Diogenes, a Search for

Unintended Consequences Process

A Class Project for SIE-454/554 Fall 2010

A. Terry Bahill

<http://www.sie.arizona.edu/sysengr/sie554/>

December 14, 2010

Copyright ©, 2010, Bahill

He who loves correction loves knowledge,

but he who hates reproof is stupid. Proverbs 12:1

This document is being written iteratively in the fall of 2010 to help the Systems and Industrial Engineering students in SIE-454/554 at the University of Arizona to model and design systems. This document is **not** intended to be a *complete* design (for obvious reasons): instead, it tries to show a few good examples of each item that might be in a typical set of design documents. It has only a few use cases and only one tradeoff study. This document is located at

<http://sysengr.engr.arizona.edu/forStudents/index.html>

This document is being developed iteratively. It is being developed in parallel with the students’ designs, but a little bit earlier. I am exposing my mistakes in real time. The students are supposed to watch and learn from my mistakes. I then incorporate the best of their ideas into this document.

Document 1: The Problem Situation Document

This package contains the design for a process to help systems engineers predict unintended, but foreseeable, consequences of a new system that is being designed. This package contains the requirements, a risk analysis, the required behavior (functions), use cases, design diagrams, the procedure used to test the process, validation and verification for this process.

This process is named after Diogenes of Sinope, a Greek philosopher who lived in the fifth century B. C. Diogenes was a Cynic and disdained conventional wisdom and political correctness. He carried a lantern in the daytime saying that he was "in search of an honest man."

**Product Position Statement:** For systems engineers who need to ensure the global success of a system being designed, Diogenes is a process that will help predict unintended, but foreseeable, consequences of the new system. Unlike risk discovery and failure analysis, Diogenes identifies future effects on other systems that might be caused by the system being designed.

Abbreviations that will be used often

SystemZ is the name of a new system being designed that Diogenes will be applied to.

PAL is the process assets library, the place where all of the project’s important files are kept.

UiCs is unintended consequences.

BiST is Built-in Self-Test.

BICS is Bahill Intelligent Computer Systems

Philosophy

The purpose of Diogenes is to predict unintended, but foreseeable, consequences of a new system being designed. However, funding for a new process like Diogenes is problematic. Luckily, discovering negative unintended consequences can be done with a process similar to risk discovery. Therefore, Diogenes will be packaged with risk analysis and will be owned by the Risk Management or the Quality Assurance departments of a company. However, we have found that defects, positive unintended consequences and opportunities for BiST could also be discovered at the same time. By searching for all five (defects, risks, opportunities for BiST, positive UiCs and negative UiCs) at the same time, Diogenes can add value, while not adding organizational structure or funding channels.

Of course, like all systems engineering processes, this process is iterative. The emphasis will shift in each iteration. In the early phases of the system life cycle, it will discover mostly defects and risks. In later phases, it will discover more opportunities for BiST, positive UiCs and negative UiCs.

For engineers analyzing design artifacts of a SystemZ, Diogenes will produce five prioritized lists: (1) defects in development documents such as requirements, programming code, test plans and designs, (2) risks that could adversely affect SystemZ, (risks could be divided into risks and opportunities, but this is seldom done in industry), (3) opportunities for BiST, (4) positive UiCs that could beneficially affect other systems (5) negative UiCs that could adversely affect other systems.

Owner

Diogenes will be owned by the Risk Management or the Quality Assurance departments of a company. It will be a normal part of the risk analysis process. The importance of this paragraph is that it states how the process would get funded.

Work Products

Diogenes will create and maintain five databases: defects, risks, opportunities for BiST, positive UiCs and negative UiCs. The Moderator and the Systems Engineer will consolidate and edit the five databases to create the following five prioritized lists.

(1) The list of defects in development documents such as requirements, programming code, test plans and designs is given to the Author/Designer for resolution. Correcting these defects could be managed by the Moderator or they could be submitted to a change control board (CCB).

(2) The list of risks that could adversely affect SystemZ (risks could be divided into risks and opportunities, but this is seldom done in industry) is given to Risk Management, which assigns likelihood of occurrence and severity of consequences.

(3) The list of opportunities for inexpensive Built-in Self-Test (BiST) is given to Test Engineering.

(4) The list of positive UiCs, which could beneficially affect other systems, is given to Marketing, because discovering positive unintended consequences could provide substantial additional revenue.

And finally our original target,

(5) The list of negative UiCs, which could adversely affect other systems, is given to Management and the company lawyers.

Diogenes puts these prioritized lists in the project process assets library (PAL).

Case for change

The United States of America has problems such as global warming, rising national debt, poverty, illegal immigrants, ineffective educational systems, terrorist threats and corruption and greed by officers in our financial institutions. Many of these problems are of our own doing. Quite often, a government organization makes rules or regulations that unknowingly have harmful unintended consequences (UiCs). However, it is possible to create a system that will help decision makers to discover potential negative unintended consequences of their decisions. Diogenes is such a system. Thwarting negative UiCs will give Americans a safer and more pleasant living environment and it will reduce the national debt. Politicians will be able to scathingly deride negative UiCs of legislation introduced by the opposing party.

In a traditional company, lawyers, ethics committees and shareholders would be the most concerned with negative UiCs. A program manager might say, “Why should I care if my system adversely affects another system? That will not affect my bottom line.” But Taguchi said that it is not good enough to be within tolerances: the product should be right on target. If the product is off of its target, then something is wrong and it should be fixed. Taguchi called the penalty of this unknown wrong a cost to Society. The program manager might say that the harmed system has no power over him and his system. He may be right, but the Customer and the Customer’s Customer do have that power; although their effects are not direct. However, as systems become more complex and globally interconnected, the health of the whole ecosystem will affect the program manager’s bottom line. Therefore, it *is* in the best interests of the program manger to avoid negative UiCs.

You get something for nothing. With Diogenes, while you search for negative unintended consequences, you can also find defects, risks, opportunities for BiST and positive unintended consequences.

Victims

Will Diogenes itself have UiCs? If people find out that politicians could, but do not look for UiCs of the laws they pass, the people might vote them out of office.

On May 24, 2010, the Dow Jones industrial average dropped 1000 points in one hour. No one knows how that could have happened, except for the people who studied potential UiCs and made lots of money on the aberration.

Disciples of Machiavelli

Linked cause and effect

The desire to understand UiCs is old, as is documented in this 600-year-old poem.

For want of a nail, a shoe was lost

For want of a shoe, a horse was lost

For want of a horse, a rider was lost

For want of a rider, a message was lost

For want of a message, a battle was lost

For want of a battle, a kingdom was lost

All for want of a nail

Examples of unintended consequences

The federal government tried to reduce our dependence on Middle East oil: so in the last few decades, they spent vast amounts of money to help people grow and ferment corn into alcohol. With the federal subsidies, this process was economically successful. However, it drove the price of corn so high that Mexican peasants could no longer afford to buy tortillas, their staple of life.

In 2007, Congress passed an energy bill that placed stringent efficiency requirements on light bulbs in an attempt to replace incandescent bulbs with more expensive but more energy-efficient compact fluorescence light (CFL) bulbs. Politicians said it would save consumers money, increase domestic security, reduce greenhouse gas emissions and stop global warming. But the politicians failed to see the unintended consequences. For example, CFLs use mercury and if the bulbs are broken people will be exposed to mercury vapor. CFLs are too big to fit into many light fixtures. CFLs do not work with dimmer switches. The lifespan of CFLs diminishes when turned off and on frequently. And the manufacturing of light bulbs has now been outsourced to China.

Local habitats and species have suffered from the introduction of exotic animals and plants for food, for decorative purposes, or to control unwanted species, which often leads to more harm than good. For example, Kudzu was introduced into the United States in 1876, as a forage crop and an ornamental plant. Later the US government encouraged farmers in the southeast to plant kudzu to reduce soil erosion. It was subsequently discovered that the southeastern US has near-perfect conditions for kudzu to grow out of control. Kudzu was named a pest weed by the US Department of Agriculture in 1953.

Asian carp were imported into the United States in the 1970s to remove algae from catfish farms. They have spread rapidly along the Mississippi, Ohio and Illinois rivers where they are crowding out native species. Conservationists are struggling mightily to prevent them from spreading into the great lakes.

In 1997, (in order to protect the Cactus Ferruginous Pygmy Owl around Tucson, Arizona) the U. S. Fish and Wildlife Service under the Endangered Species Act of 1973, proposed designating 1.2 million acres of largely undeveloped land as part of the owl’s critical habitat. After the designations were published in December 1998 but before the regulations took effect in August 1999, land identified as likely critical habitat parcels lost 22% of their value. Furthermore, critical habitat parcels were developed on average about one year earlier than similar non-critical habitat parcels. A likely conclusion is that when people saw the value of their land drop, they decided to develop their land quickly, before the new rules could “take” their land. This undoubtedly reduced the number of Pygmy Owls.

Buffelgrass was introduced intentionally into southern Arizona in the 1940s as cattle forage and to control erosion. Today buffelgrass grows densely and crowds out native plants of similar size. Competition for water can weaken and kill larger desert plants. Dense roots and ground shading prevent germination of native plant seeds. It appears that buffelgrass can kill most native plants by these means alone. However, buffelgrass also makes the desert more fire prone. In 2005, Arizona statute listed buffelgrass as a Regulated and Restricted Noxious Weed. The severity of the threat has prompted actions by individuals and institutions. A variety of control methods have been used, including pulling by hand and with machines, and spraying an herbicide from helicopters and trucks.

Rabbits were introduced into Australia in 1859 as a game animal for hunting. This was followed by an explosive growth in the rabbit population; their 200 million rabbits have become a major feral pest.

In early 1970s, Nestles promoted the use of baby formula instead of breast-feeding in third world countries. Lack of clean water and refrigeration caused lots of baby deaths by dehydration and water born diseases.

Brazilian researchers imported honey bees from Africa in 1956 in an effort to produce a honeybee better suited to the South American tropics. They were successful. Unfortunately, Africanized honeybees are aggressive. If the bees feel that their colony is threatened, large numbers may sting people, pets and livestock.

The University of Wisconsin at Green Bay learned that the Century Gothic font used about 30% less ink than Arial. So in the spring of 2010, they switched the university default printer font from Ariel to Century Gothic. They expected to save $10,000 per year. However, they later learned that the Century Gothic font is wider. So that a document that is one page in Arial could extend onto a second page if printed in Century Gothic. It is not known if the university saved money or saved the environment.

Requiring strong passwords for access to computer systems causes many users to write their passwords on paper (as they are now impossible to remember), negating the security advantage of strong passwords.

LED traffic signals are seemingly ubiquitous these days. They use less electricity, last longer and are highly directional. These factors save money, energy and labor. But there is a negative UiC. They do not produce enough heat to melt the snow off of the lights. As a result, drivers cannot see whether the light is red, green or yellow.

If a person has a bacterial infection, then his or her doctor might prescribe a treatment of antibiotics. If that works, then the antibiotics will kill the bad bacteria and the patient will be fine. However, there may be UiCs. If the antibiotic dose is not strong enough or the duration is too short to kill all of the bacteria, then the strongest will survive and after they are flushed out of the patient’s body, then may mutate into antibiotic resistant bacteria. This is a serious negative UiC. Another negative UiC is that the antibiotic also kills good bacteria that inhabit the patient’s gut. These good bacteria kill bad bacteria such as *Clostridium difficile*. If this happens, the patient usually gets diarrhea. Then to kill the *Clostridium difficile* the doctors give the patient, vancomycin, an antibiotic, etc.

tittipu

The Dow Jones industrial average dropped 1000 points in one hour on May 24, 2010. No one knows exactly how that could have happened, except for the people who studied UiCs and made lots of money on the aberration.

An article in The New York Times on September 30, 1999 stated, “In a move that could help increase home ownership rates among minorities and low-income customers, the Fannie Mae Corporation is easing the credit requirements on loans that it will purchase from banks and other lenders… Fannie Mae, the nation’s biggest underwriter of home mortgages, has been under increasing pressure from the Clinton Administration to expand mortgage loans among low and moderate income people… subprime borrowers [Holmes, 1999].” Later, Wall Street and the bankers found a way to disguise these subprime mortgages by mixing them with good loans. The UiCs of all this was the Great Recession of 2008, where most of the people lost half of their retirement funds, but a few people made a lot of money

Not all unintended consequences are undesirable.

Not all UiCs are undesirable. For example in the 1960s, scientists at 3M produced glue that did not stick very well. The unintended consequence was the invention of the ubiquitous Post-it® notes.

In the 1950s, IBM could have bought the patents for Xerox’s photocopy machine very inexpensively. But they did a market research study and concluded that no one would pay thousands of dollars for a machine that would replace carbon paper. The unintended consequence of the copy machine was that they could delight their customers with a machine that provided dozens of copies in just minutes.

Wolves were reintroduced into Yellowstone Park in the 1990s. This caused a sequence of UiCs. Elk became skittish and stopped eating young aspens, cottonwoods and willows; beavers, feeding on flourishing willows, recolonized streams; reestablished stream side vegetation retarded erosion and provided cover for birds, fish and invertebrates; wolf kills supported more scavengers; pronghorn antelope lost fewer calves to coyotes;

Cause and effect analysis

So how should one go about creating a search for UiCs process? The most appropriate existing tools might be those for cause and effect analysis.

A cause makes something else happen. To find a cause we ask, "Why did that happen?" An effect happens because of a cause. To determine an effect we ask, "What happened?" Temporally, the cause comes before the effect.

Traditionally, the purpose of a cause and effect analysis (a. k. a. causal analysis, root cause analysis and Ishikawa fishbone diagrams), is to find the root cause of an event. The event (usually a failure) has already occurred and we are trying to find its cause. To determine a cause or to move backward in time we ask *why*? “Why did this happen?” Whereas, a failure modes and effects analysis (as commonly used in risk analysis) tries to predict the future. To determine an effect or to move forward in time we ask *what*? “If a certain failure mode were to manifest itself, what would be the effects?” “What if this happened?” “What problem could that cause?” How is similar to What. “How could that harm us?”

The following tools were designed for cause and effect analysis, to help find the root cause of a problem. Hence, they are written in the past or present tense. But they could be rewritten in the future tense to help find UiCs.

A very simple cause and effect analysis merely asks, Why? Why? Why? Why? Why? For example, Why did the space shuttle Challenger blow up? Because a hot flame (6000 ºF) from the side of its right solid fuel rocket booster melted its structure. Why did the flame spring from the side of solid fuel rocket booster? Because the O-rings failed to seal in hot gasses from the rocket motor. Why did the O-rings fail to seal in the hot gasses? Because the O-rings were brittle. Why were the O-rings brittle? Because the temperature was too cold. Why did management launch the shuttle if it was too cold?

To show the difference between a cause and effect analysis looking back in time and a forward-looking search for UiCs, let us consider the Ariane missiles. The Ariane 4 missile was successful in launching satellites. However, the French thought that they could make more money if they made this missile bigger. So they built the Ariane 5. It blew up on its first launch destroying a billion dollars worth of satellites [Kunzig, 1997; Bahill and Henderson, 2005]. The French engineers did not search for UiCs, but if they had, it might have gone like this. The Ariane 5 is larger than the Ariane 4, but it uses the same software. So, what harm could that do? The software could get into an untested state. How could that happen? Being bigger, the missile will accelerate faster. So, what is the untested state? It will gain altitude quicker and will reach the attitude where it tilts from vertical to flight that is more horizontal earlier. How would that affect us? In this new state, there would be much more horizontal acceleration than in the Ariane 4. What problem could that cause? If the horizontal acceleration were still being integrated into horizontal velocity, then the 32-bit horizontal-velocity storage register would overflow. What harm would that do? If the horizontal velocity were still being used, then the CPU would encounter a fault. What would that do? That CPU might turn itself off, even though its backup CPU was already turned off. What would happen then? It would fly out of control and the range officer would have to blow it up.

System failures and UiCs are often caused by failure to identify or test for inappropriate interactions between systems [Bahill and Henderson, 2005] or failure to test functionality in certain states [Botta, Bahill and Bahill, 2006]. Part of the reason for this is the amount of work that would be entailed.

A cause and effect analysis can be put into a tree representation, as shown in figure 1-1 for the sinking of the RMS Titanic [*Titanic,* 1997; Hooper, Foecke, Graham and Weihs, 2003; Bahill and Henderson, 2005].



Figure 1-1. Cause and effect analysis for the sinking of the Titanic

A more formal root cause analysis method is Juran’s [1989] method described here.

1. Analyze Symptoms: A symptom is the outward observable evidence of the effect of a failure. By analyzing these symptoms, we can understand the nature and extent of the problem to be solved.

2. Formulate Theories: A search for the root cause should speculate about possible causes of the problem but never jump to a solution.

3. Test Theories: Before accepting any theory as true, a root cause analysis should systematically test it with data.

4. Identify Root Causes: A root cause is the source of the problem, which, when removed, will sharply reduce or eliminate the deficiency

For the RMS Titanic, the main symptom was that the ship sank rapidly. This then is the effect that we want to find a root cause for. Some theories that had been investigated include (1) low strength of steel in the hull plates and rivets due to poor quality, cold water (-2 ºC) or under design, (2) the ship was in the wrong place at the wrong time due to navigation decisions, poor visibility, high speed and poor look-out training, (3) the ship broke in half due to poor design of such an unprecedentedly large ship. Data would have to be collected for all alternative theories. The discovery of the Titanic wreckage at a depth of 3700 meters in 1985 allowed several theories to be tested. Simulations using finite element analysis showed that flooding of the bow section caused stresses above the material yield stress around the second expansion joint. This caused the Titanic to break in half. Metrological examination and testing showed that the hull plates were of the best steel available at the time. However, the quality of the wrought iron in the rivets, particularly those in the starboard bow section were of very poor quality [McCarty and Foecke*,* 2008].

Ishikawa cause and effect diagrams

Ishikawa cause and effect diagrams were proposed in the 1960s by Kaoru Ishikawa, who pioneered quality management processes in the Kawasaki shipyards. His first step was to identify possible causes of failure, which were traditionally grouped into these major categories: manpower, methods, machines, materials, measurements and the environment.

Ishikawa diagrams are called fishbone diagrams, because they are drawn to look like the skeleton of a fish [Ishikawa, 1990]. See figure 4-4. The head of the fish contains the problem statement. The main bones are the primary probable causes of the problem. Each of these bones can have smaller bones attached, representing subcauses. To create a fishbone diagram do the following steps.

1. Write a problem statement (effect). Put its label in a box on the right and draw a horizontal arrow (the backbone) running to it.

2. Draw four to six main bones. These are labeled with the names of the major categories of probable causes.

3. Use brainstorming to add bones to the diagram. Draw arrows to show cause and effect relationships. Be sure to identify causes rather than symptoms. About each cause ask, “Why did this happen?” Write subcauses branching off the causes. Continue to ask “Why?” and generate deeper levels of causes.

4. Prioritize [Botta and Bahill, 2007] and identify the most important causes.

5. Take corrective action.

6. Verify through testing.

Document 2: Customer Requirements

The system requirements are discovered in the use cases of document 6 (Daniels and Bahill 2004). They are formalized in document 2. Test engineers can use these requirements immediately to start writing test cases and test procedures. Finally new requirements are derived based on the customer requirements, the mission statement and the concept of operation (Bahill and Dean 2009) and they are put into document 3 (Derived Requirements).

