiple alternatives. *Journal of the American Medical Association*, 273(4), 302-305.
35. Smith, E. D., Son, Y. J., Piattelli-Palmarini, M., & Bahill, A. T. (2007). Ameliorating mental mistakes in tradeoff studies. *Systems Engineering*, 10(3), 222-240.

Chapter 12: Curriculum Vita

Bharath Bhushan Dantu was born in Rajahmundry, India. The first son of Pardha Saradhi and Sundari, he graduated from S.C.S.V.M.V. deemed university, Tamil Nadu, India in the spring of 2008. He started to pursue his Master of Science degree from Industrial Engineering at University of Texas at El Paso from fall 2009. At UTEP he worked as a research assistant, graduate assistant and as a teaching assistant in different areas. He is also member of Tau Beta Pi engineering honor society.

Permanent Address: 712, Prospect Street, Apt #12, El Paso, Texas – 79902.

This thesis was typed by Bharath Bhushan Dantu.