Regular Paper

Diogenes, a Process for Identifying Unintended Consequences

A. Terry Bahill*

Systems and Industrial Engineering, University of Arizona, Tucson, AZ 85721-0020

Received 2 February 2011; Revised 1 August 2011; Accepted 22 October 2011, after one or more revisions
Published online 13 February 2012 in Wiley Online Library (wileyonlinelibrary.com).
DOI 10.1002/sys.20208

ABSTRACT

Individuals, companies, and even governments often create procedures, processes, or products that solve a particular problem, only to discover that their solution has created a second problem, worse than the first. These secondary problems are called unintended consequences. Searching for unintended consequences, as a part of the development process, will likely increase safety, reduce financial risk, and improve customer satisfaction. This paper contains the design for a new process, named Diogenes, that will help systems engineers identify unintended, but foreseeable, consequences of a new system that is being designed. It contains the required behavior (functions), use cases, design diagrams, the test procedure, validation, and verification for Diogenes. © 2012 Wiley Periodicals, Inc. Syst Eng 15: 287–306, 2012

Key words: emergent behaviors; system design; verification; validation

1. INTRODUCTION

"[T]he problem of the unanticipated consequences of purposive action has been treated by virtually every substantial contributor to the long history of social thought. ... [T]hough the process has been widely recognized and its importance equally appreciated, it still awaits a systematic treatment" [Merton, 1936, p. 894]. This statement is still true. However, in this paper, we present an incipient systematic process for identifying unintended consequences.

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

This paper starts with the motivation, first showing negative unintended consequences (UiCs), positive UiCs, a case for change, and emergent behaviors. Then it presents the Diogenes design model. This is followed by an extensive validation of Diogenes. A verification and test plan follows. Finally, it explains the features that are desirable in a process for indentifying unintended consequences.

2. MOTIVATION

2.1. Examples of Negative Unintended Consequences

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

* E-mail: terry@sie.arizona.edu

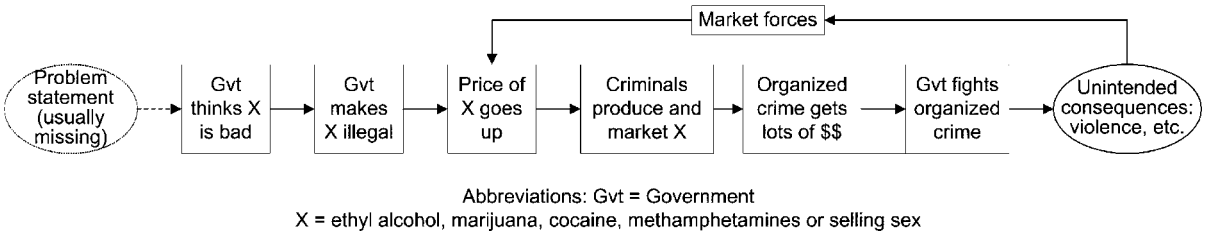

Figure 1. A process repeated often by the government.

The pattern shown in Figure 1 has been repeated over and over again. The US government, perhaps spurred by activists (like religious fanatics, self-righteous moralists, and zealots), decides that a certain inexpensive chemical is not good for the people, so they make it illegal. They do not bother to state the problem or to investigate alternative solutions. Then criminals produce and market this chemical on the black market at high prices. This concentrates a large amount of money into the hands of criminals, which produces the UiCs of murders, kidnappings, robberies, burglaries, police and judicial corruption, explosion of prison populations, spread of diseases, and increased violence. The government has done this with ethyl alcohol, marijuana, gambling, cocaine, magic mushrooms, ecstasy, and selling sex."Those who cannot remember the past are condemned to repeat it" [Santayana, 1905, p. 92].

In 1997 (in order to protect the Cactus Ferruginous Pygmy Owl around Tucson, Arizona) the US 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 1 year earlier than similar noncritical habitat parcels [List, Margolis, and Osgood, 2006]. 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.

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; with the UiC of 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 has the UiC of causing many users to write their passwords on paper (as they are now impossible to remember), negating the security advantage of strong passwords.

The systems engineer should strive to continuously improve the performance and also the cost of his system. There is an urban legend that, in 1915, Henry Ford told his engineers to search automobile junkyards and find out why his Model T's were there. The results showed that some had broken transmissions, some had worn-out motors, etc., but none of them were in the junkyards because the kingpins had failed. So Ford immediately ordered that the kingpins be made smaller.

An event can cause a UiC. Then that consequence can cause another UiC, producing a series of UiCs. As an apocryphal example, once, a long time ago, a legendary king, anticipating Henry Ford's process, ordered that the number of nails in the horseshoes be reduced so that they could use the saved iron to make extra swords. In the first battle, the king realized that he had made a mistake, when horses began to lose their shoes. This 600-year-old rhyme tells the saga:

> 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.

2.2. Positive UiCs

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

Adam Smith in *The Wealth of Nations* [Smith, 1776] was one of the first to suggest positive UiCs. Norton [1993, p. 92] described it this way:

> The concept of unintended consequences is one of the building blocks of economics. Adam Smith's "invisible hand," the most famous metaphor in social science, is an example of a positive unintended consequence. Smith maintained that each individual, seeking only his own gain, "is led by an invisible hand to promote an end which was no part of his intention," that end being the public interest. "It is not from the benevolence of the butcher, or the baker, that we expect our dinner," Smith wrote, "but from regard to their own self interest."

Similarly, a designer strives to make simple models for complex systems, because that is his job. However, creating an exquisite design will also satisfy his deep personal intellectual needs and allow other designers to learn from his design.

In the 1950s, IBM® could have bought the patents for Xerox's photocopy machine at a low price. 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 positive UiCs of the copy machine were that owners could delight their customers with a machine that provided dozens of copies in just minutes and that the original could be used again and again.

In 1928, Alexander Fleming was doing research on bacteria that had infected patients at his hospital. One morning he noticed that he had accidentally left the cover off of a Petri dish of Staphylococcus bacteria. Overnight a green mold had blown in through a window and killed the bacteria in the Petri dish. He isolated and cultured this green mold and eventually identified it as penicillium. This is an example of serendipity or sagacious positive UiCs.