**CuR0-1** Diogenes shall be capable of accepting input of MSOffice 2010 files (not Linux).

**CuR0-2** Diogenes shall operate on IBM compatible personnel computers (not Macintosh).

**CuR0-3** For requirements, Diogenes shall be compatible with Excel and DOORS (not Access).

**CuR1-1:** BICS shall write a sales brochure that includes a case for change. DeriveReqt: FR1-1 of Sell Diogenes use case. Notes: DeriveReqt means that this requirement is derived from the indicated requirement in the use cases. Names of use cases are being set in the Verdana font.

**CuR1-2:** BICS shall send letters to the department heads of Risk Management and Quality Assurance of Potential Customers. DeriveReqt: FR1-2 of Sell Diogenes use case.

**CuR1-3:** BICS shall be capable of communicating with Customers by phone, letter and e-mail. DeriveReqt: FR1-3 of Sell Diogenes use case.

**CuR1-4:** BICS shall be capable of negotiating contracts, arranging schedules and billing. DeriveReqt: FR1-4 of Sell Diogenes use case.

**CuR1-5:** BICS shall create short-course materials (PowerPoint presentations and handouts). DeriveReqt: FR1-5 of Sell Diogenes use case.

**CuR1-6:** Brain Trust members shall be trained to deliver the short course. DeriveReqt: FR1-6 of Sell Diogenes use case.

**CuR1-7:** BICS shall develop, validate and summarize course evaluation questionnaires. DeriveReqt: FR1-7 of Sell Diogenes use case.

**CuR2-1** Diogenes shall execute Built-in Self-Tests (BiST). (Derived from BICS company policy.)

**CuR2-2** Diogenes shall be capable of reading and writing the project PAL. DeriveReqt: FR2-2 of the Search for Unintended Consequences use case.

**CuR2-3** Diogenes shall have cause and effect tools (in both tree and fishbone formats) that have been modified for making UiCs diagrams. DeriveReqt: FR2-3 of Search for Unintended Consequences use case.

**CuR2-4** Diogenes shall have the capability of creating and maintaining five databases: the defects, risks, opportunities for BiST, positive untended consequences and negative unintended consequences of SystemZ. DeriveReqt: FR2-4 of Search for Unintended Consequences use case.

**CuR2-5** Diogenes shall have tools for prioritizing lists. DeriveReqt: FR2-5 of Search for Unintended Consequences use case.

**CuR3-1** The Moderator shall form the Inspection Team. DeriveReqt: FR3-1 of Perform Formal Inspection use case.

**CuR3-2** The Moderator shall collect the inspection work products and other relevant material and distribute them to the Inspection Team TBD days before the inspection. DeriveReqt: FR3-2 of Perform Formal Inspection use case.

**CuR3-3** The Moderator shall chair the overview meeting. DeriveReqt: FR3-3 of Perform Formal Inspection use case.

**CuR3-4** Each member of the Inspection Team shall examine the work products prior to the actual inspection meeting looking for defects, risks, opportunities for BiST, positive unintended consequences and negative unintended consequences of SystemZ. DeriveReqt: FR3-4 of Perform Formal Inspection use case.

**CuR3-5** Each member of the Inspection Team shall record and report the number of hours he or she spent inspecting the materials. Typically, this will be two hours. DeriveReqt: FR3-5 of Perform Formal Inspection use case.

**CuR3-6** The Moderator shall conduct the inspection meeting. DeriveReqt: FR3-6 of Perform Formal Inspection use case.

**CuR3-7** The Recorder shall create and maintain five databases that contain defects, risks, opportunities for BiST, positive untended consequences and negative unintended consequences of SystemZ. DeriveReqt: FR3-7 of Perform Formal Inspection use case.

**CuR3-8** The Moderator and the Systems Engineer shall consolidate and edit the five databases to create five prioritized lists. DeriveReqt: FR3-8 of Perform Formal Inspection use case.

**CuR3-9** The Systems Engineer shall deliver the five lists to their respective owners. **Stipulation** Each owner will know what to do with his list. DeriveReqt: FR3-9 of Perform Formal Inspection use case.

**CuR3-10** Diogenes shall put these five prioritized lists in the project process assets library (PAL). DeriveReqt: FR3-10 of Perform Formal Inspection use case.

**CuR3-11** The Moderator shall verify that all fixes are effective and that no additional defects have been created. The Moderator shall check the exit criteria for completing of an inspection. DeriveReqt: FR3-11 of Perform Formal Inspection use case.

**CuR3-12** The Moderator shall schedule the inspection meeting for two hours. The Moderator shall prepare two dozen pages of documentation for each inspection. DeriveReqt: NFR3-1 of Perform Formal Inspection use case.

Mandatory requirements

This mandatory requirements section contains the Key Decisions of document 1 as well as some additional requirements. Mandatory requirements specify necessary and sufficient capabilities, use the verb *shall,* are passed or failed with no in between and should not be included in tradeoff studies. An example is “The system shall not violate federal, state or local laws.”

Mandatory requirements are not tested only once at total system test. They are monitored continually. They are mentioned at every design review. Because they are so important, some mandatory requirements will become technical performance measures (TPMs) (Bahill and Dean 2009).

Document 3: Derived Requirements

Each requirement must be verifiable by (ordered by increasing cost) logical argument, inspection, modeling, simulation, analysis, test or demonstration (Bahill and Dean 2009). Here are dictionary definitions for these terms.

Logical argument: a series of logical deductions

Inspection: to examine carefully and critically, especially for flaws

Modeling: a simplified representation of some aspect of a system

Simulation: execution of a model, usually with a computer program

Analysis: a series of logical deductions using mathematics and models

Test: applying inputs and measuring outputs under controlled conditions (a laboratory environment)

Demonstration: to show by experiment or practical application (flight or road test). However, some sources say demonstration is less quantitative than test. Demonstrations can be performed on electronic breadboards, plastic models, sterolithography models, prototypes made in the laboratory by technicians, preproduction hardware made in the plant using developmental tooling and processes, and production hardware using full plant tooling and production processes.

The attributes of the following requirements are listed row by row, because this is a Word document. If this were a spreadsheet, they would be listed column by column.

Functional Requirements

**FR0-1** Diogenes shall be capable of accepting input of MSOffice 2010 files.

**FR0-2** Diogenes shall operate on IBM compatible personnel computers (not Linux).

**FR0-3** For requirements, Diogenes shall be compatible with Excel and DOORS (not Access).

**FR1-1:** BICS shall write a sales brochure that includes a case for change.

**FR1-2:** BICS shall send letters to the department heads of Risk Management and Quality Assurance of Potential Customers.

**FR1-3:** BICS shall be capable of communicating with Customers by phone, letter and e-mail.

**FR1-4:** BICS shall be capable of negotiating contracts, arranging schedules and billing.

**FR1-5:** BICS shall create short-course materials (PowerPoint presentations and handouts).

**FR1-6:** Brain Trust members shall be trained to deliver the short course. DeriveReqt: FR1-6 of Sell Diogenes use case.

**FR1-7:** BICS shall develop, validate and summarize course evaluation questionnaires.

|  |  |
| --- | --- |
| Identification tag (Id) | FR2-1 |
| Name | Execute BiST |
| Text | Diogenes shall execute Built-in Self-Tests (BiST). |
| Comment | This comes from BICS company policy. |
| DeriveReqt: | CuR2-1 |
| Verify method | During design and construction, this requirement will be verified by test, thereafter it will be verified by weekly demonstration. |
| Priority | High |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Identification tag (Id) | FR2-2 |
| Name | Read & Write PAL |
| Text | Diogenes shall be capable of reading and writing the project PAL. |
| Comment |  |
| DeriveReqt: | CuR2-2 |
| Refined by |  |
| Verify method | Initially inspection, later demonstration |
| Priority | High |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Identification tag (Id) | FR2-3 |
| Name | Cause & Effect Tools |
| Text | Diogenes shall have cause and effect tools (in both tree and fishbone formats) that have been modified for making UiCs diagrams. |
| Comment |  |
| DeriveReqt: | CuR2-3 |
| Refined by |  |
| Verify method | Initially inspection, later demonstration |
| Priority | Medium |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Identification tag (Id) | FR2-4 |
| Name | Creating & Maintaining Databases |
| Text | Diogenes shall have the capability of creating and maintaining five databases: (1) defects in requirements, programming code, test plans and designs, (2) risks that could adversely affect SystemZ, (3) opportunities for inexpensive Built-in Self-Test (BiST), (4) positive UiCs that could beneficially affect other systems and (5) negative UiCs that could adversely affect other systems. |
| Comment |  |
| DeriveReqt: | CuR2-4 |
| Verify method | Test during design and construction, thereafter demonstration: |
| Refined by |  |
| Priority | High |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Identification tag (Id) | FR2-5 |
| Name | Prioritize Lists. |
| Text | Diogenes shall have tools for prioritizing lists. |
| Comment |  |
| DeriveReqt: | CuR2-5 |
| Verify method | Test during design and construction, thereafter demonstration: |
| Priority | Low |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Identification tag (Id) | FRVV-1 |
| Name | Gather V&V Data |
| Text | The system shall facilitate gathering evidence that can be used to prove verification and validation (V&V) of the system and the requirements. |
| DeriveReqt: | Gather Evidence of Verification and Validation use case |
| Verify method | Inspection |
| Priority | High |
| Date of last change | September 25, 2010 |

We can impose requirements on our system, but we cannot impose requirements on operators, pilots and other secondary actors. This is still true. However, in the following requirements we specify the Moderator and members of the Inspection Team. That is all right, because they are a part of Diogenes.

**FR3-1** The Moderator shall form the Inspection Team.

**FR3-2** The Moderator shall collect the inspection work products and other relevant material and distribute them to the Inspection Team TBD days before the inspection.

**FR3-3** The Moderator shall schedule and chair the overview meeting and the inspection meeting.

**FR3-4** Each member of the Inspection Team shall examine the work products prior to the actual inspection meeting looking for defects, risks, opportunities for BiST, positive unintended consequences and negative unintended consequences of SystemZ.

**FR3-5** Each member of the Inspection Team shall record and report the number of hours he or she spent inspecting the materials.

**FR3-6** The Moderator shall conduct the inspection meeting.

**FR3-7** The Recorder shall create and maintain five databases that contain defects, risks, opportunities for BiST, positive untended consequences and negative unintended consequences of SystemZ.

**FR3-8** The Moderator and the Systems Engineer shall consolidate and edit the five databases to create five prioritized lists.

**FR3-9** The Systems Engineer shall deliver the five lists to their respective owners. **Stipulation** Each owner will know what to do with his list.

**FR3-10** Diogenes shall put these five prioritized lists in the project process assets library (PAL).

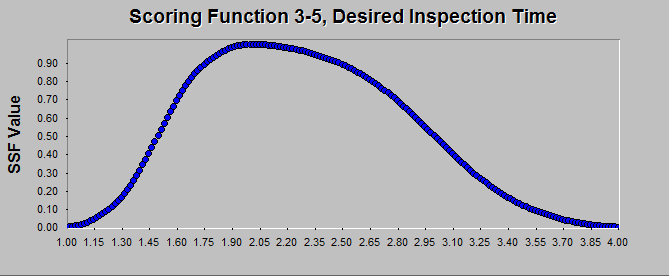
**FR3-11** The Moderator shall verify that all fixes are effective and that no additional defects have been created. The Moderator shall check the exit criteria for completing of an inspection.

**Nonfunctional performance requirements**

**NFPR3-1** The Moderator shall schedule the inspection meeting for two hours. The Moderator shall prepare two dozen pages of documentation for each inspection. Trace to FR3-2 and FR3-3.

**NFPR3-2** Each member of the Inspection Team shall record and report the number of hours he or she spent inspecting the materials. Typically, this will be two hours. Trace to FR3-5.

|  |  |
| --- | --- |
| Identification tag (Id) | NFPR3-2 |
| Name | Desired Inspection Time |
| Text | Before the inspection meeting, each member of the Inspection Team shall spend two hours inspecting the materials in accordance with scoring function 3-5: bitonic hill shaped, LL = 1.0, BL = 1.5, S = 2, MP=2.0, BL2= 3.0, S2=-1, UL = 4.0 |
| DeriveReqt: | CuR3-5 |
| Refines | FR3-5 |
| Verify method | For each inspection, the TestEngineer will record the amount of time spent by each member of the inspection team, compute the average and the standard deviation of the inspection time and present the results to the Brain Trust. |
| Priority | Medium |
| Date of last change | November 8, 2010 |



Test requirements

|  |  |
| --- | --- |
| Id | TR1 |
| Name | Provide questionnaires |
| Text | The system shall provide student course evaluation questionnaires and manager course evaluation questionnaires for use by the course instructor and the Brain Trust. |
| Comment | At the end of each short course, where BICS teaches companies how to use Diogenes, the students will be given a course evaluation questionnaire. Also each student will tell his or her manager what the student learned in the course and give the manager a form. The form will ask questions such as, “Did this student’s learning match your expectations for this course?” This form and the questionnaire will be returned to the instructor. The student and manager responses will be reviewed by the Brain Trust. |
| DeriveReqt: | Document 4b: BiST for Diogenes |
| Verify method | Inspection, look and see if the questionnaires are ready |
| Priority | Medium |
| Date of last change | November 9, 2010 |

**TR2** The system shall provide a forum for the Brain Trust to discuss the student course evaluation questionnaires and the manager course evaluation questionnaires.

**TR3** The system shall provide a complete explanation of Diogenes for use by the TestEngineer.

**TR4** The system shall provide a means for the TestEngineer to schedule and interview members of the Inspection Team.

**TR5** The system shall provide a means for the TestEngineer to record the test results and put them into the PAL.

**TR6** The system shall provide a means for the Recorder to capture data from the inspection and put it into the PAL.

**TR7** The system shall insure that Diogenes is documented, approved and placed among the company processes.

**TR8** Before each inspection, the system shall notify the head of Risk Management, the head of Test Engineering, the head of Marketing, the head of Legal and the Project Manager of the location of Diogenes in the company processes.

Cost Requirements

|  |  |
| --- | --- |
| Id | CoR1 |
| Name | Selling Price |
| Text | The manufacturer’s (BICS) recommended selling price of Diogenes shall not exceed $150,000. |
| Comment | This is still a soft target. |
| DeriveReqt: |  |
| Verify method | Analysis |
| Priority | Medium |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Id | CoR2 |
| Name | Total Lifecycle Cost |
| Text | The total lifecycle cost shall be computed from the design costs, purchase costs, government subsidies, installation costs, operation and maintenance costs, and retirement and replacement costs, in conjunction with the amortization period. |
| Comment | This requires a definition of the design life. A rough suggestion is that companies buy the two-week course and then buy two-day courses annually. |
| DeriveReqt: | Evaluation criteria |
| Verify method | Analysis |
| Priority | Medium |
| Date of last change | September 25, 2010 |

|  |  |
| --- | --- |
| Id | CoR3 |
| Name | Operations Cost |
| Text |  |
| Comment |  |
| DeriveReqt: | Evaluation criteria |
| Verify method | Analysis |
| Priority | Medium |
| Date of last change | October 5, 2010 |

Schedule Requirements

|  |  |
| --- | --- |
| Id | SR1 |
| Name | Due Date |
| Text | The documents describing the design model for Diogenes shall be submitted on or before 3 PM, December 8, 2010. |
| Associated performance requirement | Late projects will be assessed a penalty of 5% per day. |
| DeriveReqt: | Customer’s statement in the SIE-454/554 syllabus |
| Verify method | Inspection |
| Priority | High |
| Date of last change | September 25, 2010 |

Risk Requirements

|  |  |
| --- | --- |
| Id | RR1 |
| Name | Check for Competing Patents |
| Text | Initiate patent search. If no other product is found, then initiate patent application. Otherwise, discuss licensing. Immediately contact BICS’s lawyer and ask if putting this document on my web site constitutes disclosure. |
| DeriveReqt: | Risk analysis done on September 25, 2010 |
| Rationale | If a similar system has already been patented, then we will not be able to market the BIMS. |
| Verify method | Analysis |
| Priority | Medium |
| Date of last change | September 25, 2010 |

Document 4a: Validation of Diogenes using BIMS

To validate Diogenes we will apply it to an existing, well-documented, system design (which will be called SystemZ) and see if Diogenes discovers the defects, risks, opportunities for BiST, positive UiCs and negative UiCs of SystemZ. Let us apply it to the Bahill Illuminance Management System (BIMS) [Bahill, 2010]. Because this system design exists as a document on my web site, we will use MS Word for the implementation. Later we will further validate Diogenes by applying it to the Spin Coach <http://www.sie.arizona.edu/sysengr/sie554/SpinCoach/index.html>. During actual operation, Diogenes will look at all of SystemZ’s documentation. But in this validation document, it will not look at the risk analysis documents of BIMS (because we are trying to show that it can find these risks).

BIMS concept of operations (ConOps)

Our customer needs a light-management system for the operations rooms of telescope enclosures to be built on Mauna Kea in Hawaii and on Cerro Pachón near **Vicuña** in Chile. However, BIMS must be designed so that it can be adapted for other structures. This system will be named the Bahill Illuminance Management System (BIMS).

Astronomers will use the operations room day and night while observing both the Sun and nighttime targets such as stars and galaxies. In the daytime, the astronomers want a constant illuminance of 400 lux. During the night, the astronomers will be continually going in and out of the operations room. There will be no lights outside, because that would interfere with the telescopes. So they want the light inside the operations room to be dim and constant, so that (1) the astronomers do not have to wait minutes for their eyes to dark-adapt and (2) the light inside does not leak out and interfere with the telescopes. Therefore, they want the inside illuminance at night to be 0.4 lux.

BIMS must conserve energy and provide a natural daylight color spectrum. An efficient way of doing this is to have BIMS use daylight instead of artificial lighting as much as possible. BIMS will be politically correct (environmentally green) because it will use renewable-energy electric-generators. This is a political decision, not an economic or scientific decision. BIMS will probably use solar panels to generate electricity. But other power generating alternatives such as wind turbines and geothermal systems near the Kilauea volcano should be considered. BIMS shall include all equipment needed for connecting to the local AC electric power grid.

BIMS risks, design alternatives, most sensitive parameters, test plans and design are given in [Bahill [2010]](http://www.sie.arizona.edu/sysengr/publishedPapers/BIMS.pdf).

Examine BIMS’s use cases

As an example, we will use the following use case model. Suggestions of defects, risks, opportunities for BiST, positive UiCs and negative UiCs will be marked in the Gill Sans MT font. This will be followed by a classifier (defect, risk, BiST, pUiCs or nUiCs) and finally a short name.

**Name:** Control Illuminance During the Day

**Iteration:** 3.6

**Derived from:** Concept of operations

**Brief description:** The sun rises and sets, but the Bahill Illuminance Management System (BIMS) will keep the illuminance in the operations room constant.

