Making Complications Simple: using Planguage

Version 30 Sept 2015, 1st Draft

Abstract:
If a system increases in 'complexity', more parts and more connections, then it might appear to be more 'complicated' for us to understand it.

But, this depends not only on the system complexity, but on
- our personal intellectual capability (experience, knowledge, intelligence), and
- the 'tools' we choose to apply to analyzing the system.

The main purpose of this paper is to present certain tools, from our 'Planning language' ('Planguage'), which arguably make complex systems 'less complicated' to the intelligent tooled-up observer.

It is our contention that much literature makes a fundamental error of classifying systems as 'difficult to understand' when that literature fails to acknowledge that certain 'perception' tools would make them easier to understand.

The 'system structure' (complexity) is only one element to consider. The ability to understand, and to predict behavior also requires consideration of human capability, and suitable tools to enhance that human perception ability.

Definitions:
There are differing opinions on how to use these terms, but I have carefully and consciously chosen definitions close to dictionary definitions. It is is *irrelevant* that other people choose different terms, the *essence* is about the **concepts**, the definitions. Call the concepts whatever you like. [5]

Complex:
Compound in structure; consisting of interconnected parts.

Complicated:
Anything, that is difficult to understand, estimate or predict the actions, reactions, and attributes of.

Ten Principles of Simplification: 'Making Complexity Uncomplicated'

1. **Lord Kelvin's Principle 1**: If you *quantify* a variable attribute, it becomes more intelligible.[8]
2. **Lord Kelvin's Principle 2:** If you *measure* a variable attribute the system becomes more intelligible. [8]
3. **The Deming Feedback Principle: PDSA** [10]: If you compare attribute measurements with earlier estimations of them, you will get more understanding of a complex system.
4. **The Cartesian Decomposition Principle**: If you decompose a high-level generic attribute (like Usability, Maintainability, Security [CE5]) into a set of sub-attributes, you will get a tool for understanding the system, and understanding some critical aspects of it.

5. **Santayana's Learning from History Principle**: If you attempt to determine 'benchmark' points (such as Past levels, Records and Trends [CE, VP]) on critical attribute scales of measure, you will get, and can share, basic insights about a complex system, at a level *above* the complexity that generates the benchmark levels.
6. **The Scientific Experiment Principle**: If you build or modify any system, using new design strategies, one small incremental step at a time; you can get an understanding of the multiple performance values contributions, and multiple costs, that are due to that particular strategy, incremental measurement [Evo].
7. **The Side Effects Principle**: if you consciously model, estimate and measure the side effects of your individual designs and strategies, for example using an Impact Estimation table [CE, VP] you will better understand the bottom-line effects of a complex set of system components, interactions and their architecture.
8. **The Small Change Principle**: If you undertake to measure incremental effects of any change to a system, early, frequently, and with small doses, then you will gain a better understanding of the system component's interactions.
9. **The Who, Where, What and If Principle**: If you decompose your objectives and strategy application, by 'critically different dimensions' of people, tasks, places and events then you can much better understand the behavior of the system, under these specified conditions. (Scale parameters, VP 1.9)
10. **The Formal Model Principle**: if you take the effort to build a multi-dimensional and multi-level quantified relationship model of your system, using Planguage, for example and a tool [9], you have a much better chance to understand aspects of the system, on an as-needed basis.

Planguage Tools for Simplification (VP: **tinyurl.com/valueplanning**)

1. Value Quantification. See VP 1.1 (and other places)
2. Value Decomposition. VP 5.3
3. Numeric Value Relationships. VP 6.4
4. Relationship Specifications. VP 9.3
5. Strategy Impact Estimation VP Part 4, 6.5
6. Impact Uncertainty. VP 9.5
7. Impact Credibility. VP 9.5
8. Early, frequent numeric value and cost feedback. VP 9.10
9. Impact Tables: Multiple Strategies and effects on multiple objectives and costs.VP 9.7
10. Automated Modelling Tools. VP 1.3
11. Scale Parameters as a modelling and decomposition device. VP 1.9

References:

1. [VP] Value Planning Manuscript, 2015.Gilb. http://tinyurl.com/**value**planning

2. [CE] Competitive Engineering book, 2005. Gilb. tinyurl.com/CEset2015
http://www.gilb.com//dl540
Gilb, Tom, Competitive Engineering, A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage, ISBN 0750665076, 2005, Publisher: Elsevier Butterworth-Heinemann.

3. TEDx Talk Quality Quantification. http://bit.ly/TomTED

4. Concept Glossary, Planguage, tinyurl.com/CEset2015

5. 'A Conceptual Glossary for Systems Engineering: Define the Concept, don't quibble about the terms'. Gilb. 2006. http://www.gilb.com/dl565 .

6. [CE5] Competitive Engineering book chapter 5, Scales of Measure: http://www.gilb.com/tiki-download_file.php?fileId=26

7. [Evo] CE Chapter 10: Evolutionary Project Management: http://www.gilb.com//tiki-download_file.php?fileId=77

8. Kelvin, inVP 1.6. ""In physical science the first essential step in the direction of learning any subject is to find principles of numerical reckoning and **practicable methods** for **measuring some quality** connected with it."

9. Richard Smith's Planguage Tool: http://needsandmeans.com

There have been several tools supporting Planguage. This tool by Richard Smith rsmith@rsbatechnology.co.uk is emerging in 2015. We have used it on training courses in 2015. I am impressed by it capabilities and ease of use. It is very helpful on the *dynamic prioritization* methods described in the VP book in Section 6.1 and on.

10. **W Edwards Deming,**
    a. 'Out of the Crisis'. book
    http://mitpress.mit.edu/books/out-crisis
    B. **PDSA Plan Do Study Act The Deming Cycle**
    Original Deming Lecture to Top Management 1950 Japan:
    http://deming-network.org/deming_1950.htm
    C. Evolution of the PDSA Cycle:
    http://www.cologic.nu/files/evolution_of_the_pdsa_cycle.pdf