In 1938, a DuPont chemist was attempting to use tetrafluoroethylene to make a new gas for refrigeration. However, as a positive UiC, the iron of the pressurized storage container catalyzed the polymerization of tetrafluoroethylene into polytetrafluoroethylene**,** Teflon®!

2.3. Case for Change

The United States of America has problems such as global warming, rising national debt, poverty, illegal immigration, 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 negative UiCs. However, it is possible to create a system that will help decision makers to discover potential negative UiCs 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 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. Because, 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*. Nevertheless, the program manager might say that the harmed system has no power over him and his system. And he may be right, but the Customer and the Customer's Customer do have that power, although their effects are not direct. Moreover, 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.

Most companies already have processes for discovering defects and risks. With Diogenes, while you search for defects and risks, you can also find opportunities for Built-in Self-Test (BiST), positive UiCs, and negative UiCs. This process does not generate or need new data, but rather it takes information that is already generated for development and reviews and uses it in a way that may anticipate (and thus avoid) future problems.

Diogenes will be a new process in any company. Therefore, to get management support, it will need a powerful sponsor or champion. Perhaps the Vice President for Engineering would be appropriate for this role.

2.4. Emergent Behaviors

Complex systems contain diverse, interdependent, and adaptive components. Most of the systems cited in this paper contain human components, and humans are diverse, interdependent, and adaptive; therefore, most of these systems are complex systems. In complex systems, unintended consequences manifest as emergent behaviors [Sheard and Mostashari, 2009]. Complex systems exhibit two forms of emergence, weak and strong.

Weak emergence is defined as system behaviors that cannot be found in the behaviors of the system's parts, but that can be explained. For example, non-steroidal, anti-inflammatory drugs (NSAIDs) such as Ibupropren and Aleve *reduce* the effectiveness of angiotensin converting enzyme (ACE) inhibitors, which are used to treat hypertension. Whereas grapefruit interferes with enzymes that metabolize statins (Zocor, Lipitor) in the digestive system, this can *increase* the potency of the medication to dangerous levels. Similarly, the performance of a system can be greater than the sum of its subsystems (cooperation). For example, a pair of chopsticks performs more than twice as well as an individual chopstick; two lions chasing a Thomson's Gazelle are more than twice as likely to catch it, than a single lion.

Strong emergence is defined as system behaviors that cannot be found in the behaviors of the system's parts or in the interactions between the parts.

> The possibility of strong emergence follows from an ensemble perspective, which states that physical systems are only meaningful as ensembles rather than individual states. Emergent properties reside in the properties of the ensemble rather than of any individual state. A simple example is the case of a string of bits including a parity bit, i.e., the bits are constrained to have, e.g., an odd number of ON bits. This constraint is a property of the entire system that cannot be identified through any set of observations of the state of any or all subsystems of the system. It is a property that can only be found in observations of the state of the system as a whole [Bar-Yam, 2004, p. 15].

An ant colony also manifests strong emergent behavior. The queen does not tell individual ants what to do. Instead, the olfactory system of each ant reacts to odor molecules from larvae, other ants, intruders, food, and waste. Each ant leaves behind a pheromone trail, which provides an olfactory stimulus to other ants. Other examples exhibiting strong emergent behavior include the World Wide Web, Internet traffic governed by the TCP, and web-based social networking.

Diogenes should be able to help predict weak emergent behaviors, but not strong emergent behaviors.

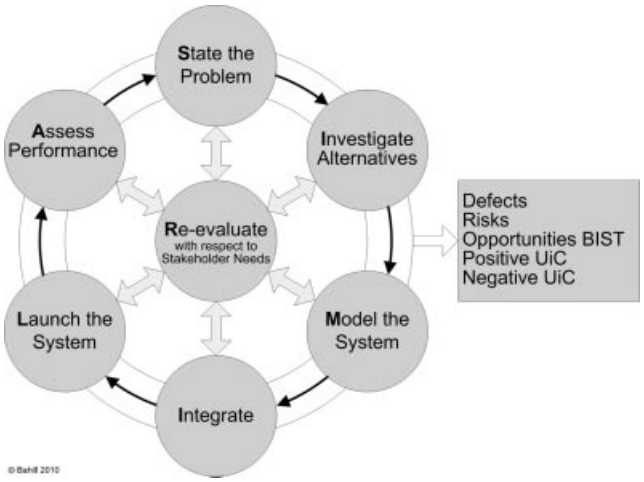

Figure 2. High-level functional block diagram for Diogenes, a process for finding UiCs, based on Bahill and Gissing [1998].

3. THE DIOGENES DESIGN MODEL

Product Position Statement: For systems engineers, who need to ensure the global success of a new system that they are designing, Diogenes is a process that will help identify unintended, but foreseeable, consequences of the new system. Unlike risk and failure analyses, Diogenes identifies *future* effects on *other* systems that might be caused by the new system being designed (see Figs. 2 and 3 and Table I).

This search for UiCs process is named after Diogenes of Sinope, a Greek philosopher who lived in the 5th century B.C. Diogenes was a Cynic who disdained conventional wisdom and political correctness, which he thought caused most people to be intellectually dishonest. He carried a lantern in the daytime saying that he was "in search of an honest man."

3.1. 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 UiCs can be done with a process similar to risk discovery. Therefore, Diogenes might be packaged with risk analysis and could be owned by the Risk Management or the Quality Assurance departments of a company. Fortunately, we have found that defects, positive UiCs, 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 system design processes, this process is iterative and hierarchal. In each iteration, the emphasis on what is being discovered will shift. 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 new 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, (3) opportunities for BiST, (4) positive UiCs that could beneficially affect other systems, and (5) negative UiCs that could adversely affect other systems.

3.2. 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 could get funded.