**Level:** high

**Priority:** This is of the highest priority.

**Scope:** Theoperations room of a telescope facility on a remote mountaintop, a renewable-energy electric-generator and a connection to the local electric power grid. Put you mind in that location and ask what bad things could happen up here?

(1) The Kilauea volcano could erupt or another volcano could erupt covering the sky with ash and rendering the solar panels useless; risk, Kilauea volcano erupts.

(2) BIMS could offend Poliahu, the snow goddess of Mauna Kea. It is not known how this could happen. But if the native Hawaiians think that Poliahu is upset, then we will have problems. For instance, they have already asked the state of Hawaii change the annual rent from $1 to $50M; nUiCs, BIMS offends Poliahu.

(3) Because Mauna Kea is a remote mountaintop at 13,800 feet, costs will be higher than expected. Transportation is more expensive. Electricity is more expensive. Backup electric generators will be necessary. Labor is more expensive; risk, Geographical location causes higher costs.

**Added value:** Astronomers are more comfortable and more productive

**Goal:** Maintain specified illuminance in the daytime.

**Primary actors:** Astronomer, Engineer, Tester

**Supporting actors:** Sun, Clouds (and during the night the Moon)

**Frequency**: It will be used every day. What will happen when it gets old? There must be a plan and a budget for decommissioning each mountaintop structure at the end of its design life. These costs could be quite surprising. For example, in 1959 the UofA purchased and installed a nuclear reactor at a cost of $150k. It is now being decommissioned at a cost of $2M. nUiCs, Money is needed for decommissioning.

**Precondition:** BIMS has passed all of its Built-in Self-Tests and Tester or the Engineer has started BIMS.

**Trigger:** The sun rises.

**Main Success Scenario:**

1. The sun rises in the morning.

2. BIMS turns up the lights and closes the window screens or curtains.

3. BIMS senses the illuminance in the room with light sensors and adjusts the illuminance with light dimmers and window screens or curtains.

4. The sun moves across the sky and the illuminance from the sun starts to increase. (Actually, the earth rotates, but it more intuitive to say that the sun moves.)

5. BIMS decreases illuminance from the lights and partially opens the window screens or curtains. The tradeoff between these two would be determined by sunlight shining on computer monitors, heating and cooling considerations as well as electricity usage.

6. The sun rises to its zenith.

7. If it will not waste heating or cooling energy, BIMS opens the window screens or curtains.

8. BIMS senses the illuminance in the room with light sensors and adjusts the illuminance with light dimmers and window screens or curtains. Every time BIMS changes the power to the lights or the positions of the window screens or curtains, it should wait one minute and then record the measured illuminance in the room. If this is outside the limits, it should report an error; BiST, Record changes.

9. The sun starts to set.

10. BIMS slowly adjusts the illuminance to its nighttime level. (Due to its complexity, this step will probably become a separate use case in future models.)

11. The sun sets.

12. Include theControl Illuminance During the Night use case.

**Clouds Cover the Sun Unanchored Alternate Flow:**

1. Electric generation falls due to the wind dropping, waves disappearing or clouds covering the sun.

2. BIMS opens the window screens or curtains.

3. BIMS draws energy from the commercial AC electric grid. What problem could this cause? The Hawaii Electric Light Company will have to buy backup power generators that can provide the total load of BIMS at any time. nUiCs, Increased costs to electric company.

4. Electric generation resumes due to the wind increasing, waves coming back or clouds blowing away.

5. BIMS delivers energy to the AC electric grid. How could this activity hurt another system? Incorrect frequency or phase for the connection to the electric grid could harm equipment or destabilize the grid; nUiCs, Improper connection to the grid.

It would not be useful for BiST to display the phase and frequency to the human, because the human is not fast enough to make the connection. The connection must be made by the system. BiST shall record the difference between in phase and frequency when a connection is made and indicate failure when either is outside of TBD limits. BiST

6. BIMS readjusts the light dimmers and window screens or curtains. [return to the main success scenario.]

**Postcondition:** BIMS is in theControl Illuminance During the Night use case.

**Specific Requirements** [Daniels and Bahill 2004]

**Functional Requirements:**

Req1-1 BIMS shall use an ephemeris, tables, models, firmware or similar methods to anticipate sunrise, sunset, moonrise, moonset and the phase of the moon.

Req1-2 BIMS shall control the illuminance of the lights.

Req1-3 BIMS shall control the opening and closing of window screens or curtains.

Req1-4 BIMS shall sense the illuminance in the operations room.

Req1-5 BIMS shall buy electricity from and sell electricity to the AC electric power grid. What could screw up this activity? The commercial electric distribution company could fail to buy or sell electricity, or they could set unfavorable rates. BIMS cost could exceed the local area rate. risk, Electric company policy.

Req1-6 BIMS shall generate electricity. Here are some common examples of renewable-energy generating sources: photovoltaic panels, wind turbines, ocean waves, ocean tides and geothermal systems. What could cause these sources to fail to provide enough energy at the appropriate time? Clouds could cover the sun, the wind could fail, the ocean could come becalmed. risk, Sudden drop in generated electricity.

High elevation and cold temperature might reduce efficiency. risk, Reduced efficiency.

Req1-7 BIMS shall execute Built-in Self-Tests (BiST) (derived from BICS company policy).

**Nonfunctional Requirements:**

Req1-8 BIMS shall maintain the daytime illuminance in the operations room at 400 ± 40 lux (≈40 ± 4 fc). Trace to Req1-2, Req1-3 and Req1-4.

Req1-9 BIMS shall maintain the nighttime illuminance in the operations room at 0.4 ± 0.2 lux (≈0.04 ± 0.02 fc). Trace to Req1-2, Req1-3 and Req1-4.

Req1-10 BIMS shall generate electricity at a cost competitive with commercial electricity costs at that location, after Federal subsidies, etc. Trace to Req1-5 and Req1-6. Changes in interest rates, government policies or TEP practices would change the economic analysis. risk, Economic conditions change.

**Author/owner:** Walt Zaharchuk

**Last changed:** December 3, 2009

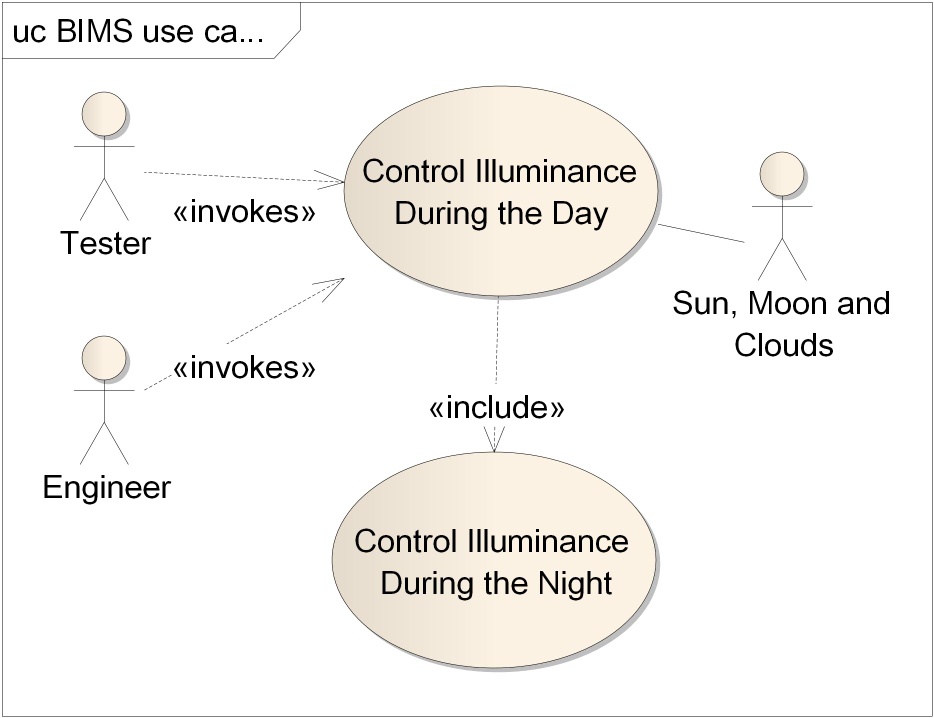


Figure 4-1. A use case diagram for BIMS. A use case diagram is the table of contents of a use case model. It shows all of the use cases that have been described.

Examine BIMS’s physical structure



Figure 4-2. Block diagram showing the scope of BIMS

In figure 4-2, energy flow lines between the control system and the Interface and the Operations Room are missing; defect.

One part of SystemZ is the interface with the AC electric power grid. In keeping with the risk analysis technique of creating a hierarchy of overlapping models, we show a decomposition of this interface in figure 4-3. This particular study was done with Tucson Electric Power company, not the power company on Hawaii, because we live in Tucson. Also figures 4-3 and 4-4 do not come from the BIMS documentation. They come from Chaves and Bahill [2011].

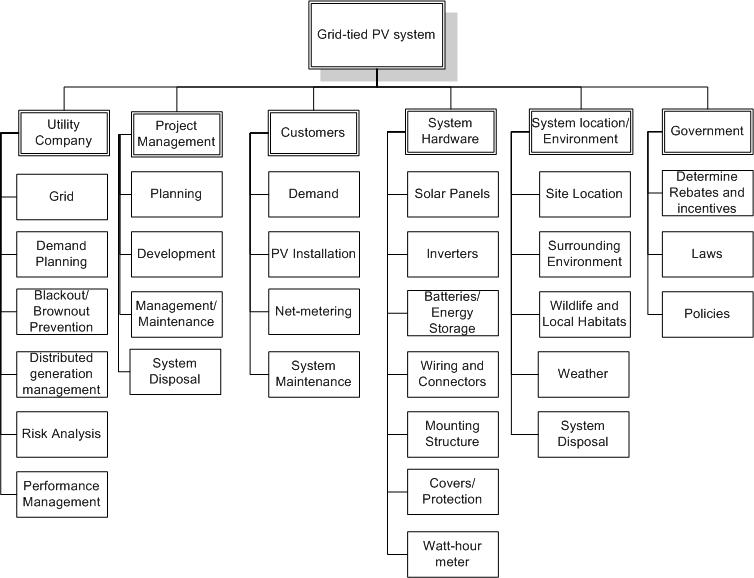


Figure 4-3. A hierarchy of models for grid-tied photovoltaic systems

Any one of these blocks could fail constituting a risk to BIMS. The TEP experts have provided numerical data for failure of the DC to AC invertors on solar panels. This risk has been included in the risk analysis of SystemZ [Chaves and Bahill, 2011]. risk, DC to AC invertors fail.

Cause and effect tools

We will now create an Ishikawa fishbone diagram for the example of incorporating photovoltaic solar panels into an existing commercial electric power grid. Suppose that the event of interest is that Tucson Electric Power (TEP) might fail to meet its load demand. For this event, the major categories of possible causes are performance, environment, project management, economics and government actions [Chaves and Bahill, 2011]. We will now break these down into subcauses and subsubcauses. In the following table, we will do this in both directions (cause and effect), but in the fishbone diagram we will only show the causes direction.

|  |  |
| --- | --- |
| **Performance Causes** | **Causes and Effects** |
| Solar panel output power changes 60 MW in 15 minutes | *Cause:* Clouds cover a large area of solar panels, the clouds were produced by monsoon thunder storms  *Effect:* Not enough power is produced. Grid voltage could drop, frequency could drop and breakers could trip. TEP might have to initiate a controlled brownout with load shedding. Customers could lose trust in TEP. To ameliorate these potential problems TEP must buy and operate backup generators and negotiate purchase agreements with other electric suppliers. |
| Out of phase power peaks | *Cause:* solar panel output peaks at noon, but customer load demand peaks at 5 PM.  *Effect:* This creates a need for extra generating or storage capability. |
| Short to ground on the distribution grid | *Cause:* snakes, raptors, trees, wind, automobile crashes  *Effect:* damage to transformers and capacitor banks |
| Lightning strikes BIMS | *Effect:* components are damaged, system could crash |
| Individual solar panel output decreases | *Cause:* Solar panels accumulate layers of dust  *Effect:* Power production would decrease. |
| Failure of the main power generators | *Effect:* grid could crash |
| Failure of backup generators | *Cause:* backup motor-generator (MG) sets could take longer than 10 minutes to start up, software failure, MG sets connect to the grid at the wrong phase or frequency, or violent storms could damage the MG sets  *Effect:* brownout and load shedding |

|  |
| --- |
| **Environment Causes** |
| Interference with native Indian religious ceremonies |
| Modification of animal migration paths |
| Destruction of natural habitats or riparian areas |
| Higher than expected disposal or recycling cost |
| Infrastructure damage from violent storms or flooding |
| **Project Management Causes** |
| Accidents causing injury to humans requiring medical attention |
| Drastic human mistakes resulting in human fatalities |
| Project costs become higher than projected |
| Maintenance costs become higher than expected |
| **Economic Causes** |
| Interest rates change |
| Cost of solar panels changes |
| **Government Action Causes** |
| Carbon emission regulations change |
| Early elimination of rebates |



Figure 4-4. Ishikawa fishbone diagram for incorporating photovoltaic solar panels into an existing commercial electric power grid. Obviously, each item in this diagram will be described with sentences and paragraphs elsewhere in the documentation.

Let us now use the fishbone diagram of figure 4-4 to look for UiCs of incorporating photovoltaic solar panels into an existing commercial electric power grid.

We will start with economic causes. What if new technology dramatically drove down the cost of solar panels? This would increase the number of customers who install solar panels. The electric company would have to increase their backup capacity in order to handle customer peak load demands during the period around 5 PM in spite of total cloud coverage. During the day these customers would buy less electric energy from TEP and on sunny days TEP would be required to buy the surplus electricity produced by these customers. This would affect TEP’s bottom line: they would lose revenue. Two things could then happen, TEP could lose money from decreased revenues and increased net-metering costs or TEP could substantially reduce net-metering payments and rebates. This would eliminate incentives for residential customers to acquire photovoltaic solar panel systems. This is an unintended negative feedback loop. nUiCs, Destabilizing the solar panel economy.

Now let us apply Juran’s cause and effect analysis to the example of incorporating photovoltaic solar panels into an existing commercial electric power grid, we might find as a main symptom that sporadically TEP could not meet customer demands. An explanatory theory could be that variations in sunlight intensity could be changing the amount of electricity that was being delivered by photovoltaic solar panels. We could collect sun intensity data for a typical 24-hour day and also for yearly variations. These could be compared to TEP’s typical load demands. These data could confirm that variations in sun intensity would make electric power regulation difficult. The root cause of the difficulty would be clouds covering the sun; risk, Sudden drop in generated electricity.

Another method for searching for UiCs is the “what-if” analysis. In a “what-if” analysis, speculate on what could go wrong in each state of BIMS. Ask, What if this hypothetical event occurred? What would be the effects on BIMS? For example, what would be the effects of incorporating photovoltaic solar panels into an existing commercial electric power grid? Photovoltaic solar panels transform sunlight it into electricity and reflect sunlight back into the atmosphere. Therefore, photovoltaic solar panels prevent sunlight from hitting the ground and being absorbed. This reduces the amount of energy absorbed by the Earth and therefore contributes to *global cooling;* pUiCs, Global cooling.

We will now look at a SysML block definition diagram. For each block we will ask, “What unexpected events could this entity cause?”

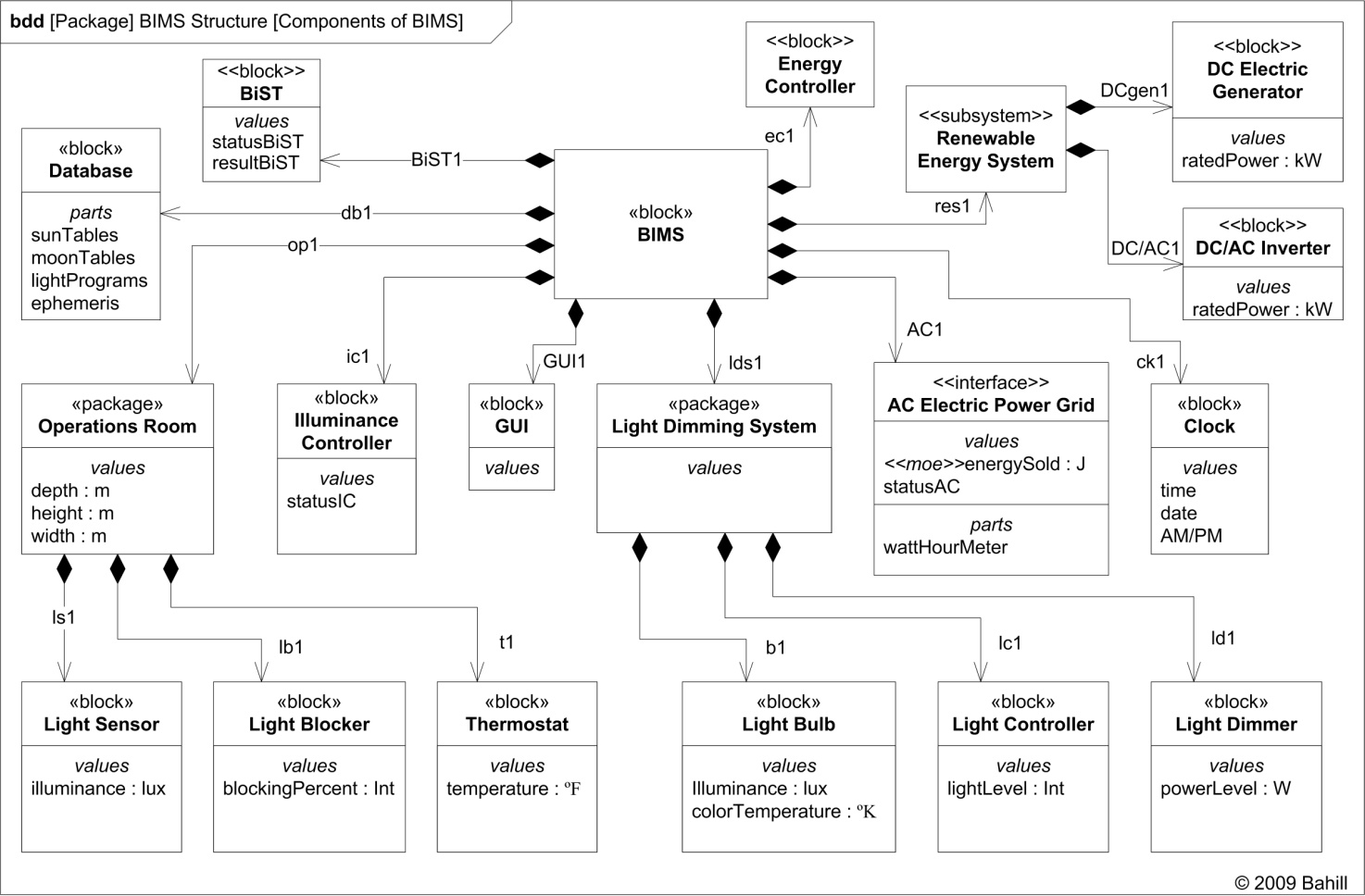


Figure 4-5. SysML block definition diagram (**bdd**) showing the structure of BIMS.

Studying the blocks of figure 4-5, prompted the following question. What nonvisible electromagnetic radiations could interfere with particular telescopes? To ameliorate this problem, the spectrum of each telescope must be determined and the noise emissions of each component will have to be computed and measured in each of these bandwidths; nUiCs, Electromagnetic radiation interfering with telescopes.

The hierarchy of models of Haimes [2009], the fishbone diagrams of Ishikawa and the block definition diagrams of SysML express the same concept: namely, produce a wide collection of entities for the experts to look at. This will help them to discover defects, risks, opportunities for BiST, positive UiCs and negative UiCs.

Commercial off the Shelf (COTS) software will be used to predict, on a minute-by-minute basis, the amount of electric energy that will be bought from or sold to the Hawaii Electric Light Company. BIMS will compute the amount of electric energy that is actually bought from or sold to the Hawaii Electric Light Company. These data are stored in the database. If the daily averages differ by more than plus or minus two standard deviation, then BiST will trigger an alarm. BiST

Examine BIMS’s behavioral models



Figure 4-6. State machine diagram (**stm**) for the BIMS Illuminance Controller. The illuminance limits could be parameters instead of fixed values. The illumination controller increases or decreases the illuminance by commanding a different lightLevel and blockingPercent. Running the sunrise program is accomplished by commanding a sequence of preprogrammed lightLevel and blockingPercent. BIMS will have many built in illumination programs, such as a sunset program, a sunrise program, a program for watching PowerPoint presentations.

How would precisely controlled illuminance over a long period of time affect humans? The effects of high altitude on human physiology are well known and they have been managed effectively for decades on Mauna Kea. However, humans are not used to living in a precisely controlled illuminance environment as described in figure 4. Therefore, studies of the Polaris ballistic missile fleet sailors should be reviewed to see if such a regulated illuminance environment would cause undesirable entrainment of human circadian rhythms. risk, Controlled illuminance harms humans.

  
Figure 4-7. State machine diagram (**stm**) for the BIMS energy management controller

What if BIMS is in the state of Storing Energy when input-port 1 signals *less* and simultaneously input-port 2 signals *full*? Similarly, What if BIMS is in the state of Using Stored Energy when input-port 1 signals *more* and simultaneously input-port 2 signals *empty*? Actually, this is not a serious problem, because the logic can be designed to prevent transitions to unwanted states; risk, Hazards and races.

The initial pseudo states should transition to Using Stored Energy not to Buying AC Electricity. There is an unstated assumption the Hawaii Electric Company is always there for us, i. e. it is an infinite source or sink. The equation for neither should be

I2 = neither, means 5% ≤ percentFull ≤ 95%.

The BIMS energy management controller of figure 4-7 can be modeled with these Boolean equations

Buy = less And empty

Sell = Not less And full

Use = less And Not empty

Store = Not less And Not full

Therefore, it is a static problem. A static problem does not require a dynamic solution. We can keep the state machine diagram, if we think that is the best way to communicate our design. But we could also represent this behavior more simply with Boolean equations. By the principle of Occam’s razor the simpler model is better, if they model the problem equally well. Although the designer may claim otherwise, this actually was a mistake; defect, State machine not necessary.

Replacing the state machine with Boolean equations eliminates this risk.

BiST philosophy. A long time ago, BiST used a single light that indicated go/no-go for the whole system. Now we expect each component to display not only go/no-go but also some intermediate form of system health. For example, for the state machine diagram of figure 4-7, the system could have binary indicators for the inputs “more” or “less” and “full” or “empty,” and the outputs “storeEnergy” or “useStoredEnergy” and “buyACelectricity” or “sellACelectricity.” But we think it would be just as easy and more meaningful to display the attribute “percentFull” and the state diagram of figure 4-7 indicating which state the system is in. Providing state information would also help with developmental testing.

The electric industry has many models for the generation, transmission and use of electricity generated from renewable energy sources [GE Energy, 2010]

Examine BIMS’s interfaces and interconnections

Look for interfaces and interactions when components are integrated together. The biggest potential interface problem is the interconnection between BIMS and the Hawaii Electric Light Company AC electric power grid.

What problems could the interconnection of systems cause? Presume that BIMS is in the state of Selling AC Electricity in the state machine of figure 4-7, when clouds suddenly cover the sun (or the wind fails, or the waves disappear, etc.). The voltage generated by the solar panels will drop as will the illumination in the operations room. The sensors will sense this drop in illumination and will command the lights to produce more illuminance. The lights will draw more power from the source. This will produce a bigger voltage drop across the source internal impedance, which will further drop the operating voltage. This is a positive feedback loop that could cause BIMS to become unstable. Furthermore, BIMS will soon deplete its small local energy store and will switch to the Buying AC Electricity state. This will increase the operating voltage. This is a negative feedback loop, but it contains a significant time delay. Time delays make systems susceptible to instabilities. Because of these potential stability problems, we would recommend that the project manager start a detailed simulation of these systems to investigate potential instabilities. nUiCs, Destabilizing the electric grid.

How could changes in government regulations affect BIMS? Changes in carbon emissions policies would have an impact on the viability and size of photovoltaic systems. Policy changes would make the electric utility’s renewable energy portfolio plan obsolete and would require re-planning of their strategies. The early elimination of rebates is another government-induced risk. It would affect customer incentives to convert to solar-powered generation. Any reduction in consumer incentives to adopt solar energy would have a significant impact on distributed electric generation. nUiCs, Destabilizing the U. S. energy economy.

Stakeholders

BIMS stakeholders include dealers, architects, contractors, distributors, suppliers, sales people, end users, astronomers, NOAO, university students, a surrogate customer (an in-house person designated to have knowledge of end user needs and expectations), potential victims (such as competing companies, homeowners who get shocked due to faulty wiring, other astronomers on the mountain, construction workers, Hawaii Electric Light Company Inc, the environment and Poliahu the snow goddess of Mauna Kea), environmentalists, the National Science Foundation and the U. S. Congress.

The U. S. Congress and the National Science Foundation could cut off funding for BIMS. risk, Politics change.

The output of “Diogenes applied to BIMS”

The above concerns have been edited slightly after the initial investigation. They were used to produce the five Diogenes databases. Next, in accordance with the Search for Unintended Consequences use case of document 6, these five output lists were edited to produce the following lists.

List of potential risks to BIMS

Kilauea volcano erupts. BICS should ensure that BIMS is connected to the USGS Volcano Hazards Program, at the Hawaiian Volcano Observatory in case we have to evacuate the mountaintop. Furthermore, all solar panel installations must consider the effects of a cloud of ash drifting from an erupting volcano, such as the Eyjafjallajökull volcano in Iceland in 2010.

Geographical location increases cost. Because it is a remote mountaintop at 13,800 feet, costs will be higher than expected. Transportation is more expensive. Electricity is more expensive. Backup electric generators will be needed. Labor is more expensive. The supply chain will be more expensive.

Reduced efficiency. The renewable-energy generating systems might have reduced efficiency due to high elevation and cold temperature. Electronic equipment cooling fans are less efficient at high altitudes and may need to be upgraded.

High altitude affects humans. High altitude affects human mental and physical processes. These are known and they have been managed effectively for decades on Mauna Kea.

Controlled illuminance affects humans. Precisely controlled illuminance over a long period of time may affect human mental and physical processes. Humans are not used to living in a precisely controlled illuminance environment. Studies of the Polaris ballistic missile fleet sailors should be reviewed to see if such a regulated environment would cause undesirable entrainment of human circadian rhythms.

Sudden drop in generated electricity. Clouds covering the sun (the wind failing or the ocean becoming becalmed) could cause a sudden drop in generated electricity. This could trip breakers and leave customers without electric power. Voltage and frequency on the grid could drop. The electric company would have to initiate a controlled brownout with load shedding. To ameliorate these risks, the electric company must buy and operate backup generators

DC to AC invertors fail. DC to AC invertors on the solar panel systems could fail. This is the most common failure mode on present grid-tied photovoltaic solar systems.

Electric company policy. The commercial electric distribution company could fail to buy or sell electricity, or they could set unfavorable rates. BIMS cost could exceed the local area rate. However, federal laws require electric companies to buy electricity from their consumers.

Economic conditions change. Changes in interest rates, government policies or local electric power company rebates would change the economic analysis.

Politics change. The U. S. Congress or the National Science Foundation could cut off funding for BIMS.

Hazards and races. Concerning the state machine diagram for the BIMS energy management controller, What if BIMS were in the state of Storing Energy when input-port 1 signaled *less* and simultaneously input-port 2 signaled *full*? Similarly, What if BIMS were in the state of Using Stored Energy when input-port 1 signaled *more* and simultaneously input-port 2 signaled *empty*? BICS must take precautions in designing the logic to prevent transitions to unwanted states.

This analysis discovered 11 risks for BIMS. The original BIMS risk analysis had 14 risks. This study missed (1) a similar system has already been patented (2) the commercial AC electric power grid may fail for hours at a time and (3) software failures. This study disclosed two risks that the original study did not have: (1) controlled illuminance might harm humans and (2) hazards and races. The numbers do not add exactly, because some risks were combined in one study or the other, some were eliminated or mitigated, some were treated in other sections and some were at different levels of detail. Furthermore, this example only inspected one of BIMS’ use cases.

List of negative UiCs

BIMS offends Poliahu. BIMS could offend Poliahu, the snow goddess of Mauna Kea. It is not known how this could happen. But if the native Hawaiians think that Poliahu is offended, then *we* will have problems. For instance, they have already asked the state of Hawaii change the annual rent from $1 to $50M. Observatories face heightened community scrutiny due to their prominent siting. Proactively seeking accommodation with environmental concerns is one ingredient to a successful project. BIMS must fend off environmental activists who might try to prevent funding and construction of the facility.

Money is needed for decommissioning.There must be a plan and a budget for decommissioning each mountaintop structure at the end of its design life.

Increased costs to electric company. BIMS draws energy from the commercial AC electric grid. Therefore, the Hawaii Electric Light Company will have to buy backup power generators that can provide the total load of BIMS at any time.

Improper connection to the grid. BIMS delivers energy to the AC electric grid. Incorrect frequency or phase while connecting to the electric grid could harm equipment and destabilize the grid.

Electromagnetic radiation interfering with telescopes. Nonvisible electromagnetic radiation could interfere with particular telescopes. To ameliorate this problem, the spectrum of each telescope must be determined and the noise emissions of each component will have to be computed and measured in each of these bandwidths.

Destabilizing the solar panel economy. What if new technology dramatically drove down the cost of solar panels? This would increase the number of customers who install solar panels. The electric company would have to increase their backup capacity in order to handle customer peak load demands during the period around 5 PM in spite of total cloud coverage. During the day, these customers would buy less electric energy from the electric company and on sunny days, the electric company would be required to buy the surplus electricity produced by these customers. This would affect the electric company’s bottom line: they would lose revenue. Two things could then happen, the electric company could lose money from decreased revenues and increased net-metering costs or the electric company could substantially reduce net-metering payments and rebates. This would eliminate incentives for residential customers to acquire photovoltaic solar panel systems. This is an unintended negative *feedback loop with a time delay*, which could cause instability.

Destabilizing the U. S. energy economy. Changes in government regulations will affect BIMS. Changes in carbon emissions policies would have an impact on the viability and size of photovoltaic systems. Policy changes would make the electric utility’s renewable energy portfolio plan obsolete and would require re-planning of their strategies. The early elimination of rebates is another government-induced risk. It would affect customer incentives to convert to solar-powered generation. Any reduction in consumer incentives to adopt solar energy would have a significant impact on distributed electric generation.

Destabilizing the electric grid. (1) Connecting to the AC commercial AC electric grid will cause problems. Presume that BIMS is in the state of Selling AC Electricity in the state machine of figure 5, when clouds suddenly cover the sun (or the wind fails, or the waves disappear, etc.). The voltage generated by the solar panels will drop as will the illumination in the operations room. The sensors will sense this drop in illumination and will command the lights to produce more illuminance. The lights will draw more power from the source. This will produce a bigger voltage across the source internal impedance, which will further drop the operating voltage. This is a *positive feedback loop* that could cause BIMS to become unstable. (2) When the sun is blocked by clouds, BIMS will quickly deplete its small local energy store and will switch to the Buying AC Electricity state. This will increase the operating voltage. This is a negative *feedback loop*, but it contains a significant *time delay*. Time delays make systems susceptible to instabilities. Because of these potential stability problems, we recommend that the project manager start a detailed simulation of these systems to investigate potential instabilities.

Although the use of this negative UiCs list is beyond the scope of Diogenes, it is expected that the negative UiCs will eventually become a part of the risk management process.

This analysis discovered eight negative UiCs of BIMS. The original BIMS documentation had only one, offending Poliahu, which was treated as a risk.

List of positive UiCs

This application of Diogenes to BIMS revealed only one positive UiC, global cooling.

The whole manufacturing, installing and operating process of a typical solar station has a net negative carbon footprint. Manufacturing the solar panels and other hardware for TEP’s Springerville Solar Generating Station consumed 12 MWh AC per kW DC generation capability installed. This facility reduces the carbon footprint by 36.5 tons of CO2 per kW DC installed. Therefore, the energy payback time would be 2.8 years, which is less than the 30 year expected life of this facility. Thus, this facility reduces global warming [Chaves and Bahill, 2011].

What would be the effects of incorporating photovoltaic solar panels into an existing commercial electric power grid? Photovoltaic solar panels transform sunlight it into electricity and reflect sunlight back into the atmosphere. Therefore, photovoltaic solar panels prevent sunlight from hitting the ground and being absorbed. This reduces the amount of energy absorbed by the Earth and therefore contributes to *global cooling*.

List of potential Opportunities for BiST

Commercial off the Shelf (COTS) software will be used to predict, on a minute-by-minute basis, the amount of electric energy that will be bought from or sold to the Hawaii Electric Light Company. BIMS will compute the amount of electric energy that is actually bought from or sold to the Hawaii Electric Light Company. These data will be stored in the database. If the daily averages differ by more than plus or minus two standard deviation, then BiST will trigger an alarm.

A long time ago, BiST had a single light that indicated go/no-go for the whole system: now each component is expected to display its status. However, each component should display not just go/no-go, but rather intermediate values and prognosticators of system health. Furthermore, rather than just system health, BiST could show the status of inputs, outputs, key attributes and system states. The best way to present state information would be to display a state diagram indicating which state the system is in. Providing state information would also help with developmental testing. However, Tester must not be overloaded, so complete BiST information would only be displayed on request or in event of failure.

Every time BIMS changes power to the lights or the positions of the window screens or curtains, it should wait one minute and then record the measured illuminance in the room. If this is outside the limits, it should report an error.

List of defects

This application of Diogenes to BIMS revealed two defects: a lack of energy flows in the block diagram of the scope of BIMS (figure 4-2) and a dynamic solution for a static problem in figure 4-7.

The BIMS energy management controller is a static problem. Therefore it does not require a dynamic solution. We can keep the state machine diagram, if we think that is the best way to communicate our design. But we could also represent this behavior more simply with Boolean equations. Replacing the dynamic solution with a static solution would eliminate the risk of hazards and races. Although the designer may claim otherwise, this actually was a mistake; defect, State machine not necessary.

The next step in validating Diogenes will be applying it to the Spin Coach.

<http://www.sie.arizona.edu/sysengr/sie554/SpinCoach/index.html>

Document 4b: BiST for Diogenes

At the end of each short course where we teach companies how to use Diogenes, the students will be given a course evaluation questionnaire. Also each student should tell his or her manager what the student learned in the course and give the manager a form. The form will ask questions such as, “Did this student’s learning match your expectations for this course?” This form and the questionnaire will be returned to the instructor. The student and manager responses will be reviewed by the Brain Trust.

Document 4c: Verification Plan for Diogenes

Now that we have shown that Diogenes does what it is supposed to do, we want to verify the work products of Diogenes. We cannot wait until the system is in use to collect verification data. And it would be too expensive to ask for a dry run, just to get verification data. So for the following test plan, TestEngineer will interview participants. We will ask the Brain Trust to serve as a surrogate Inspection Team.

This is the beginning of the test of the main success scenario of the Perform Formal Inspection use case

1. **Planning** The Moderator selects the Inspection Team, obtains work products to be inspected from the Author/Designer and distributes them along with other relevant documents to the Inspection Team.

TestEngineer interviews the Moderator to ensure that he or she knows who should be on the team, what type of work products should be chosen, what the entry criteria should be, how long the inspection meeting should last, how long each team member should spend preparing for the inspection, and how much material should be in the inspection package. If the Moderator does not know something, then TestEngineer explains it. This tests **FR3-1,** **FR3-2** and **NFR3-1**. TestEngineer documents this meeting.

2. **Overview meeting** The Moderator explains the inspection process to the Inspection Team. The Author/Designer may describe the important features of the work products.

TestEngineer interviews the Moderator to ensure that he or she knows the purpose of the overview meeting and the inspection process. If the Moderator does not know something, then TestEngineer explains. This tests **FR3-3**. TestEngineer documents this meeting.

3. **Preparation** Each participant examines the work products prior to the actual inspection meeting. Typically, this will take two hours for each participant. The amount of time each person spent will be recorded. Each participant should be looking for five things simultaneously: defects, risks, opportunities for BiST, positive UiCs and negative UiCs of SystemZ.

TestEngineer interviews the members of the inspection team, to ensure that they understand what their responsibilities are, the inspection process, that they are supposed to record and report the hours they spent in preparation, and that they should be looking for five things simultaneously: defects, risks, opportunities for BiST, positive and negative UiCs of SystemZ. If a team member does not know something, then TestEngineer explains it to them. This tests **FR3-4** and **FR3-5**. TestEngineer documents these meetings.

4. **Inspection meeting** The Moderator and Reader lead the team through the work products. The issues are brought up one by one and each one is discussed in a round robin fashion where each member comments on each issue. During the discussion, all inspectors can report defects, risks, opportunities for BiST, positive UiCs and negative UiCs of SystemZ, all of which are documented by the Recorder. The meeting should last no more than two hours.

TestEngineer interviews the Recorder to find out how the Recorder will capture the data from the inspection (paper forms, laptop computer, desktop computer, Excel, MS Word, Access, etc.). TestEngineer ensures that such material will be available in a typical inspection room. TestEngineer examines the Recorder’s access to the PAL. This tests **FR3-6**. TestEngineer documents this meeting.

5. Diogenes creates and maintains five databases that contain defects, risks, opportunities for BiST, positive and negative UiCs of SystemZ.

TestEngineer examines and records the location of the databases. This tests **FR3-7**.

6. The Moderator and the Systems Engineer consolidate and edit the five databases to create five prioritized lists.

The list of defects is given to the Author/Designer for rework and resolution.

The list of risks that could adversely affect SystemZ is given to Risk Management.

The list of opportunities for Built-in Self-Test (BiST) is given to Test Engineering.

The list of positive UiCs that could beneficially affect other systems is given to Marketing.

The list of negative UiCs that could adversely affect other systems is given to Management.

TestEngineer interviews the Moderator to ensure that he or she knows that he or she is responsible for editing the databases into the five prioritized lists. TestEngineer requests contact information for the head of Risk Management, head of Test Engineering, head of Marketing, and the Project Manager. This tests **FR3-8**. TestEngineer documents this meeting.