3.3. Work Products

Diogenes will create and maintain five databases: defects, risks, opportunities for BiST, positive UiCs and negative UiCs. The Moderator and the SystemsEngineer 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 will be given to the Author/Designer for resolution. Correcting these defects could be managed by the

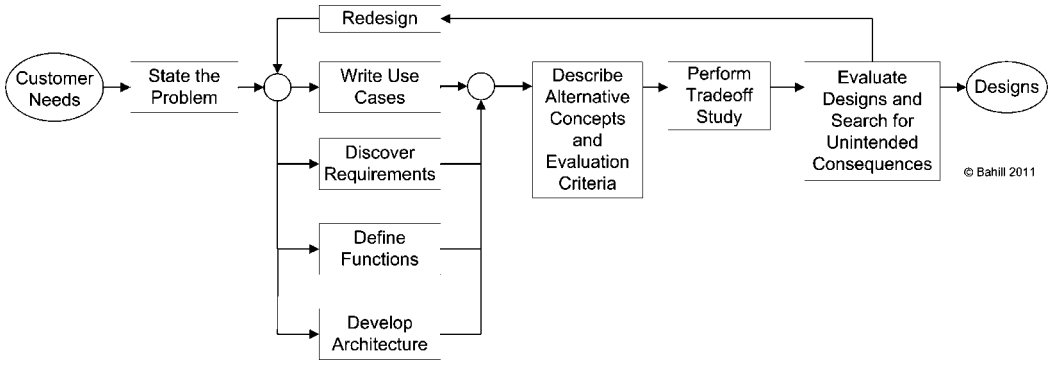

Figure 3. Diogenes is embedded in this low-level functional flow block diagram for the system design process.

Table I. Acronyms

BICS	Bahill Intelligent Computer Systems
BIMS	BICS Illuminance Management System
BiST	Built-in Self-Test
FR	Functional Requirement
NFR	Nonfunctional Requirement
nUiCs	Negative Unintended Consequences
pUiCs	Positive Unintended Consequences
PAL	Process Assets Library.
SystemZ	Name of the new system being designed that Diogenes will be applied to.
UiCs	Unintended Consequences

Moderator or they could be submitted to a change control board.

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) will be 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) will be given to Test Engineering.
4. The list of positive UiCs, which could beneficially affect other systems, will be given to Marketing, because discovering positive UiCs could provide additional revenue.

And, finally, our original target:

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

Some new medical devices are being required to document possible human abuse of the device, both intentional (like a Little Leaguer hitting rocks with a $150 carbon-fiber baseball bat) and unintentional (like driving a car at 15 mph in fourth gear). Strategies for ameliorating such abuse must be documented. This could constitute a sixth prioritized list for Diogenes.

Diogenes puts these prioritized lists in the project process assets library (PAL), which is the place where all of the project's important files are kept. These include the five prioritized lists as well as the problem statement, the UiC report, and inspection materials.

3.4. When Should Diogenes Be Used?

The search for UiCs should occur near the end of each phase of SystemZ's life cycle (Fig. 4). 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, during or after the Critical Design Review (CDR), because, at this point, 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, the designer should search for UiCs near the end of testing, production, and delivery.

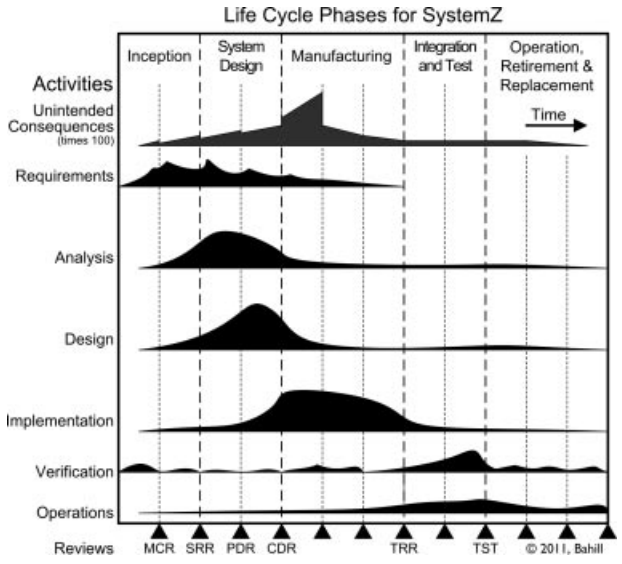

Figure 4. Activities as a function of time in the system life cycle. If you cannot afford to run Diogenes in each iteration, then raise the horizontal axis of the UiCs activity to eliminate the iterations you cannot afford. Acronyms: MCR, Mission concept review: SRR, System requirements review: PDR, Preliminary design review: CDR, Critical design review: TRR, Test readiness review: TST, Total system test.

Bahill's system design process is a use-case-based iterative process. It starts with a problem statement followed by a rough schedule of who does what and when. Now we write the use cases that describe the behavior of the system. While we are writing the use cases, we develop functional and nonfunctional requirements. These go into the customer requirements document and also into the requirements verification section. Test design 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 form evaluation criteria that will be used in the tradeoff studies. For Diogenes these requirements documents are eight pages long [see Bahill, 2011].This process is very iterative: It is not a serial process. There must be many iterations, and there are many opportunities for parallel processing.

3.5. The Use Case Model

A *use case* is an abstraction of required behavior 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 [Jacobson, 2000; Cockburn, 2001]. Use case 1 is to Sell Diogenes: It has been omitted from this paper.

Use Case 2. This is a concrete use case (meaning that it is invoked by the primary actor).

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

Iteration: 3.1

Derived from: Concept of operations (ConOps)

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

Scope: The SystemsEngineer, the DesignTeam, 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: SystemsEngineer (which could be a single person or a whole department), and Moderator

Supporting actors: Society, Boss, PAL and DesignTeam (which consists of design engineers, domain experts, and managers)

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. The Boss creates a Systems Engineering Team to search for UiCs of SystemZ, unless one already exists.
3a. If a UiCs Report is in the process assets library (PAL), then Diogenes retrieves it.
4. TheModerator works with the SystemsEngineer, uses cause and effect tools, and creates UiCs attribute and impact diagrams. These tools are built into Diogenes.
5. *Include* the Perform Formal Inspection use case.
6. The SystemsEngineer shows the prioritized lists of potential negative UiCs to management and assesses the results.
7a. Management authorizes remedial action.
8. *Include* the Ameliorate Negative UiCs 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 SystemsEngineer 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 SystemsEngineer 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 SystemsEngineer is ready to start a new iteration.

Specific Requirements: See Daniels and Bahill [2004].

Functional Requirements (FRs):

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 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 UiCs, and negative UiCs of SystemZ.

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

Author/owner: Terry Bahill
Last changed: October 4, 2011

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 5 is our first use case diagram.

Use Case 3. This is an abstract included use case.
Name: Perform Formal Inspection
Iteration: 3.1
Derived from: concept exploration document [Bahill, 2011; Fagan, 2011]

Brief description: A formal inspection is 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 the Search for Unintended Consequences use case. When this subflow ends, the use case instance continues where this included use case was called.

Scope: The InspectionTeam, the work products to be inspected, and the PAL

Added value: The company will be able to look for defects, risks, opportunities for BiST, and positive and negative 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, and positive and negative UiCs of SystemZ.

Primary actors: InspectionTeam comprised of Moderator, Author/Designer, Reader, Recorder, SystemsEngineer, and additional Inspectors:

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. 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, or other members of the design team, cannot serve as Moderator, Reader, or Recorder.

During the meeting, the **Reader** leads the InspectionTeam through the work products being inspected, interprets

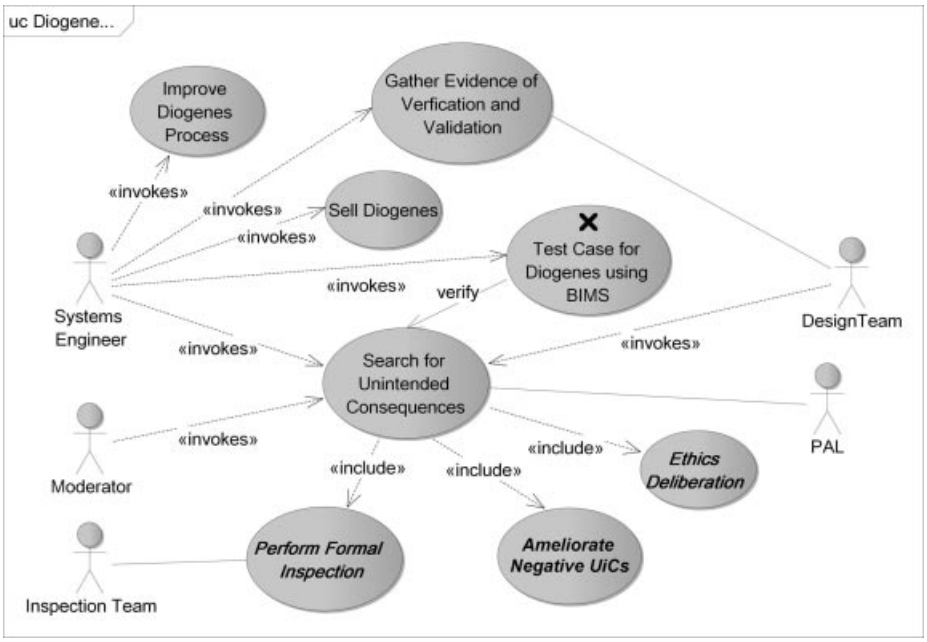

Figure 5. A use case diagram (**uc**) for Diogenes. The SystemsEngineer is a primary actor: The DesignTeam is not: because the DesignTeam will not aggressively search for negative UiCs.

sections of the artifact by paraphrase, and highlights important parts**.**

The **Recorder** classifies and records defects, risks, opportunities for BiST, positive and negative UiCs, and issues raised during the inspection. The Moderator might perform this role for a small InspectionTeam.

The **Inspector** attempts to find errors in the work products. This role can be filled by one or several people. 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: the Process Assets Library (PAL)

Frequency: once 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 activity:** The Moderator obtains the problem statement and the work products to be inspected from the Author/Designer and distributes them along with other relevant documents to the InspectionTeam.
2. **Overview meeting:** The Moderator explains the Diogenes process to the InspectionTeam. This will take from 10 min to 3 h, depending on the backgrounds of the team members. The Author/Designer may describe the important features of the work products.
3. **Preparation:** Each member of the InspectionTeam examines the work products prior to the actual inspection meeting. Each member should be looking for five things simultaneously: defects, risks, opportunities for BiST, and positive and negative UiCs of SystemZ. Typically, this will take 2 h for each member. The amount of time each person spends will be recorded. This time would be substantially increased for an Inspector running models and simulations to validate the system.
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, and positive and negative UiCs of SystemZ, all of which are documented by the Recorder. The meeting should last no more than 2 h.

How can the inspectors be primed to look for UiC? Tell them to ask questions like these. What other systems could this thing (object, activity, interface, risk, etc.) affect? How could this thing affect other systems? Why would this thing affect other systems? What-if this thing failed? Inspectors should also look for common mental mistakes that people make [Smith et al., 2007], particularly for attribute substitution, which is the commonest of the mental mistakes [Smith and Bahill, 2010]. Inspectors should verify that the designers used fundamental principles of good design [Bahill and Botta, 2008], including design for resiliency [Jackson, 2010]. But

we really want the mindset of looking for UiCs to become a part of company culture.

5. **Databases:** Diogenes creates and maintains five databases that contain defects, risks, opportunities for BiST, and positive and negative UiCs of SystemZ.
6. **Prioritized lists:** The Moderator and theSystemsEngineer consolidate and edit the five databases to create five prioritized [Botta and Bahill, 2007] lists:
   The prioritized list of defects is given to the Author/Designer for rework and resolution.
   The prioritized list of risks that could adversely affect SystemZ is given to Risk Management.
   The prioritized list of opportunities for Built-in Self-Test (BiST) is given to Test Engineering.
   The prioritized list of positive UiCs that could beneficially affect other systems is given to Marketing.
   The prioritized list of negative UiCs that could adversely affect other systems is given to Management and the Legal department.
7. **PAL:** 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. **Update PAL:** Diogenes updates the project PAL [exit use case].