TestEngineer interviews the head of Risk Management, the head of Test Engineering, the head of Marketing and the Project Manager and records what they say that they will do with their lists. This tests **FR3-9**. TestEngineer documents these meetings.

7. Diogenes puts these prioritized lists in the project PAL.

TestEngineer writes a dummy file into the project PAL. This tests **FR3-10**. TestEngineer records the result.

8. **Rework** The Author/Designer fixes the defects. We assume that the other owners will know what to do with their lists.

TestEngineer interviews the head of Risk Management, the head of Test Engineering, the head of Marketing and the Project Manager and records what they say that they will do with their lists. This tests **FR3-9**. TestEngineer documents these meetings. This is the same activity as in step 6 above.

9. **Follow-up** The Moderator must verify that all fixes are effective and that no additional defects have been created. The Moderator checks the exit criteria for completion of an inspection.

TestEngineer interviews the Moderator to ensure that he or she knows how to verify that all fixes are effective, that no additional defects have been created and how to write exit criteria. This tests **FR3-11**. TestEngineer documents this meeting.

10. Diogenes updates the project PAL [exit use case].

TestEngineer changes the dummy file that he put into the project PAL in step 7 above. This tests **FR3-10**. TestEngineer records the result.

TestEngineer reviews his report ensures that all functional and nonfunctional requirements of this use case have been tested and submits his report to the head of Test Engineering. BIMS has not yet written metrics, thresholds or scores for this report that will ordain pass or fail of Diogenes.

This is the end of the test of the Perform Formal Inspection use case.

The other use cases will be verified in a similar manner.

Document 5: The Concept Exploration Document

The system design process is a use-case-based iterative process. Use cases describe the required functionality of the system: the functional and nonfunctional requirements are developed while writing the use cases. These customer requirements are used to derive technical requirements and create test procedures. Some of the highest priority requirements will be made into evaluation criteria for use in tradeoff studies. The system functions (described in the use cases) are mapped to classes and blocks, and these are interconnected to reveal the system architecture. Testing is more effective if it is planned right along with the system. Basing the tests on the use cases allows development of the testing procedures to begin right at the beginning of the system design. This should help produce more complete test plans and develop customer support for the test plans. The test plans should also include system experiments that will test the logic of the state machine diagrams. The system design process is not serial; it is very iterative and there are many opportunities for parallel processing.

Using Bahill’s system design process to design a process is certainly different. Not surprisingly, some of the documents are unusual. This is the case with document 5. In this document will consider the following four decisions.

1. Alternative implementations: paper and pencil, a computer, an Intranet with all data staying behind a firewall, an Internet system or a two-week on-site short course. The choice will be based on the type of system documentation that is available and the required level of confidentiality.

2. Alternative sources of knowledge: a systems engineer, a domain expert, a Delphi or a mixed team of domain experts and systems engineers.

3. Alternative incipient processes: a process for implementing tradeoff studies, a process for implementing risk analyses, failure modes and effects analysis (FMEA), a process for implementing sensitivity analyses, a process for creating comprehensive test plans, a process for formal inspections, a process for discovering requirements, mind mapping, concept mapping and finally the SIMILAR process. If the implementation were to be an electronic circuit, then we would also consider sneak circuit analysis.

4. Alternative names: Search for unintended consequences process (SUCP), BogSatBud, Diogenes, Dropn (defects, risks, opportunities for BIST, positive and negative unintended consequences)

Potential evaluation criteria

We will need evaluation criteria for each tradeoff study that we do. Some of the criteria may be reusable in several tradeoff studies. Here are the criteria for choosing a recommended process.

**Performance**

Name of criterion: *Ease of Use.* The system should be intuitive to use. There should not be a long learning curve. The system should hide the mathematics from the user. Low complexity is desired. It should back up its data frequently. It should be deterministic, not stochastic.

Name of criterion: *Looks Forward.* The system must look forward in time. Tools can be modified to look forward or backward in time. But for tools that have been in use for decades, there is so much literature and knowledge available, that switching would be confusing.

Name of criterion: *Has Tools.* The system should have tools to help the systems engineer to discover UiCs. Brainstorming is an example of a primitive tool. A fishbone diagram is a sophisticated tool.

Name of criterion: *Inside or Outside***.** The system must look for effects of the system being designed (SystemZ) on other systems, external to SystemZ.

**Cost**

Name of criterion: *Total Life Cycle Cost* (thousands of U. S. dollars). The total lifecycle cost shall be computed from the design costs, purchase costs, installation costs, operation and maintenance costs, and retirement and replacement costs over a predetermined period of time, such as five years. As of September 8, 2010, the numbers in the tradeoff study do not include operation, maintenance, retirement and replacement costs.

Name of criterion: *Training Time* (hours per person)**.** The average amount of time that it takes to train one systems engineer or member of an inspection team to use the system.

**Company Policy**

Name of criterion: *Built-in self-test*: It is difficult to get data for this criterion because either the manufacturers do not have BiST or they do not advertize it. BiST could be passive, where the BiST system monitors outputs and check points and displays status: or BiST could be active, where the BiST system generates signals and applies them to the system inputs.

Name of criterion: *Reusability*: It is company policy that non-developmental products be considered the primary solution to addressing customer needs with customized implementations being secondary solutions. To enhance the potential reuse in other present and future BICS systems, a Reuse Design Review with participation of top management shall be scheduled before the Systems Requirements Review. During the design phase, all engineers must be alert for reuse potential. In the tradeoff study, each alternative will be given points for use of COTS products and will be deducted points for use of custom-made products. Each alternative will be given points if the analysis and results are likely to be reused in different political environments. For example, it is federal, state and local government subsidies that make renewable-energy electric-generation systems economically feasible. So the economic analyses will be much different in different countries.

Name of criterion: *Vendor Evaluation*

Description: Each vendor will be evaluated on History in the marketplace, Financial stability, Support, Responsiveness, Upgrades, Enhancements, Custom enhancements, Third party support, Technology, Product maturity, Product stability, Vendor stability, Licensing cost, Site licenses, Licensing approach, Reputation, Maintenance approach, Complementarity and whether it is a Small business.

Weight of importance (priority): 8

Basic measure: A number between 0 and 1 that indicates the “goodness” of the vendor. It also warns of specific risks.

Units: none

Measurement method: The Analyst fills out the BICS Vendor Evaluation spreadsheet.

Input: The Analyst assigns cardinal numbers (probably integers) between 1 and 10.

Scoring function: no scoring function is necessary

Output: 0 to 1

Traces to Company Policy

|  |  |  |
| --- | --- | --- |
| Table 5-1. Prioritization of Evaluation Criteria | | |
| Evaluation Criteria | Weight of Importance | Trace to |
| Ease of Use. | 8 | CuR0-1, CuR0-3 |
| Looks Forward | 10 |  |
| Has Tools | 4 | FR2-3, FR2-5 |
| Inside or Outside | 8 |  |
| Total Life Cycle Cost | 4 | CoR2 |
| Operations Cost | 5 |  |
| BiST | 10 | FR2-1 |
| Reusability | 6 | Company policy |
| Vendor Evaluation | 7 | Company policy |

Alternative architectures

In the beginning, we will investigate the following eight alternative processes: (1) the SIMILAR process, (2) a risk analyses processes, (3) a cause and effect process, (4) a tradeoff study process, (5) a comprehensive testing process, (6) a sensitivity analysis process, (7) a formal inspection process and (8) a requirements discovery process.. Later we will perform tradeoff studies for different decisions.

Alternative 1, Do Nothing, the SIMILAR Process

The SIMILAR process will play the role of the Do Nothing or status quo alternative. In its simplest implementation, it is merely educated common sense. It involves stating the problem, investigating alternatives, modeling the system, launching the system, assessing performance and re-evaluating the process [Bahill and Gissing. 1998].

***S****tate the problem.* Stating the problem properly is the most important task. The problem statement describes the customer’s needs, states the goals of the project, prescribes the system’s capabilities, delineates the scope of the system, reports the concept of operations, describes the stakeholders, lists the deliverables and presents the key decisions that must be made

***I****nvestigate alternatives.* Alternatives solutions are identified, analyzed and compared using evaluation criteria.

***M****odel the system.* Models will be developed for most alternative designs. The model for the preferred alternative will be expanded and used to help manage the system throughout its entire life cycle. Models for dynamic system will be state-based.

***I****ntegrate*. Integration means designing interfaces and bringing system elements together so they work as a whole. This requires extensive communication and coordination.

***L****aunch the system.* Launching the system means running the system and producing outputs -- making the system do what it was intended to do. The system should be designed so that the processes are iterative and many tasks are done in parallel.

***A****ssess performance.* Performance is assessed using evaluation criteria, technical performance measures and metrics -- measurement is the key. If you cannot measure it, you cannot control it. If you cannot control it, you cannot improve it.

***R****e-evaluate.* Engineers use feedback to help control systems and improve performance: it is one of the most fundamental engineering tools. Re-evaluation should be a continual and iterative process with many parallel feedback loops.

This process can be summarized with the acronym SIMILAR [Bahill and Gissing, 1998].



Figure 5-1. The SIMILAR process

The outline of this section is that we explain the SIMILAR process and then processes for risk analyses, cause and effect analysis, creating tradeoff studies, writing comprehensive testing plans and doing sensitivity analyses. These processes provide alternative concepts for a process to search for unintended, but foreseeable, consequences.

Alternative 2, Risk Analyses

Risk is an expression of the potential harm or loss associated with an activity executed in an uncertain environment [Bahill and Smith, 2009]. Since 1662, it has been written that risk had at least two components. “Fear of some harm ought to be proportional not only to the magnitude of the harm, but also to the probability of the event” [Arnauld and Nicole, 1996, p. 274]. This is the first use of the **phrases** *magnitude of harm* and *probability of the event*. There are some ancient Greek, Chinese and biblical sources that have the concept of risk; but they do not have these words. It is unlikely that any older source has the words, because probability was not invented until the seventeenth century [Pascal, 1654].

The main tools of a risk analysis are a risk register, risk plots, mitigation plan (risk waterfall chart) and a hierarchy of models.

The first step in a risk analysis is to identify things that could go wrong, the failure modes. Haimes [2009] recommends having several models one for each aspect of the system. Some models will overlap: some will not. He calls these a hierarchical holographic model (HHM). Typical aspects include business models (with purchase orders, invoices, costs, schedules, and return on investment); architectural models; use case models; behavioral/functional models; requirements models; physical structure/component models; and performance/parametric models.

Next, the effects of those failures should be explained. These things that could go wrong are the project risks. These risks could affect cost, schedule, performance, operations, safety, etc. Next the likelihood of each risk occurring should be estimated. If the project has a lot of statistical data, then the probability of each event might be calculated. But typically such data are not available, so the frequency of occurrence in some given time interval is estimated.

The risk register is a spreadsheet summary of risks: it is examined monthly (or maybe more often) by management and the customer. It lists the most important risks along with event names, consequences, likelihood, severity, risk magnitude and who is responsible. It is based on the risk database, which contains sentences, paragraphs and test results that document each item.

The estimated frequency of occurrence is Arnauld’s probability of the event: the other half of his definition of risk is the magnitude of harm, or the severity of consequences. For insurance companies that is easy: they use dollar values. Often this will not work and the severity of consequences must be estimated qualitatively. Bahill and Smith [2009] provide a process. Usually risk is quantified as the product of frequency of occurrence and severity of consequences. It is very important that the range of frequency of occurrence and the range of severity of consequences be the same.

Many other attributes, such as difficulty of detection and vulnerability, have been included in the risk definition. If a problem is difficult to detect (in testing, verification, etc.) we should worry more about it and increase the risk number [Bahill and Karnavas, 2000]. If we know that we are vulnerable in a certain area, we should worry more about that area and increase the risk number. Haimes [2009] includes the attributes probability, severity of consequences, safety, efficiency, reliability and resilience.

The search for UiCs process will probably be related to resilience analysis. Within the US government, both the Department of Homeland Security and the White House have an Office of Resilience. The premise of these offices is that no country has the resources to protect all its infrastructure systems (power, water, transportation, etc.) against unpredicted disruptions, such as terrorist attacks or natural disasters. The theory is that systems can be designed to recover from such disruptions. To accomplish this recovery the system needs four attributes: capacity, flexibility, tolerance and cohesion as described by Jackson [2010]. Capacity is the degree to which the system has been designed to absorb the disruption. Flexibility is the ability of the system to reorganize in the face of a disruption. Tolerance is the capability to degrade gradually following a disruption. Cohesion is the ability of the elements of the system to communicate, cooperate and collaborate with each other. Jackson [2010] provides about 40 heuristics that enable the designer to achieve these attributes.

The risk analysis process is highly iterative and many tasks are done in parallel. After completing a risk analysis, you should look at (1) the high-risk events, (2) the high consequence events (no matter how unlikely) and (3) estimates that have large uncertainty. In the next iteration you should focus resources on these three items [Haimes, 2009].

|  |  |
| --- | --- |
| SIMILAR process | Risk Analysis |
| State the problem | Problem statement |
| Investigate alternatives | Identify multiple possible risks |
| Model the system | Identify frequency of occurrence and severity of consequences for each risk |
| Integrate | Organize the data into one file so the numbers can be compared. |
| Launch the system | Rank order the risks and discuss this with the decision maker and domain experts |
| Assess performance | Every month the risk register is presented to management and progress in addressing each risk is shown |
| Re-evaluate | Risk analysis is highly iterative |

Alternative 3, Cause and Effect Analysis

The purpose of a cause and effect analysis is to find the root cause of a failure. The main tools of a cause and effect analysis are Ishikawa fishbone diagrams and cause and effect diagrams. A lengthy discussion of cause and effect analysis was given at the end of document 1. An example fishbone diagram was given in figure 4-4.

|  |  |
| --- | --- |
| SIMILAR process | Cause and Effect Analysis |
| State the problem | Write a problem statement describing symptoms of the failure (the effects). Put its label in a box on the right (the fish head) and draw a horizontal arrow (the backbone) running to it. |
| Investigate alternatives | Draw four to six main bones. Labeled them with names of the primary probable causes of the problem. |
| Model the system | Fishbone diagram |
| Integrate | Fishbone diagram |
| Launch the system | Add bones to the diagram. Draw arrows to show cause and effect relationships. Ask, “Why did this happen?” Write subcauses branching off the causes. Continue to ask, “Why?” and generate deeper levels of causes. Prioritize and identify the most important causes. |
| Assess performance | Gather data to test the most important causes. |
| Re-evaluate | Start another iteration. |

Alternative 4, Tradeoff Studies

Humans often make poor decisions. To help them be better decision-makers, we teach systems engineers to create tradeoff studies. Tradeoff studies are broadly recognized as *the* method for simultaneously considering multiple alternatives with many criteria, and as such are recommended and mandated in the CMMI® Decision Analysis and Resolution process [CMMI, 2006]. Tradeoff studies, which are often called trade studies, involve a simultaneous mathematical consideration of many evaluation criteria for many alternatives, in parallel. An early description of a tradeoff study that summed weighted scores is given in a 1772 letter from Benjamin Franklin to Joseph Priestley [Bahill, 2007].

As with all systems engineering processes, creating a tradeoff study is a highly iterative process. Many iterations are done in each phase of the system design and the results are updated throughout [Daniels, Werner and Bahill 2001; Smith, Son, Piattelli-Palmarini and Bahill 2007]. In early iterations, this exploration helps produce the incipient system architecture [Rechtin 2000]. The tradeoff study is rewritten continually as more information becomes available.

The fundamental task of a tradeoff study is quantifying human values and preferences. The seminal academic descriptions of tradeoff studies appeared in Keeney and Raiffa [1976] and Edwards [1977]. Edwards focused on the numerical determination of subjective values. Keeney and Raiffa are best known for their axiomatic derivation of value and utility functions from conditions of preferential or utility independence. Tradeoff studies are prescribed in industry for choosing and ranking alternative concepts, designs, processes, hardware and techniques.



Figure 5-x. The tradeoff study process.

The components of a tradeoff study, include the (1) problem statement, (2) evaluation criteria, (3) weights of importance, (4) alternate solutions, (5) evaluation data, (6) scoring functions, (7) normalized scores, (8) combining functions, (9) preferred alternatives and (10) sensitivity analysis [Daniels, Werner and Bahill 2001; Smith, Son, Piattelli-Palmarini and Bahill 2007]. The tradeoff study process is highly iterative and many tasks are done in parallel.

|  |  |
| --- | --- |
| SIMILAR process | Tradeoff Study |
| State the problem | Problem statement |
| Investigate alternatives | Evaluation criteria, weights of importance, alternate solutions, evaluation data, scoring functions, scores, combining functions, preferred alternatives |
| Model the system | Each of the alternatives may have to be simulated in order to obtain reliable evaluation data. |
| Integrate | Combine all data into one representation and highlight the preferred alternatives |
| Launch the system | Create a spreadsheet or a database that contains all of the comparison data. Then compute numbers. |
| Assess performance | Present the results to the decision maker and see if he or she concurs. Perform a sensitivity analysis of the tradeoff study. |
| Re-evaluate | The tradeoff study process is not a serial process, many tasks are done in parallel and it is highly iterative. |

Alternative 5, Creating a Test Plan

The system design process is a use-case-based iterative process [Bahill, 2010]. It starts with a problem statement. Next, we make a rough schedule of who does what and when. Then we write the use cases that describe the required behavior of the system. While we are writing the use cases, we develop functional and nonfunctional requirements. These requirements must be verified. Design of tests can start as soon as the use cases are written. The systems engineers then derive the technical requirements and the test engineers create the test requirements. Now that we have some requirements, we can use them to form evaluation criteria that will be used in the tradeoff studies. Designing the system test plan is very iterative: it is not a serial process and as many tests as possible should be scheduled to be done in parallel. It should be based on the use cases.

|  |  |
| --- | --- |
| SIMILAR process | Test Plan |
| State the problem | Use cases |
| Investigate alternatives | Find all of the requirements that must be tested |
| Model the system | Find evaluation criteria and measures of effectiveness |
| Integrate | Roll up all of the subtests into tests |
| Launch the system | Perform the tests, doing as many tests in parallel as possible |
| Assess performance | Compile results of all the tests and discuss with the decision makers |
| Re-evaluate | Regression testing is iterative and cumulative |

Alternative 6, Sensitivity Analyses

A sensitivity analysis is a powerful technique for understanding systems. A sensitivity analysis identifies the most important parameters in a tradeoff study, often these are the cost drivers that are worthy of further investment [Smith, Szidarovszky, Karnavas and Bahill, 2008]. The earliest sensitivity analyses that we have found are the genetics studies on the pea reported by Gregor Mendel in 1865 [Stern and Sherwoods, 1966] and the statistics studies on the Irish hops crops by Gosset writing under the pseudonym Student around 1890 [Box, 1981].

In a sensitivity analysis, one or many parameters are mathematically or experimentally changed and then changes in outputs or performance indices are observed.