Postcondition: The project PAL has been updated, and the SystemsEngineer is ready to initiate a new inspection.

Specific Requirements: See Daniels and Bahill [2004].

Functional Requirements (FRs):

FR3-1. The Boss shall form the InspectionTeam.

FR3-2. The Moderator shall collect the inspection work products and other relevant material and distribute them to the InspectionTeam to be determined (TBD) days before the inspection.

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

FR3-4. Each member of the InspectionTeam shall examine the work products prior to the actual inspection meeting looking for defects, risks, opportunities for BiST, and positive UiCsand negative UiCs of SystemZ.

FR3-5. Each member of the InspectionTeam shall record and report the number of hours he or she spent inspecting the materials. Typically, this will be 2 h.

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 UiCs, and negative UiCs of SystemZ.

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

FR3-9. The SystemsEngineer shall deliver the five lists to their respective owners.
Stipulation: Each owner will know what to do with his or her list.

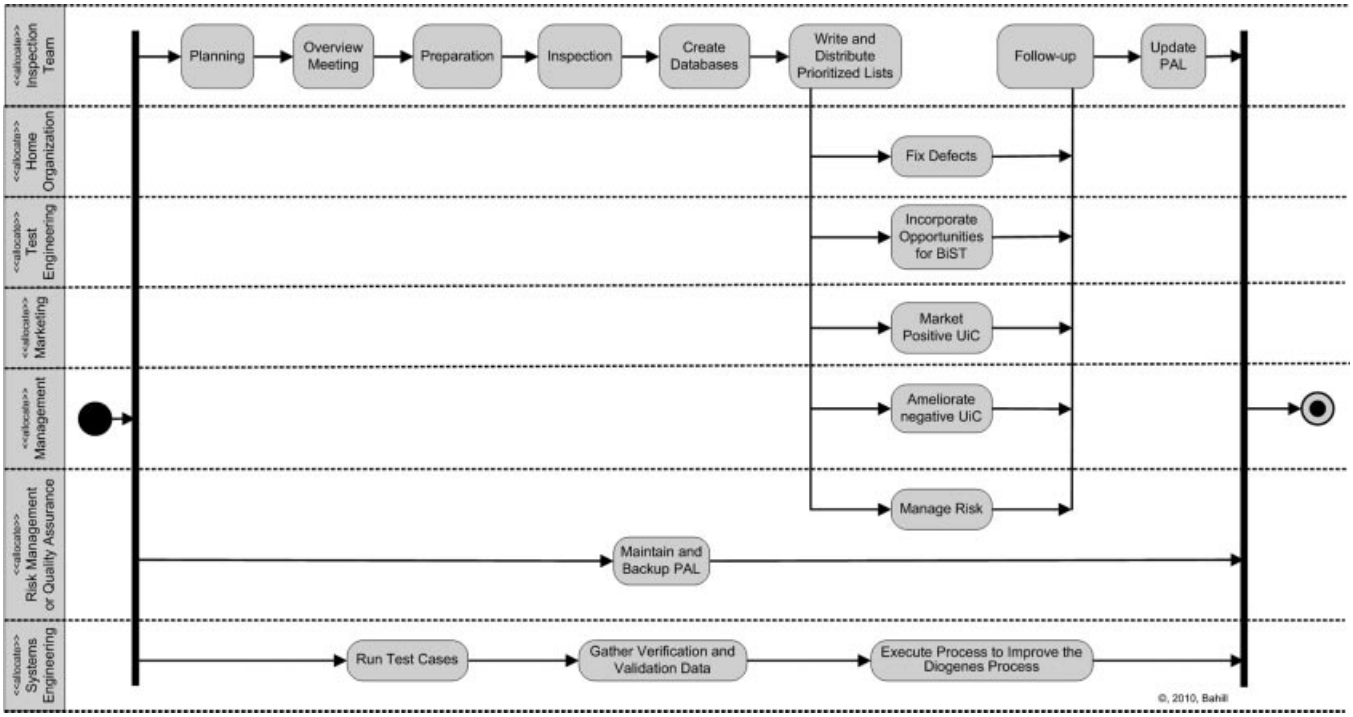

Figure 6. Activity diagram (**act**) for the use case model of Diogenes.

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.

It is often said that we can impose requirements on our system, but we cannot impose requirements on operators, pilots, and other supporting actors. This is still true. However, here we are imposing requirements on the Moderator and members of the InspectionTeam. But that is all right, because they are a part of Diogenes.

Nonfunctional Requirements (NFRs):

NFR3-1. The Moderator shall schedule the inspection meeting for 2 h. The Moderator shall prepare two dozen pages of documentation for each inspection.

Author/owner: Terry Bahill

Last changed: October 4, 2011

The activity diagram of Figure 6 represents the use case model for Diogenes.

3.6. Other Use Case Models

A complete system design for Diogenes would probably have dozens of use cases. So far, we have written four use cases: the two above and Sell Diogenes and Gather Evidence of Verification and Validation. Some other proposed use cases include Ameliorate Negative Unintended Consequences, Improve the Diogenes Process, and Ethics Deliberation.

Journal space constraints have forced us to leave out many of the design documents of Diogenes, such as the design alternatives tradeoff study (25 pages), the sensitivity analysis (4 pages), the risk analysis (5 pages), and the requirements (8 pages). They are available in Bahill [2011].

4. VALIDATION OF DIOGENES USING BIMS

To validate Diogenes, we applied it to an existing, well-documented, system design (which will be called SystemZ) and showed that Diogenes discovered defects, risks, opportunities for BiST, positive UiCs, and negative UiCs of SystemZ. We applied it to the Bahill Illuminance Management System (BIMS) [Bahill, 2010, 2011]. Because this system design existed as a document on Bahill's web site, we used MS Word for the implementation. During actual operation, Diogenes looks at all of SystemZ's documentation. But in this validation test, it did not look at the risk analysis documents of BIMS, because we were trying to show that Diogenes could find these risks.

Note to the reader: This paper is about Diogenes, the process for indentifying unintended consequences: It is not about BIMS. Diogenes is being applied to BIMS in order to verify and validate Diogenes. Therefore, the reader should not focus attention on the details of BIMS.

4.1. BIMS Concept of Operations

Our customer needs a light-management system for the operations rooms of telescope enclosures to be built on Mauna Kea in Hawaii. 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. Consequently, 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, (2) the light inside does not leak out and interfere with the telescopes and (3) the astronomers can see the stars from inside the operations room. As a result, 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 will be considered. BIMS shall include all equipment needed for connecting to the local AC electric power grid.

BIMS risks, design alternatives, sensitivity analysis, test plans, and designs are available in Bahill [2010, 2011].

4.2. Diogenes Applied to BIMS's Use Case

This section contains the main use case of BIMS, in the Times New Roman font. To save space, steps have been elided. Suggestions of Diogenes for defects, risks, opportunities for BiST, positive UiCs, and negative UiCs are marked in the Century Gothic font. This is followed by a punctuation mark (!), a classifier (defect, risk, BiST, pUiC, or nUiC), 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: The operations room of a telescope facility on a remote mountaintop, a renewable-energy electric-generator and a connection to the local electric power grid. Put your mind in that location and ask what bad things could happen up there?

1. The Kilauea volcano could erupt or another far away volcano could erupt, covering the

sky with ash and rendering the solar panels useless! Risk, a volcano erupts.
2. If the native Hawaiians think that BIMS offended Poliahu, the snow goddess of Mauna Kea, then we will have problems! nUiC, BIMS offends Poliahu.
3. Mauna Kea is a remote mountaintop; therefore, costs will be higher than on a normal project. Transportation, electricity, and labor will be more expensive. Backup electric generators will be necessary! 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 University of Arizona purchased and installed a nuclear reactor at a cost of $150K. It is now being decommissioned at a cost of $2M! nUiC, 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.

…

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 1 min and then record the measured illuminance in the room. If this is outside the TBD 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* the Control 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! nUiC, 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! nUiC, 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. Therefore, the connection must be made by the system. BiST shall record the difference in phase and frequency between the inverter output and the electric grid 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 the Control Illuminance During the Night use case.

Specific Requirements: See Daniels and Bahill [2004].

Functional Requirements (FRs):

…

FR1-5. 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.

FR1-6. 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 the efficiency of men and equipment! Risk, reduced efficiency.

FR1-7. BIMS shall execute Built-in Self-Tests (BiST) (derived from BICS company policy).

Nonfunctional Requirements (NFRs):

NFR1-8. BIMS shall maintain the daytime illuminance in the operations room at 400 ± 40 lux ($\approx 40 \pm 4$ fc). Trace to FR1-2, FR1-3, and FR1-4.

NFR1-9. BIMS shall maintain the nighttime illuminance in the operations room at 0.4 ± 0.2 lux ($\approx 0.04 \pm 0.02$ fc). Trace to FR1-2, FR1-3, and FR1-4.

NFR1-10. BIMS shall generate electricity at a cost competitive with commercial electricity costs at that location, after Federal, state, and electric company subsidies, etc. Trace to

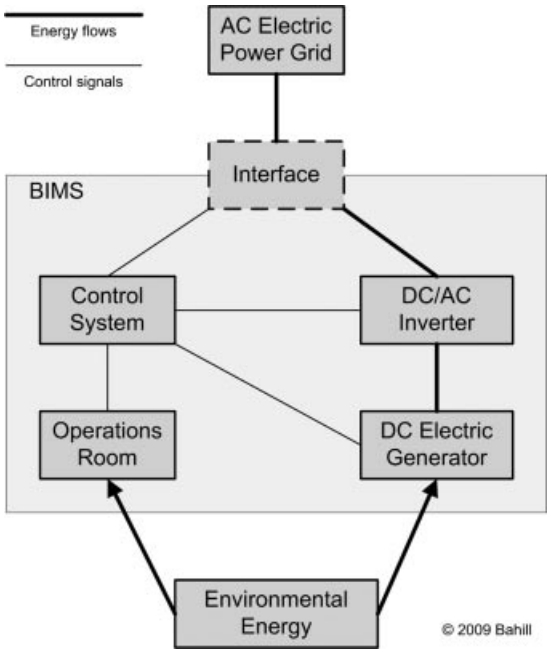

Figure 7. Block diagram showing the scope of BIMS [Bahill, 2010].

FR1-5 and FR1-6. Changes in interest rates, government policies, or electric company rebates would change the economic analysis! Risk, economic conditions change.

Author/owner: Walt Zaharchuk
Last changed: December 3, 2009

4.3. Diogenes Applied to BIMS's Physical Structure

In Figure 7, the absence of energy flow lines to the Control System and the Operations Room is a documentation mistake! Defect, mistake in block diagram.

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 [Haimes, 2009], Figure 8 shows a decomposition of this interface (from Chaves and Bahill [2011]).

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

Studying the blocks of Figure 8, prompted the following question. Which of these components could emit nonvisible electromagnetic radiation that 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! nUiC, electromagnetic radiation interferes with telescopes.

The hierarchy of models of Haimes [2009], the fishbone diagrams of Ishikawa [1990], 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 electric company. The amount of electric energy that is actually bought from or sold will be computed and stored in the database. If the daily averages differ by more than plus or minus 2 standard deviations, then BiST will trigger an alarm! BiST.