A sensitivity analysis should be performed anytime a model is created, a set of requirements is written, a system is designed, a decision is made, a tradeoff study iteration is completed, a risk analysis is originated or when you want to discover cost drivers. In a sensitivity analysis, values of parameters or inputs (or architectural features) are changed and changes in outputs or performance indices are measured. The results of a sensitivity analysis can be used to validate a model, warn of unrealistic model behavior, point out important assumptions, help formulate model structure, simplify a model, suggest new experiments, guide future data collection efforts, suggest accuracy for calculating parameters, adjust numerical values of parameters, choose an operating point, allocate resources, detect critical criteria, suggest tolerance for manufacturing parts and identify cost drivers.

A sensitivity analysis tells which parameters are the most important and most likely to affect system behavior and/or predictions of the model. Following a sensitivity analysis, values of critical parameters can be refined while parameters that have little effect can be simplified or ignored. In the manufacturing environment, they can be used to allocate resources to critical parts allowing casual treatment of less sensitive parts. If the sensitivity coefficients are calculated as functions of time, it can be seen *when* each parameter has the greatest effect on the output function of interest. This can be used to adjust numerical values for the parameters. The values of the parameters should be chosen to match the physical data at the times when they have the most effect on the output.

|  |  |
| --- | --- |
| SIMILAR process | Sensitivity Analysis |
| State the problem | Problem statement |
| Investigate alternatives | In a sensitivity analysis, selected system parameters are mathematically or experimentally changed and then changes in outputs or performance indices are computed. |
| Model the system | A sensitivity analysis can be done on a computer simulation or a physical system. |
| Integrate | Find the cost drivers |
| Launch the system | Compute the sensitivities for all of the parameters and discuss this with the decision makers. |
| Assess performance | Show the most important parameters to the decision maker and see if there are any surprises. |
| Re-evaluate | The sensitivity analysis process is very iterative and many tasks are done in parallel. |

Alternative 7, Formal Inspection Process

Formal software inspections were created by Michael Fagan in the 1970s. He emphasized that the cost of fixing a defect rises dramatically the later it is found in the development lifecycle of the program. Experience has shown that it costs about one man-hour to fix defects detected during inspection. It costs about three man-hours to fix defects detected during testing. It costs thirty man-hours to fix defects detected after release of the product. [Requirements engineers use the term inspection in a different sense. They say that each requirement must be verified by logical argument, inspection, modeling, simulation, analysis, test or demonstration. To them inspection means merely observing the object.]

For a system design, the material to be inspected will be use cases, tradeoff studies, verification and validation documents including test plans, and system designs for example sequence diagrams, class diagrams, state machine diagrams, block definition diagrams, parametric diagrams, etc. In this document, these will be called work products. The following is a more detailed listing of the work products broken down by the RUP’s like cycle phases of business model, requirements model, analysis model, design model, implementation model, operations model and the retirement model. (I could regroup these into SysML’s business models, architectural models, use case models, behavioral/functional models, requirements models, physical structure/component models and performance/parametric models.) The whole product is never inspected; small parts of it are examined at each inspection.

Work products

Work products of the business model

* Problem statement
  + describes the customer’s needs,
  + states the goals of the project,
  + delineates the scope of the problem,
  + reports the concept of operations,
  + describes the stakeholders,
  + lists the deliverables and
  + presents the key decisions that must be made.
* Concept of operations (ConOps)
* Context model (an object or data flow diagram) showing how the system fits into its overall environment
* High-level business requirements (essential use cases)
* Domain model (a class diagram) depicting major business classes
* Business process model (an activity diagram) depicting a high-level overview of the business process
* Glossary

Work products of the requirements model

* Operation concept description (OCD)
* System requirements specification (SRS)
* Requirements traceability
* Interface requirements specification
* Preliminary risk analysis
* Use case models for high-priority, high-risk items
* Technical performance measures
* Alternative concepts
* Incipient architectural description
* Process assets library
* Financial analysis
* Business case
* Glossary

Work products of the analysis model

* Elaborations of entities of requirements model
* Analysis packages composed of
  + high-level use cases
  + analysis classes
  + interface definitions
* Use case realizations with
  + textual descriptions
  + use case diagrams
  + class diagrams
  + activity diagrams
* Special Requirements sections containing
  + formal shall statement functional requirements
  + identification of the nonfunctional requirements
* Supplemental entities
  + functional flow block diagrams
  + object (context) diagrams
* The analysis model is a static, because it shows objects and their interrelationships, but not time-dependent dynamics.
* Tradeoff studies and key decisions
* Candidate architecture
* Risk analysis
* Financial analysis
* Business case
* Lessons learned
* Glossary

Work products of the design model

* Contains subsystems composed of
  + design classes
  + use case realizations
    - textual descriptions
    - class diagrams
    - sequence diagrams
  + interfaces
    - interfaces of the system and its environment
    - interactions when components of the system are integrated together
    - feedback loops within the system and with its environment
* Special Requirements sections contain
  + formal shall statement functional requirements
  + nonfunctional (performance) requirements
* Supplemental entities
  + state machine diagrams
  + block definition diagrams
  + parametric diagrams
* The design model is dynamic because, using sequence diagrams and state machine diagrams, it shows the timing of the functions being performed and the messages being passed
* Tradeoff studies and key decisions
* Architectural description
* Risk analysis
* Lessons learned
* Glossary

Work products of the implementation model

* Contains
  + components
  + interfaces
  + integration build plan
  + unit test cases and procedures
* Supplemental entities
  + dependencies between components
* Test plans There are three common techniques for testing systems
  + test vectors (input/output behavior) and test procedures
  + system experiments (state-based)
  + instances of use cases (scenario-based)
  + State-based testing is the best.

The operations model (usually a computer simulation) should be built from the implementation model. It should reflect the structure of the system as it was actually built. It will be used to manage and improve the operational system. It will be updated anytime the operational system is changed. Most importantly, it will used to help with retirement of the system. An activity diagram will be used to show the workflows, that is who does which tasks during the operations phase.

**Primary Actors**

The formal inspection typically involves an Inspection Team consisting of three to eight members who cover the following roles.

The **Moderator** leads the inspection, schedules meetings, distributes inspection materials, controls the meetings, reports inspection results and follows up on rework issues. Moderators should be trained in how to conduct inspections. The risk or quality assurance managers often serve in this role.

The **Author/Designer c**reates and/or maintains the work products being inspected. The Author/Designer answers questions asked about the work products during the inspection, looks for defects and fixes defects. The Author/Designer cannot serve as Moderator, Reader or recorder.

During the meeting, the **Reader** leads the Inspection Team through the work products being inspected, interprets sections of the artifact by paraphrase and highlights important parts**.**

The **Recorder** classifies and records defects and issues raised during the inspection. The Moderator might perform this role in a small Inspection Team.

The **Inspector** attempts to find errors in the work products. This role can be filled by one or several people. However, all participants act as inspectors, in addition to any other responsibilities. The following may make good inspectors: the person who wrote the specification for the work products being inspected; the people responsible for implementing, testing or maintaining the work product; a quality assurance representative; a representative of the user community; and someone who is not involved in the project but has infinite experience and perfect wisdom.

**Secondary Actors:** Process Asset Library, PAL

**Inspection activities**

**Planning**

The Moderator selects the Inspection Team, assigns roles, obtains work products to be inspected from the Author/Designer, and distributes them and other relevant documents to the Inspection Team two or three days prior to the inspection. The Moderator confirms that the material meets the entry criteria.

**Overview meeting**

This meeting gives the Author/Designer an opportunity to describe the important features of the work products to the Inspection Team.

**Preparation**

Each participant must examine the work products prior to the actual inspection meeting, independently noting any defects found or issues to be raised. Typically, this will take two hours for each participant. The amount of time each person spent will be recorded. Most of the errors found during inspections are identified during this preparation step. The work products should be compared against any predecessor documents or standards to assess completeness and correctness.

**Inspection meeting**

The Moderator and the Reader lead the team through the work products. At the beginning of the meeting, if the Moderator determines that reviewers are missing, that the reviewers did not spend sufficient time reviewing the material or that too much material had been scheduled for inspection, then the meeting should be rescheduled. Michael Fagan recommends inspecting about 130 lines of source code or requirements per hour. The discussion is usually structured as a round robin where each member comments on each issue. During the discussion, all inspectors can report defects or raise other issues, which are documented by the recorder. The Author/Designer can ask for clarification on points raised but is not allowed to defend or explain any defects. Make sure that everyone has the same version of the documents, because the Author/Designer is likely to have a newer version. The meeting should last no more than two hours.

**Rework**

The Author/Designer is responsible for resolving all issues raised during the inspection. This does not mean making every change that was suggested, but an explicit decision must be made about how each issue or defect will be dealt with.

**Follow-up**

The Moderator is responsible for following up with the Author/Designer to verify that the defects fixed in the rework phase have been performed properly. The Moderator should ensure that exit criteria have been satisfied.

**Process to Improve the Process**

All systems should have a Process to Improve the Process (PtItP). The inspection process can be a part of the PtItP. It can reveal kinds of defects being created and process changes that can be made to prevent them. This causal analysis can lead to improved quality on future work by helping to avoid making the same mistakes in the future.

Alternative 8, Requirements Discovery Process



Tradeoff Study Matrix for Diogenes

This may be an old version. Check the Excel file on our class web site for the latest version.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tradeoff Study Matrix for Diogenes using a Sum Combining Function and semirelative sensitivity functions. It answers the question, “What is the best architecture to use for a search for unintended consequences process?” | | | | | | | | | | | | | | | | | | | |
| Criteria | Wt. | Norm. Criteria Weights | Norm Sub Criteria Weights | Alt 1 Do Nothing, SIMILAR | | Alt 2 Risk Analysis | | Alt 3 Cause and Effect Analysis | | Alt 4 Tradeoff Studies | | Alt 5 Test Plan | | Alt 6 Sensitivity Analysis | | Alt 7 Formal Inspections | | Alt 8 Requirements Discovery | |
| **number of alternatives, m =** | 8 |  |  | Sc | Wt×Sc | Sc | Wt×Sc | Sc | Wt×Sc | Sc | Wt×Sc | Sc | Wt×Sc | Sc | Wt×Sc | Sc | Wt×Sc | Sc | Wt×Sc |
| **Performance** | 8 | 0.40 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Ease of Use | 8 |  | 0.27 | 0.9 | 0.24 | 0.3 | 0.08 | 0.5 | 0.13 | 0.4 | 0.11 | 0.3 | 0.08 | 0.1 | 0.03 | 0.8 | 0.21 | 0.5 | 0.13 |
| Looks Forward | 10 |  | 0.33 | 0.8 | 0.27 | 1.0 | 0.33 | 0.0 | 0.00 | 0.6 | 0.20 | 1.0 | 0.33 | 0.2 | 0.07 | 0.6 | 0.20 | 0.8 | 0.27 |
| Has Tools | 4 |  | 0.13 | 0.2 | 0.03 | 0.9 | 0.12 | 0.8 | 0.11 | 0.2 | 0.03 | 0.1 | 0.01 | 0.7 | 0.09 | 0.4 | 0.05 | 0.6 | 0.08 |
| Inside or Outside | 8 |  | 0.27 | 0.8 | 0.21 | 0.0 | 0.00 | 0.4 | 0.11 | 0.6 | 0.16 | 0.0 | 0.00 | 0.2 | 0.05 | 0.6 | 0.16 | 0.0 | 0.00 |
| **Cost** | 3 | 0.15 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Total Lifecycle Cost | 4 |  | 0.44 | 0.8 | 0.36 | 0.5 | 0.22 | 0.5 | 0.22 | 0.5 | 0.22 | 0.5 | 0.22 | 0.5 | 0.22 | 0.5 | 0.22 | 0.5 | 0.22 |
| Operating Cost | 5 |  | 0.56 | 0.5 | 0.28 | 0.5 | 0.28 | 0.5 | 0.28 | 0.5 | 0.28 | 0.5 | 0.28 | 0.5 | 0.28 | 0.5 | 0.28 | 0.5 | 0.28 |
| **Company Policy** | 9 | 0.45 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BiST | 10 |  | 0.43 | 0.2 | 0.09 | 0.4 | 0.17 | 0.2 | 0.09 | 0.8 | 0.35 | 0.0 | 0.00 | 0.8 | 0.35 | 0.3 | 0.13 | 0.5 | 0.22 |
| Reusability | 6 |  | 0.26 | 0.9 | 0.23 | 0.4 | 0.10 | 0.4 | 0.10 | 0.8 | 0.21 | 0.0 | 0.00 | 0.8 | 0.21 | 0.9 | 0.23 | 0.7 | 0.18 |
| Vendor Evaluation | 7 |  | 0.30 | 0.5 | 0.15 | 0.5 | 0.15 | 0.5 | 0.15 | 0.5 | 0.15 | 0.5 | 0.15 | 0.5 | 0.15 | 0.5 | 0.15 | 0.5 | 0.15 |
| **Alternative Rating** |  |  |  |  | **0.61** |  | **0.48** |  | **0.37** |  | **0.59** |  | **0.31** |  | **0.49** |  | **0.56** |  | **0.52** |

The Tradeoff Study Matrix above shows that the recommended alternative is the Do Nothing alternative, the SIMILAR process.

The values for the cells were derived by a panel of domain experts using the Delphi method. The values came from manufacturers’ data, peer reviewed journal papers and Internet web sites. A value was derived for each alternative for each criterion. This value was put into the scoring function for that criterion and the resulting score (Sc) was put into the matrix. *{I do not have scoring functions yet.}* Each score was then multiplied by its corresponding normalized weight (Wt) and these products were summed in each column to produce the Alternative Ratings.

This tradeoff study answered the question “What is the best architecture to use for a search for UiCs process?” by stating that the preferred alternative is the Do Nothing alternative, the SIMILAR process.

Each number in a tradeoff study matrix should have a verbal explanation. For example, consider the evaluation criterion *Ease of Use*. The system should be intuitive to use. There should not be a long learning curve. The system should hide the mathematics from the user. Low complexity is desired.

(1) The SIMILAR process has been used by millions of lay people for eons. So it gets a score of 0.9.

(2) For a risk analyses process (as used in industry), the risk register, risk plots and risk waterfall are easy to use, but finding the risks and gathering quantitative data are difficult. So it gets a score of 0.3.

(3) A cause and effect process is easy to use: but using it well takes a lot of experience. It gets a score of 0.5.

(4) A tradeoff study matrix is easy to construct: but avoiding mistakes is difficult. The sensitivity analyses part of a tradeoff study is difficult. It gets a score of 0.4.

(5) Traditionally, a comprehensive test plan was difficult to create: but now deriving one from use cases is easy. It gets a score of 0.3.

(6) A sensitivity analysis is used to evaluate the sensitivity of SystemZ to its inputs and parameters. Understanding the mathematics is difficult for lay people. It gets a score of 0.1.

(7) The formal inspection process was well designed. It gets a score of 0.8.

(8) The requirements discovery process requires a knowledgeable person. It gets a score of 0.5.

The values were originally estimated by Bahill, September 20, 2010. The most recent revision was done on November 6.

The second performance evaluation criterion is *Looks Forward.* The search for UiCs process must look forward in time. Admittedly, each tool could be modified to look forward or backward in time. But for tools that have been in use for decades, there is so much literature and knowledge available, that switching would be confusing.

(1) The SIMILAR process looks forward in time, backward in time and at the present time. Therefore, it gets a score of 0.8.

(2) A risk analyses process looks only forward in time. It gets a score of 1.0.

(3) A cause and effect process is used after there is a failure: it looks backward in time. It gets a score of 0.0.

(4) A tradeoff study process is used to make decisions involving the design of SystemZ. It looks at present and future times. Some evaluation criteria could be potential UiCs. It gets a score of 0.6.

(5) A comprehensive test plan process is used to test SystemZ and its requirements. It looks only forward in time. It gets a score of 1.0.

(6) A sensitivity analysis process is used to evaluate the sensitivity of SystemZ to its inputs and parameters. It computes sensitivities at the present time. But it can reveal unplanned sensitivities to mundane parameters and interactions. It gets a score of 0.2.

(7) The formal inspection process was designed to look in the present time: but there is no history that impedes it from looking forward. It gets a score of 0.6.

(8) The requirements discovery process was designed to look at present and future times. It gets a score of 0.8.

The values were originally estimated by Bahill, September 8, 2010. The most recent revision was done on November 6..

Consider the criterion *Has Tools.* The system should have tools to help the systems engineer to discover UiCs. Brainstorming is an example of a primitive tool. Prioritization is a simple tool. A fishbone diagram is a sophisticated tool.

(1) The SIMILAR process uses brainstorming, but it has nothing special, just a flow diagram. So it gets a score of 0.2.

(2) The risk analyses process has the risk register, risk plots, risk waterfall and HHM. So it gets a score of 0.9.

(3) A cause and effect process has cause and effect diagrams and Ishikawa fishbone diagrams. It gets a score of 0.8.

(4) A tradeoff study matrix and the sensitivity matrix are the tools. But these tools might not help us to find UiCs. It gets a score of 0.2.

(5) A comprehensive test plan uses test vectors and test procedures (input/output behavior), system experiments (state-based) and instances of use cases (scenario-based). But these tools will not help us find UiCs. It gets a score of 0.1.

(6) A sensitivity analysis uses partial derivatives and sensitivity functions. Unplanned excessive sensitivity to any parameter or sensitivity to interactions might flag UiCs. It gets a score of 0.7.

(7) The formal inspection process has well defined roles and activities. It can use a multitude of commercially available tools, but doesn’t have many tools of its own. It gets a score of 0.4.

(8) The requirements discovery process is well defined. MS Office tools and commercial requirements database management tools such as DOORS will suffice for most activities. It gets a score of 0.6.

The values were originally estimated by Bahill, September 20, 2010. The most recent revision was done on November 6.

Finally, let us consider the performance evaluation criterion *Inside or Outside*, whichevaluates whether the tool looks for events (causes, failures, sensitivities, etc.) that are inside of the system being designed (SystemZ) or outside of SystemZ. The search for UiCs process must look outside. Admittedly, each tool could be modified to look inside or outside of SystemZ. But for tools that have been in use for decades, there is so much literature and knowledge available, that switching would be confusing.

(1) The SIMILAR process does not care whether the event is inside or outside of SystemZ so it gets a score of 0.8.

(2) A risk analyses process (as used in industry) looks only for risks to SystemZ or its primary actors: so it gets a score of 0.0.

(3) A cause and effect process is used to look for the root causes of failures of SystemZ. It only looks for failures of SystemZ, but it can invoke causes that are outside of SystemZ. It gets a score of 0.4.

(4) A tradeoff study process is used to make decisions involving the design of SystemZ. However, we could create evaluation criteria that would look, for example, at harm to the environment. It gets a score of 0.6.

(5) A comprehensive test plan process is used to test SystemZ and its requirements. It would be hard to use it to look for outside entities. It gets a score of 0.0.

(6) A sensitivity analysis process is used to evaluate the sensitivity of SystemZ to its inputs and parameters. But the performance function could contain outside parameters. It gets a score of 0.2.

(7) The formal inspection process was designed to inspect SystemZ: but there is no historical baggage that impedes it from looking outside. It gets a score of 0.6.

(8) The requirements discovery process was designed to look at SystemZ. It gets a score of 0.0.

The values were originally estimated by Bahill, September 8, 2010. . The most recent revision was done on November 6.

One of the Company Policy criteria is the existence of *Built-in Self-Test*: BiST could be passive, where the BiST system monitors outputs and check points and displays status: or BiST could be active, where the BiST system generates signals and applies them to the system inputs.

(1) The SIMILAR process could use peer review. It gets a score of 0.2.

(2) A risk analyses process could check the numbers for validity, completeness and consistency. It gets a score of 0.4.

(3) A cause and effect process could use peer review. It gets a score of 0.2.

(4) A tradeoff study process could check the numbers for validity, completeness and consistency. Excel has lots of built in tests. It gets a score of 0.8.

(5) I have never heard of a test plan to test the test plan. It gets a score of 0.0.

(6) Examining the symmetry and regularity of the equations in a sensitivity analysis is a good error detection metric. My examples have two sensitivity matrices computed with different data and formulae, but they contains the same values. It gets a score of 0.8.

(7) The formal inspection process uses management oversight. The metrics analysis group will analyze the metrics. It gets a score of 0.3.

(8) Examining the regularity of the grammar and structure of the requirements could be used for BiST. It gets a score of 0.5.

The values were originally estimated by Bahill, September 20, 2010. The most recent revision was done on November 6.

One of the Company Policy criteria is *Reusability*: It is company policy that non-developmental products be considered the primary solution to addressing customer needs with customized implementations being secondary solutions. In the tradeoff study, each alternative will be given points for use of COTS products and will be deducted points for use of custom-made products. Each alternative will be given points if the analysis and results are likely to be reused in different environments.

(1) The SIMILAR process has been used and reused by millions of lay people for eons; and it is free. So it gets a score of 0.9.

(2) There are COTS packages for risk analyses. However, outputs of a risk analysis are marginally usable in future analyses. It gets a score of 0.4.

(3) There are COTS packages for cause and effect analyses. However, the outputs of a cause and effect analysis would not be usable in future analyses. It gets a score of 0.4.

(4) There are no excellent COTS packages for tradeoff studies; however, everyone can use my examples and templates. The artifacts are extremely reusable in future studies. It gets a score of 0.8.

(5) Test plans and procedures are extremely specific. It gets a score of 0.0.

(6) There are no good COTS packages for sensitivity analyses; however, everyone can use my examples and templates. My Excel spreadsheets are very reusable in future studies. It gets a score of 0.8.

(7) The formal inspection process is used over and over again. It gets a score of 0.9.

(8) Most requirement systems use DOORS. The requirements themselves are often reused. It gets a score of 0.7.

The values were originally estimated by Bahill, September 20, 2010. The most recent revision was done on November 6.

When filling in a tradeoff study matrix, it is important to pay attention to common mental mistakes (Smith, Son, Piattelli-Palmarini and Bahill 2007). For example, the criteria Total Lifecycle Cost and Operating Cost are dependent and evaluation criteria should be independent.

The sensitivity analysis (Smith, Szidarovszky, Karnavas and Bahill, 2008) performed on this tradeoff study showed that the most important category weight was that for *Company Policy*, the most important subcategory weights were for *Looks Forward* and *BiST*, and the most important scores were for *BiST* for alternatives 4 and 6, and *Looks Forward* for alternatives 2 and 5. Therefore, we spent extra time discussing amongst ourselves and with top management, whether *Performance*, *Cost* or *Company Policy* should have the highest weight. We expended extra resources consulting with management and the customer about the relative weights of the subcriteria. We did extra research, analysis and modeling to get scores that are more reliable for *BiST* for alternatives 4 and 6, and *Looks Forward* for alternatives 2 and 5.

|  |  |  |  |
| --- | --- | --- | --- |
| Sensitivity rank | Criteria | reliability | uncertainty |
| 1 | Company Policy weight | 8 | 8 |
| 2 | Looks forward weight | 9 | 9 |
| 3 | BiST weight | 9 | 9 |
| 4 | BiST score for Tradeoff Studies | 4 | 4 |
| 4 | BiST score for Sensitivity Analysis | 4 | 4 |
| 6 | Looks Forward score for Risk Analysis | 9 | 9 |
| 6 | Looks Forward score for Test Plan | 9 | 9 |

Document 6: The Use Case Model

None of the processes explained in document 5 satisfy all of the system requirements. Therefore, Diogenes was created as a hybrid of the eight alternatives examined in document 5. For the top-level operation, it uses the SIMILAR process, alternative 1. For low-level detailed operation, it uses the techniques of formal inspections, alternative 7. When searching for risks, it uses tools of risk analysis, alternative 2. To discover requirements, it uses the requirements discovery process of alternative 8. In preparing work products for inspection, it uses cause and effect analysis of alternative 3. It uses symmetry and regularity as exemplified in alternatives 4, 6 and 8.

A *use case* is an abstraction of required functions of a system. A use case produces an observable result of value to the user. Each use case describes a sequence of interactions between one or more actors and the system.

**Use Case 1.** This is a concrete business use case.

**Name:** Sell Diogenes

**Iteration:** 1.3

**Derived from:** Discussions with the Brain Trust and students

**Brief description:** BICS sells Diogenes to CMMI level 3 or 5 industrial companies. Hereafter they will be called Potential Customers.

**Level:** high

**Priority:** highest priority

**Scope:** Each copy ofDiogenes will be owned by the Risk Management department or the Quality Assurance department of the client company. It will be a normal part of the risk analysis process. For engineers analyzing work products of a SystemZ, Diogenes will produce five databases: the defects, risks, opportunities for BiST, positive untended consequences and negative UiCs of SystemZ

**Added value:** BICS will obtain revenue. Our Customer will learn how to identify defects, risks, opportunities for BiST, positive UiCs and negative UiCs.

**Goal**: Sell Diogenes

**Primary actors:** BICS and the Brain Trust

**Supporting actors:** Potential Customers

**Frequency**: BICS would like to sell one Diogenes system per month.

**Precondition**: BICS has Diogenes ready to sell.

**Trigger:**

**Main Success Scenario:**

1. BICS writes a sales brochure that most importantly includes a case for change.

2. BICS sends letters to the department heads of Risk Management and Quality Assurance at two dozen Potential Customers.

3. A Potential Customer responds by phone, letter or e-mail.

4. BICS describes the format (two-week on-site short course) and price ($50,000).

5a. BICS and the Customer negotiate a contract.

6. Customer sends a purchase order and a schedule is negotiated.

7. Two Brain Trust members go and deliver the course.

8. At the end of each short course, each student should tell his or her manager what the student learned in the course and give the manager a BICS evaluation form. The form will ask questions such as, “Did this student’s learning match your expectations for this course?” This form will be returned to BICS.

9. BICS bills Customer.

10. Customer pays BICS [exit use case].

**Anchored Alternate Flow:**

5b. Customer backs out [exit use case].

**Postcondition:** BICS is ready to sell Diogenes to another Customer.

**Specific Requirements** (Daniels and Bahill 2004)

**Functional Requirements:**

**FR1-1:** BICS shall write a sales brochure that includes a case for change.

**FR1-2:** BICS shall send letters to the department heads of Risk Management and Quality Assurance of Potential Customers.

**FR1-3:** BICS shall be capable of communicating with Customers by phone, letter and e-mail.

**FR1-4:** BICS shall be capable of negotiating contracts, arranging schedules and billing.

**FR1-5:** BICS shall create short course materials (PowerPoint presentations and handouts).

**FR1-6:** Brain Trust members shall be trained to deliver the short course.

**FR1-7:** BICS shall develop, validate and summarize course evaluation questionnaires.

**Nonfunctional Requirements:**

**Author/owner:** Terry Bahill

**Last changed:** September 22, 2010

**Use Case 2.** This is a concrete use case.

**Name:** Search for Unintended Consequences (Names of use cases are set in the Verdana font.)

**Iteration:** 2.3

**Derived from:** ConOps

**Brief description:** A Systems Engineer uses Diogenes to help find UiCs of SystemZ, where SystemZ is the class name for a new system being designed.

**Level:** high

**Priority:** This is of the highest priority.

**Scope:** The Systems Engineer, the Design Team, the design documents of SystemZ, Diogenes and the portion of the external world that SystemZ could affect. A typical *product* design would not have the external world within its scope.

**Added value:** The company and society might be able to ameliorate adverse UiCs of SystemZ or capitalize on positive UiCs.

**Goal:** Find potential UiCs of SystemZ.

**Primary actors:** Systems Engineer (which could be a single person or a whole department), Design Team (which consists of design engineers, domain experts and managers), Moderator

**Supporting actors:** Society, Boss, PAL

**Frequency**: Diogenes will be used daily.

**Precondition:** Diogenes has passed all of its Built-in Self-Tests.

**Trigger:** Conclusion of a design review, which is the start of a new iteration.

**Main Success Scenario:**

1a. A design review has been successfully completed and the next phase of the system life cycle has been authorized to begin.

2. If one does not already exist, the Boss creates a Systems Engineering Team to search for UiCs of SystemZ.

3a. If a UiCs Report is in the process assets library (PAL), then Diogenes retrieves it.

4. The Moderator works with the Systems Engineer, uses cause and effect tools (in either a tree or a fishbone format) and creates UiCs attribute and impact diagrams. These tools are built into Diogenes.

5. Include the Perform Formal Inspection use case.

6. The Systems Engineer shows the prioritized lists of potential negative UiCs to management and assesses the results.

7a. Management authorizes remedial action.

8. Include the Redesign to Avoid Unintended Consequences use case.

9. Diogenes puts the UiCs Report in the project PAL [exit use case].

**Anchored Alternate Flow:**

1b. The system design fails a design review and the project is cancelled or the design team is given another chance [exit use case].

3b. If a UiCs Report is not in the PAL, then the Systems Engineer uses the Problem Statement, writes a UiCs Report and Diogenes puts it in the PAL [return to step 3a].

3b1. If the Problem Statement is not available, then the Systems Engineer writes one [return to step 3b]. The problem statement contains a description of the use cases, risks, design alternatives, most sensitive parameters, test plans and the system design.

7b. Management decides to do nothing about the UiCs

7b1. Include the Ethics Deliberation use case [exit use case].

**Postcondition:** The project PAL has been updated and the Systems Engineer is ready to start a new iteration.

**Specific Requirements** (Daniels and Bahill 2004)

**Functional Requirements:**

**FR2-1** Diogenes shall execute Built-in Self-Tests (BiST). (Derived from BICS company policy)

**FR2-2** Diogenes shall be capable of reading and writing the project PAL.

**FR2-3** Diogenes shall have cause and effect tools (in both tree and fishbone formats) that have been modified for making UiCs diagrams.

**FR2-4** Diogenes shall have the capability of creating and maintaining five databases: the defects, risks, opportunities for BiST, positive untended consequences and negative unintended consequences of SystemZ

**FR2-5** Diogenes shall have tools for prioritizing lists.

**Nonfunctional Requirements:**

**Author/owner:** Terry Bahill

**Last changed:** September 9, 2010

A use case diagram is the table of contents of a use case model. It shows all of the use cases that have been described so far. Figure 6-1 is our first use case diagram.

****

Figure 6-1. A use case diagram for Diogenes

**Use Case 3.** This is an abstract included use case.

**Name:** Perform Formal Inspection

**Iteration:** 2.3

**Derived from:** document 5, Alternative Concepts

**Brief description:** A formal inspectionis a structured group review process used to find defects in requirements, programming code, test plans and designs.The flow of this use case is called from theSearch for Unintended Consequences use case. When this subflow ends, the use case instance continues where this included use case was called.

**Level:** medium

**Priority:** medium

**Scope:** The Inspection Team, the work products to be inspected and the PAL.

**Added value:** The company will be able to look for defects, risks, opportunities for BiST, negative and positive UiCs all at the same time. This should increase efficiency. Furthermore discovering positive UiCs could provide substantial revenue.

**Goal:** Find potential defects, risks, opportunities for BiST, negative and positive UiCs of SystemZ.

**Primary actors:** Inspection Team comprised of Moderator, Author/Designer, Reader, Recorder, and additional Inspectors. Note: the following explanations of these roles were copied from alternative 7 of document 5.

The **Moderator** leads the inspection, schedules meetings, distributes inspection materials, controls the meetings, reports inspection results and follows up on rework issues. Moderators should be trained in how to conduct inspections. The risk or quality assurance managers often serve in this role.

The **Author/Designer c**reates and/or maintains the work products being inspected. The Author/Designer answers questions asked about the work products during the inspection, looks for defects and fixes defects. The Author/Designer cannot serve as Moderator, Reader or recorder.

During the meeting, the **Reader** leads the Inspection Team through the work products being inspected, interprets sections of the artifact by paraphrase and highlights important parts**.**

The **Recorder** classifies and records defects and issues raised during the inspection. The Moderator might perform this role in a small Inspection Team.

The **Inspector** attempts to find errors in the work products. This role can be filled by one or several people. However, all participants act as inspectors, in addition to any other responsibilities. The following may make good inspectors: the person who wrote the specification for the work products being inspected; the people responsible for implementing, testing or maintaining the work product; a quality assurance representative; a representative of the user community; and someone who is not involved in the project but has infinite experience and perfect wisdom.

**Secondary Actors:** PAL, the process assets library, the place where all of the project’s important files are kept.

**Frequency**: one a week

**Precondition:** An Author/Designer has requested an inspection of his work product

**Trigger:** This use case will be included from the Search for Unintended Consequences use case.

**Main Success Scenario:**

1. **Planning** The Moderator selects the Inspection Team, obtains work products to be inspected from the Author/Designer and distributes them along with other relevant documents to the Inspection Team.

2. **Overview meeting** The Moderator explains the inspection process to the Inspection Team. The Author/Designer may describe the important features of the work products.

3. **Preparation** Each member of the Inspection Team examines the work products prior to the actual inspection meeting. Typically, this will take two hours for each member. The amount of time each person spent will be recorded. Each member should be looking for five things simultaneously: defects, risks, opportunities for BiST, positive and negative UiCs of SystemZ

4. **Inspection meeting** The Moderator and Reader lead the team through the work products. The issues are brought up one by one and each one is discussed in a round robin fashion where each member comments on each issue. During the discussion, all inspectors can report defects, risks, opportunities for BiST, positive and negative UiCs of SystemZ, all of which are documented by the Recorder. The meeting should last no more than two hours.

How can we prime our inspectors to look for UiC?

If they are looking at an activity, action, process, procedure, or other verb phrases (an active verb followed by a measureable noun), then tell them to ask, “What problems could this activity create for other systems?” “How could this activity hurt other systems?” “If this activity failed, how could it hurt other systems?”

If they are looking at an object, component, model, or other noun phrases, then tell them to ask, “How could this object hurt other systems?” “How could this object fail?” For each failure event ask, “How could this failure event hurt other systems?”

If they are looking at a risk, then tell them to ask, “How could this failure event hurt other systems?”

If they are looking at a use case, scenario or other sequences of events, then tell them to ask, “What-if?” For example, The user does this and the system does that.“What if it doesn’t?”

5. Diogenes creates and maintains five databases that contain defects, risks, opportunities for BiST, positive and negative UiCs of SystemZ.

6. The Moderator and the Systems Engineer consolidate and edit the five databases to create five prioritized lists.

The list of defects is given to the Author/Designer for rework and resolution.

The list of risks that could adversely affect SystemZ is given to Risk Management.

The list of opportunities for Built-in Self-Test (BiST) is given to Test Engineering.

The list of positive UiCs that could beneficially affect other systems is given to Marketing.

The list of negative UiCs that could adversely affect other systems is given to Management and the Legal department.

7. Diogenes puts these prioritized lists in the project PAL.

8. **Rework** The Author/Designer fixes the defects. Each of the other owners will know what to do with his list.

9. **Follow-up** The Moderator must verify that all fixes are effective and that no additional defects have been created. The Moderator checks the exit criteria for completing of an inspection.

10. Diogenes updates the project PAL [exit use case].

**Anchored Alternate Flow:**

**Postcondition:** The project PAL has been updated and the Systems Engineer is ready to schedule a new inspection.

**Specific Requirements** (Daniels and Bahill 2004)

**Functional Requirements:**

**FR3-1** The Moderator shall form the Inspection Team.

**FR3-2** The Moderator shall collect the inspection work products and other relevant material and distribute them to the Inspection Team TBD days before the inspection.

**FR3-3** The Moderator shall chair the overview meeting.

**FR3-4** Each member of the Inspection Team shall examine the work products prior to the actual inspection meeting looking for defects, risks, opportunities for BiST, positive unintended consequences and negative unintended consequences of SystemZ.

**FR3-5** Each member of the Inspection Team shall record and report the number of hours he or she spent inspecting the materials. Typically, this will be two hours.

**FR3-6** The Moderator shall conduct the inspection meeting.

**FR3-7** The Recorder shall create and maintain five databases that contain defects, risks, opportunities for BiST, positive untended consequences and negative unintended consequences of SystemZ.

**FR3-8** The Moderator and the Systems Engineer shall consolidate and edit the five databases to create five prioritized lists.

**FR3-9** The Systems Engineer shall deliver the five lists to their respective owners.

**Stipulation** Each owner will know what to do with his list.

**FR3-10** Diogenes shall put these five prioritized lists in the project PAL.

**FR3-11** The Moderator shall verify that all fixes are effective and that no additional defects have been created. The Moderator shall check the exit criteria for completing of an inspection.

We have often said that we can impose requirements on our system, but we cannot impose requirements on operators, pilots and other secondary actors. This is still true. Here we are imposing requirements on the Moderator and members of the Inspection Team. That is all right, because they are a part of Diogenes.

**Nonfunctional Requirements:**

**NFR3-1** The Moderator shall schedule the inspection meeting for two hours. The Moderator shall prepare two dozen pages of documentation for each inspection.

**Author/owner:** Terry Bahill

**Last changed:** October 8, 2010

**Use Case 4.** This is a concrete use case.

**Name**: Gather Evidence of Verification and Validation

**Iteration:** 3.1

**Derived from:** BICS company policy

**Brief Description:** TheSystems Engineer continually gathers evidence that can be used to prove verification and validation of the system.

**Added value:** Gathering verification and validation evidence throughout the life cycle will make verification and validation more meaningful.

**Level:** Function

**Scope:** This use case affects the entire design effort.

**Primary actor:** Systems Engineer

**Frequency:** Many times per month

**Precondition:** Design of the system must have begun

**Main Success Scenario:**

1. Systems Engineer continually gathers evidence that can be used to prove verification and validation of the system [repeat at step 1].