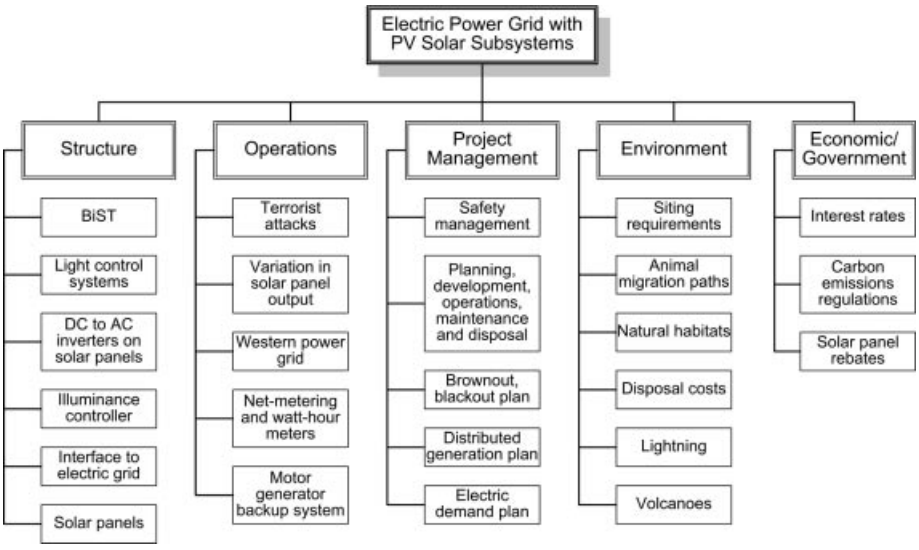

Figure 8. A hierarchy of models for grid-tied solar photovoltaic system.

4.4. Diogenes Examines BIMS's Behavioral Models

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 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 9. Therefore, studies of the Polaris ballistic missile fleet should be reviewed to see if such a regulated illuminance environment would cause undesirable entrainment of human circadian rhythms! Risk, controlled illuminance harms humans.

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*? However, 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 state should transition to Using Stored Energy not to Buying AC Electricity. Transitioning to Buying AC Electricity assumes that the Hawaii Electric Company is always there for us, that is, it is an infinite source or sink.

The equation for *neither* needs two more equal signs, like this:

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

The BIMS energy management controller of Figure 10 can be modeled with these input/output Boolean equations:

Buy = less And empty,
Sell = Not less And full,
Use = less And Not empty,
Store = Not less And Not full.

Therefore, this is a static problem. A static problem does not require a dynamic solution (state machine). 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

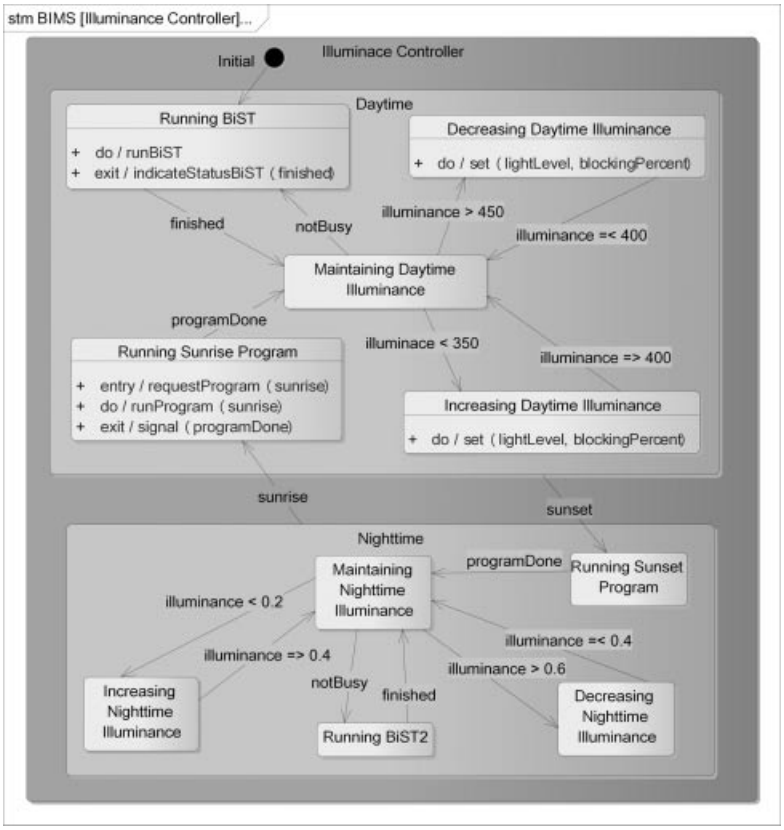

Figure 9. State machine diagram (**stm**) for the BIMS Illuminance Controller. The illuminance limits could be parameters instead of fixed values. The illumination controller increases or decreases the illuminance by commanding a different lightLevel and blockingPercent. Running the sunrise program is 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, and a program for watching PowerPoint presentations [Bahill, 2010].

behavior more simply with the 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 Replacing the state machine with Boolean equations eliminates this risk as well as the hazards and races risk! Defect, state machine is not necessary.

The state machine diagram of Figure 10 could use Boolean BiST indicators for the inputs "more" or "less" and "full" or "empty," and the outputs "storeEnergy" or "useStoredEnergy" and "buyACelectricity" or "sellACelectricity." But it would be just as easy and more meaningful to display the attribute "percentFull" and the state diagram of Figure 10 indicating which state the system is in! BiST, display percentFull.

4.5. Diogenes Examines BIMS's Interfaces and Interconnections

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

What problems could this interconnection of systems cause? If clouds cover the sun, the voltage generated by the solar panels and the illumination in the operations room will drop. The system will command the lights to produce more illuminance, thus drawing 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 the system to become unstable. Furthermore, BIMS will soon deplete its local energy store and will start buying electricity, which 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! nUiC, 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 replanning. Eliminating rebates would affect customer incentives to convert to solar-power generation. Any reduction in consumer incentives to adopt solar energy would have a significant impact on distributed electric generation! Risk, political climate changes.