**Postcondition:** Systems Engineer has gathered evidence.

**Author/owner:** Terry Bahill

**Last changed:** October 8, 2010

Other Use Case Models

So far, we have written four use cases. A complete design will probably have dozens of use cases. Here are some other proposed use cases:

Redesign to Avoid Unintended Consequences

Ethics Deliberation

|  |  |
| --- | --- |
| Names of potential classes: | Descriptions |
| Architectural Model | architectural models |
| Behavioral model | behavioral/functional models |
| Business Model | business models |
| Dynamic Object |  |
| Feedback Loop | feedback loops in SystemZ and suggests their seriousness |
| Interaction | interactions when components of SystemZ are integrated together |
| Interface | interfaces of SystemZ |
| nUiCs List | the prioritized list of potential negative UiCs |
| PAL | process assets library |
| Performance Model | performance/parametric models |
| Physical Structure Model | physical structure/component models |
| Problem Statement |  |
| pUiCs List | the prioritized list of potential positive UiCs |
| Requirements Model | requirements models |
| Risk List | the prioritized list of potential system risks |
| Risk Register | spread sheet summary of risks that is examined monthly by management |
| SystemZ | the new system being designed |
| UiCs Report | package containing … |
| Use Case Model | use case models |

Document 7: The Design Model



Figure 7-1. Diogenes, a search for UiCs process.



Figure 7-2. Diogenes, applied in the system design process.



Figure 7-2. Activity diagram for Diogenes

**Validation.** This paragraph on validation could be moved to document 4. To help validate Diogenes we will compare it to the SIMILAR process.

|  |  |
| --- | --- |
| SIMILAR process | Diogenes |
| State the problem | Planning and the Overview Meeting |
| Investigate alternatives | Fix Defects, Manage Risk, Incorporate opportunities for BiST, Market Positive UiC and Ameliorate negative UiC. |
| Model the system | Preparation is where the inspectors study models of the system and look for Dropn. |
| Integrate | There is no analogous activity. |
| Launch the system | Inspection, Create Databases and Write and Distribute Lists is where the real work is performed. |
| Assess performance | Gather Verification and Validation Data |
| Re-evaluate | Follow-up is where we re-evaluate SystemZ and Execute Process to Improve the Process is where we re-evaluate Diogenes. |

Diogenes maps well to the SIMILAR process. The only exception is that Diogenes does not have an integration activity. Should it have one?

**When** should you search for unintended, but foreseeable, consequences? You should search for UiCs near the end of each phase of the system life cycle. When a proposal is nearly finished, search for UiCs. Before each design review, search for UiCs. Perhaps the biggest search for UiCs should occur at the end of system design. The designer has already performed the tradeoff studies, risk analysis, sensitivity analysis and comprehensive testing plan. Using these studies as input, he can now broaden his scope and start looking outside of the specific problem domain. Also near the end of testing, production and delivery, he should search for UiCs.



Figure 7-3. Activities as a function of time in the system life cycle. As the systems engineering budget is reduced the horizontal axis of the Unintended Consequences activity can be raised producing instant pruning of the activity.

Re-evaluate

When designing a process, it is important to include a process to improve the process (PtItP). The PtItP is that Bahill creates a preliminary process, as in Table 4. He distributes this to the Brain Trust and certain INCOSE fellows. They provide e-mail comments and verbal discussion for a few months. Bahill then revises the process (Table x) and applies it to a newly designed system. The feedback process is used again. This process is given to seniors and graduate students in the *Systems Engineering Process* class. Their usage of the process and related documents are studied. Eventually the process will be written up in a paper and submitted for review and possible presentation at the INCOSE symposium. If it is received favorably, then it will be submitted to the journal *Systems Engineering*. There it will be subjected to rigorous review by a half dozen subject matter experts. If accepted it will be published in the journal. pUiCs

**Analysis Classes**

|  |
| --- |
| Positive UICs list <<entity class>> |
| Revision date  Properties |
| File |

The attribute “Properties” contains all of the information that MS Word gives when you right click a file icon and choose the properties entry. These could be in a file instead of an attribute. The operation “File” contains all of the functions that you see when you left click the File tab, such as Save, Save as, Open, Close, Print

|  |  |
| --- | --- |
| Functions and events | Explanation |
| From the class diagrams |  |
| blankScreen() | The system must blank out the display screen. |
| performBiST() | The system must perform BiST. |
| writePositiveUICsDatabase() | The system must write information into the entity PositiveUICsDatabase |
|  |  |
| Events from State Machine Diagrams |  |
| BiST OK | The system determines that the BiSTs are OK and sends this message. |
| BiSTfailure | The BiST fails and the system send this message. |

Document 8: Models, Mappings and Management

Schedule

A schedule for this project is given in the Fall 2010 SIE-454/554 document “Difficulty and Value of SIE-454/554 Homework Problems,” which is approximately page 19 of your course notes.

|  |  |  |
| --- | --- | --- |
| Problem Set | Problems | Due date |
| HW1 | 1-1, 2-3, 2-6, 2-13 | Aug 30 |
| Project | Docs 1, 8 and PPS | Sept 1 |
| HW2 | 3-1, 3-2, 3-3, Risk Analysis | Sept 13 |
| Project | Doc 5 with minutes of meeting | Sept 15 |
| HW3 | 3-8, 3-11, 3-14 | Sept 20 |
| Project | Doc 6 | Sept 22 |
| HW4 | Sensitivity (in CN) | Sept 27 |
| Project | Doc 2 & 4 | Sept 29 |
| HW5 | SIERRA (CN) with use case, 4-3, 4-8, 8-1, Requirements (CN) | Oct 4 |
| Project | Doc 3 & 4 | Oct 6 |
| HW6 | 5-1, 5-2, 5-3 | Oct 11 |
| Project | Doc 7 | Oct 13 |
| Exam 1 |  | Oct 20 |
| HW8 | Boeing 747 | Oct 22 |
| HW7 | Complexity | Nov 1 |
| Tinker Toy Tower Design | Room 301 of the Engineering Building | Nov 1 & 3 |
| HW10 | Mental Mistakes | Nov 15 |
| HW9 | Good Design Principles | Nov 22 |
| Class Project | All eight documents | Dec 8 |
| Final Exam |  | Dec 15 |

Acronyms

|  |  |
| --- | --- |
| BICS | Bahill Intelligent Computer Systems |
| BIMS | BICS Illuminance Management System |
| BiST | Built-in Self-Test |
| CDR | Critical design review |
| CDRL | Contract deliverable requirements list |
| ConOps | Concept of operations |
| CoR | Cost requirement |
| CuR | Customer requirement |
| Dropn | Defects, risks, opportunities for BiST, positive unintended consequences and negative unintended consequences. |
| FR | Functional requirement |
| HVAC | Heating, ventilation and air conditioning |
| IC | Illuminance controller |
| LED | Light emitting diode |
| NFPR | Nonfunctional performance requirement |
| nUiCs | Negative UiCs |
| OCD | Operational concept description |
| PAL | Process assets library |
| PDR | Preliminary design review |
| pUiCs | Positive UiCs |
| PV | Photovoltaic |
| RR | Risk requirement |
| SI | Systeme International d'Unites |
| SR | Schedule requirement |
| SRR | System requirements review |
| TBD | To be determined |
| TEP | Tucson Electric Power Company |
| UiCs | Unintended consequences |

Glossary of Terms (mostly Wikipedia pages)

Perhaps it is no longer necessary or economically feasible to provide a glossary with every document. The best alternative may be to refer the readers to a subset of the Wikipedia.

<http://minnesotafuturist.pbworks.com/Anticipatory-Thinking-i4CQuest#PagesincategoryAnticipatorythinking>

SystemZ is the name for a new system being designed.

PAL is the process assets library, the place where all of the project’s important files are kept.

Dropn is defects, risks, opportunities for BiST, positive unintended consequences and negative unintended consequences.

Business Plan

Most of the value of Diogenes will be the knowledge of the Brain Trust and BICS personnel. BICS will offer to come into a company and spend two weeks teaching people in the Risk Management or Quality Assurance departments how to use it. Cost for this would be $50,000.

Bahill Intelligent Computer Systems (BICS) will match employee contributions to charitable organizations up to $2000 per year. Unfortunately, BICS is too small to have a maternity leave or a sabbatical leave program. Riding a motorcycle without a helmet or skydiving is an announcement by a BICS employee of a desire to terminate employment.

Risk analysis

Risk is an expression of the potential harm or loss associated with an activity executed in an uncertain environment (Bahill and Smith 2009).

A possible risk for Diogenes would be that fails system validation. This would manifest in document 4 by Diogenes failing to identify UiCs of one or many SystemZ. The consequence would be that no one would use Diogenes. To mitigate this risk, Diogenes will have to be validated on many well-documented system designs.

The terms relative frequency and relative likelihood are not quite synonyms. If we have historical data for the frequency of occurrence of an event, then we use the term relative frequency. Otherwise, if we are guessing the future, then we use the term relative likelihood. The word *relative* emphasizes that it is the relationships between risks that are being illustrated. The word *likelihood* is not used in a mathematical sense, but rather in its dictionary sense to indicate the events that are most likely.

Each row in a risk table describes a *failure* *event* and the *consequences* of that event. Both the event and the consequences can have uncertainty. To estimate the frequency of occurrence of a particular consequence, we first look to see if there are historical data for it. If not then we estimate the number of events that will occur in a certain time interval, and then multiply that by the likelihood of the particular consequencesgiven that the event has taken place. For example, let us estimate the risk of dying on a trip from SF to LA. Driving from SF to LA takes about five hours. We can estimate the number of accidents in a five-hour period and then multiply that by the likelihood that a person in a freeway/interstate accident would die. In a risk table, a failure event might have multiple consequences, as in the following illustration.

|  |  |  |
| --- | --- | --- |
| Failure Event | Consequences | Expected frequency of occurrence per year |
| Number of lightning strikes per year in the Tucson area = 105 | Household electric devices are destroyed, likelihood per lightning strike = 10-2 | 1000 |
|  | A house burns down, likelihood per strike= 10-3 | 100 |
|  | A person is injured, likelihood per strike= 10-4 | 10 |
|  | A person dies, likelihood per strike= 10-5 | 1 |

Sometimes we know when an event will occur, so the likelihood that the event will occur is 1.0 and we only need to estimate the likelihood of the consequence. For example if you are about to flip a coin, the likelihood that the event will occur is 1.0 and the likelihood that the consequence will be a head is 0.5. Therefore, the likelihood that you will flip a head in the next moment is 0.5. Most gambling games are of this nature.

Sometimes there is no uncertainty in the consequence, only uncertainty that the event will occur in the specified time interval. For example, in radioactive decay of radium-226 into radon-222, we can estimate the frequency of the event as. When the event occurs, we know with absolute certainty that the consequence will be an alpha particle; so the likelihood of this consequence is 1.0 and the expected frequency of occurrence is .

We can compare these three types of events and consequences (1, uncertainty in both the event and the consequences, 2, uncertainty in only the consequences and 3, uncertainty in only the event) as long as the time interval is the same.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk Analysis for Diogenes | | | | |
| Failure Event | Consequences | Relative likelihood (events per year) | Severity of consequences | Estimated Risk |
| Diogenes fails system validation | No one would use Diogenes. | 0.6 | 300 | 180 |
| Diogenes and the Recorder cannot keep up with the Inspection Team comments in real time | Time of important people would be wasted and chaos would ensue. | 0.5 | 40 | 20 |
| No Brain Trust member wants to travel and teach companies how to use Diogenes | BICS would have to hire less qualified systems engineers. | 0.1 | 100 | 10 |
| Software mistakes | Software mistakes are ubiquitous, but hard to diagnose, particularly when they involve interacting systems. Redundancy and built in self-test help reduce the severity. Therefore, every software routine in Diogenes shall have built in self-test. | 10 | 1 | 10 |
| Terrorists or industrial spies steal systemZ’s designs. | Systems that put SystemZ’s designs on the Internet will not be viable. | 0.2 | 5 | 1 |
| Loss of Dropn data in the PAL | PAL must be restored from backups | 0.1 | 5 | 0.5 |
| A lawyer or a Wall Street banker thinks that Diogenes has exposed him, so he sues BICS. | BICS is presently a sole proprietorship, we will have to incorporate and buy liability insurance. | 0.1 | 2 | 0.2 |
| Management refuses to do anything about negative UiCs. | This would put the systems engineer in an ethical dilemma. | 0.4 | 3 | 0.12 |
| A politician thinks that Diogenes might cost him an election. | Congress taxes profits from Diogenes at 90%. | 0.03 | 2 | 0.06 |
| A similar system has already been patented | It will cost at least $10,000, but we should apply for a patent. | 0.03 | 1 | 0.03 |

The range of magnitudes for relative likelihood and severity of consequences **must** be the same. In this table, the relative likelihood covers two and a half orders of magnitude  and the severity of consequences also covers two and a half orders of magnitude.

Each of these risks should spawn customer requirements.

These numbers are for the present system. Some of these numbers are quite high. This means that certain risks must be mitigated (to some extent) before the system can progress further through its system life cycle.

By far and away, the biggest risk is that Diogenes fails system validation. Therefore, we must put a lot of effort into this risk, right now.

Summary

**Acknowledgements.** We thank the Systems Engineering Brain Trust, George Dolan, Bob Sklar, Brad Sowers, Bruce Gissing, Al Chin and Gary Lingle and INCOSE Fellows Mark Maier and Scott Jackson for valuable comments on the project. We thank Sparx Systems for a site license for Enterprise Architect.

References

*Apollo 13*, a movie produced by Imagine Entertainment and Universal Pictures, 1995.

Arnauld, Antoine, and Pierre Nicole, *Logic, or, the Art of Thinking: Containing, Besides Common Rules, Several New Observations Appropriate for Forming Judgment*, fifth edition, published in 1662, translated from French in 1996 by Jill Vance Buroker, Cambridge University Press.

Bahill, A. T., Design and Testing of an Illuminance Management System, *The ITEA Journal,* **31**(1):63-89, March 2010.

Bahill, A. T. and Botta, R., Fundamental principles of good system design, *Engineering Management Journal,* **20**(4):9-17, 2008.

Bahill, A. T. and Gissing, B., Re-evaluating systems engineering concepts using systems thinking, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* 28(4), 516-527, 1998.

Bahill, A. T. and Henderson, S. J., Requirements development, verification and validation exhibited in famous failures, *Systems Engineering*, 8(1):1-14, 2005.

Bahill, T. A. and Karnavas, W. J. Risk Analysis of a Pinewood Derby: A Case Study. *Systems Engineering,* 3(3), (2000): 143-155.

Bahill, A. T. and Smith, E., An Industry Standard Risk Analysis Technique, *Engineering Management Journal,* **21**(4):16-29, 2009.

Botta, R., Bahill, Z. and Bahill, A. T., When are observable states necessary? *Systems Engineering*, **9**(3):228-240, 2006.

Botta, R. and Bahill, A. T., A prioritization process, *Engineering Management Journal*, **19**(4):20-27, 2007.

Box, J. F., Gosset, Fisher, and the t distribution, *American Statistician,* **35**(2), pp. 61-66, summer 1981.

Chaves, A. and Bahill, A. T., Risk Analysis for Incorporating Photovoltaic Solar Panels into a Commercial Electric Power Grid, 2011.

Clausen, D. and D. D. Frey, Improving System Reliability by Failure-Mode Avoidance Including Four Concept Design Strategies, *Systems Engineering* **8**:3 (2005), pp. 245-261.

*CMMI for Development,* v 1.2, Software Engineering Institute, Pittsburgh, PA, 2006,

<http://www.sei.cmu.edu/cmmi/tools/dev/download.cfm> retrieved May 2010.

Daniels, J., Werner, P. W. and Bahill, A. T., Quantitative methods for tradeoff analyses, *Systems Engineering,* 4(3), 190-212, 2001, with a correction published in 8(1):93, 2005.

Daniels, J. and Bahill, A. T., The hybrid process combines traditional requirements and use cases, *Systems Engineering,* **7**(4):303-319, 2004.

GE Energy, *Western Wind and Solar Integration Study, 2010*, National Renewable Energy Laboratory, May 29, 2010.

<http://www.nrel.gov/wind/systemsintegration/wwsis.html>

<http://www.nrel.gov/wind/systemsintegration/pdfs/2010/wwsis_executive_summary.pdf>

<http://www.nrel.gov/wind/systemsintegration/pdfs/2010/wwsis_final_report.pdf>

retrieved May 2010.

Haimes, Yacov Y., Risk Management, in *Handbook of Systems Engineering and Management*, Andrew P. Sage, and W. B. Rouse (Editors), John Wiley & Sons (2009), pp. 155-204.

Haimes, Yacov Y., *Risk Modeling, Assessment, and Management*, third edition, 2009, John Wiley & Sons

Holmes, S. A., Fannie Mae eases credit to aid mortgage lending, *The New York Times*, September 30, 1999.

Hooper, J. J., T. Foecke, L. Graham and T. P. Weihs, The metallurgical analysis of wrought iron from the RMS Titanic, *Measurement Science and Technology,* **14**(2003) 1556–1563

Ishikawa, K. (1990); (Translated by J. H. Loftus); *Introduction to Quality Control*; 448 pages; ISBN: 490622461X 9784906224616, OCLC Number: 61341428, This book is an English translation of the third edition of "Dai-3-pan Hinshitsu Kanri Nyumon,” originally published in Japanese by JUSE Press.

Jackson, S. (2010). *Architecting Resilient Systems: Accident Avoidance and Survival and Recovery from Disruptions.* Hoboken, NJ: John Wiley & Sons.

Juran, J. M. (1989). *Juran on leadership for quality: An executive handbook.* New York: Free Press.

Kirkwood, Craig W., Decision Analysis, in *Handbook of Systems Engineering and Management,* Andrew P. Sage, and W. B. Rouse (Editors), John Wiley & Sons (2009), pp. 1223-1250.

Kunzig, R., Europe's dream, *Discover*, **18** (May 1997) 96-103.

List, J. A., Michael Margolis, and Daniel E. Osgood “Is the endangered species act endangering species?” Working Paper 12777 NBER Working Paper Series, December 2006, http://www.nber.org/papers/w12777, for a free copy go to http://graphics8.nytimes.com//images/blogs/freakonomics/pdf/FreakPDF4.pdf

Numeroff, L. J. and F. Bond, *If you give a mouse a cookie,* Harper Collins, 1985.

Pascal, Blaise, Fermat and Pascal on Probability, (1654) www.york.ac.uk/depts/maths/histstat/pascal.pdf accessed October 27, 2008.

Smith, E. D. and Bahill, A. T., Attribute substitution in systems engineering, *Systems Engineering*, **13**(2):130-148, 2010. DOI: 10.1002/sys.20138.

Smith, E. D., Son, Y. J., Piattelli-Palmarini, M. and Bahill, A. T., Ameliorating Mental Mistakes in Tradeoff Studies, *Systems Engineering,* **10**(3):222-240, 2007.

Smith, E. D., Szidarovszky, F., Karnavas, W. J. and Bahill, A. T., Sensitivity analysis, a powerful system validation technique, *The Open Cybernetics and Systemics Journal,* http://www.bentham.org/open/tocsj/openaccess2.htm, **2**:39-56, 2008, doi: 10.2174/1874110X00802010039

Stern, C. and E. R. Sherwoods, *The origins of genetics: A Mendel source book*, Freeman, San Francisco, CA, pp. 308-312, 1966.

*Titanic,* a Lightstorm Entertainment Movie Production (20th Century Fox and Paramount). 1997.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

At a Design Review held on October 15, 2010, unintended13.xls and unintended13.docx were reviewed by the following members of the Brain Trust: George Dolan, Bob Sklar, Bruce Gissing, Al Chin and Terry Bahill.