4.6. Stakeholders

BIMS stakeholders include dealers, architects, contractors, distributors, suppliers, sales people, end users, astronomers, National Optical Astronomy Observatory (NOAO), 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.,

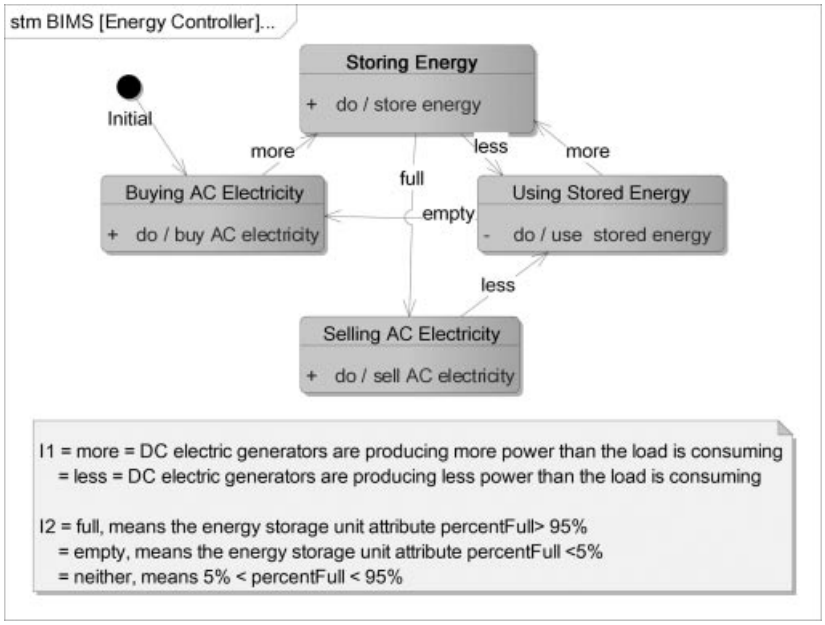

Figure 10. State machine diagram (**stm**) for the BIMS energy management controller [Bahill, 2010].

the environment, and Poliahu the snow goddess of Mauna Kea), environmentalists, the National Science Foundation, and the US Congress.

The US Congress and the National Science Foundation could cut off funding for BIMS! Risk, politics change.

4.7. The Output of "Diogenes Applied to BIMS"

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

4.7.1. List of Defects Detected

This application of Diogenes to BIMS revealed two defects: a lack of energy flows in the block diagram of the scope of BIMS (Fig. 7) and a dynamic solution for a static problem in Figure 10.

The BIMS energy management controller is a static problem. Therefore, it does not require a dynamic solution (state machine). 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 [Botta, Bahill, and Bahill, 2006]. 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.

4.7.2. Prioritized List of Risks to BIMS

These are risks to BIMS, BICS, and the primary users. The highest priority risks are listed first.

Geographical location increases cost. Because Mauna Kea is a remote mountaintop at 13,800 ft, costs might be higher than expected. Transportation, electricity, labor, and the supply chain will be more expensive. Backup electric generators will also be needed.

Sudden drop in generated electricity. Clouds covering the sun 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 and weather prediction systems.

Political climate changes. 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 replanning of their strategies. Elimination of rebates 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. Also, the US Congress or the National Science Foundation could cut off funding for BIMS.

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

Electric company policy changes. Federal laws require electric companies to buy electricity from their customers. However, the electric company could set unfavorable rates. BIMS cost could exceed the local area rate.

High altitude affects humans. High altitude affects human mental and physical processes. However, these are known and 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.

DC to AC Inverter failure. Failure of inverters is the most common hardware failure on present grid-tied photovoltaic solar systems. When an inverter fails, the owner loses a part of the electric generating capacity, but it produces minor harm to the electric company.

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.

Volcanic eruption. BICS should ensure that BIMS is connected to the USGS Volcano Hazards Program, at the Hawaiian Volcano Observatory in case astronomers 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.

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.

TBDs. All items marked as to be determined (TBD) should be collected in a database along with the person responsible for resolving the TBD and date by which it must be resolved.

Diogenes discovered the above dozen risks for BIMS. The original BIMS risk analysis had three dozen risks, but only a dozen were evaluated as important. This study missed: (1) A similar system has already been patented, (2) the commercial AC electric power grid may fail, and (3) observatories face heightened community scrutiny because of their prominent siting. This study disclosed three risks that the original study did not have: (1) Controlled illuminance and high altitude might affect human performance, (2) hazards and races, and (3) TBDs. Some risks were combined in one study or the other (for example, the five cost risks of the original BIMS study were combined into one risk in the Diogenes study), 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.

4.7.3. 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. The amount of electric energy that is actually bought from or sold to the Hawaii Electric Light Co. will be computed and stored in the database. If the daily averages differ by > ±2 standard deviations, then BiST will send an e-mail to the TestEngineer.

BiST shall record the difference in phase and frequency between the inverter output and the electric grid when a connection is made. These data shall be analyzed statistically.

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

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 the event of failure.

4.7.4. List of Positive UiCs

This application of Diogenes to BIMS revealed 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 Tucson Electric Power's Springerville Solar Generating Station consumed 12 MWh AC/kW DC generation capability installed. This facility reduces the carbon footprint by 36 tons CO_2/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*.

4.7.5. Prioritized List of Negative UiCs

Negative UiCs are bad outcomes that BIMS can create for other entities. The highest priority negative UICs are listed first.

Destabilizing the electric grid. (1) Connecting to the commercial AC electric grid will cause problems. Presume that BIMS is in the state of Selling AC Electricity in the state machine of Figure 10, 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 the system to become unstable. (2) When the sun is blocked by clouds, BIMS will quickly deplete its 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 two potential stability problems, we recommend that the project manager start a detailed simulation of these systems to investigate potential instabilities.

Increased costs to electric company. BIMS draws energy from the commercial AC electric grid. Therefore, the Hawaii Electric Light Co. will have to buy backup power generators that can provide the total load of BIMS at any time, even if the sun is obscured by clouds.

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 to 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.

Destabilizing the solar panel economy. If new technology dramatically drove down the cost of solar panels, the number of customers who install solar panels would increase. 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.

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.

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.

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.

This analysis discovered seven negative UiCs of BIMS. The original BIMS documentation discovered none.

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.

5. VERIFICATION AND TEST 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 InspectionTeam.

This is the beginning of the test of the main success scenario of the Perform Formal Inspection use case. The Times New Roman font is used for parts of the use case and the Century Gothic font is used for actions the TestEngineer must do.

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

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 InspectionTeam. 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 2 h 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.

The TestEngineer interviews the members of the InspectionTeam, to ensure that they understand what their responsibilities are, the inspection process that they are supposed to follow, and to report the hours they spent in preparation and that they should be looking for five things simultaneously: defects, risks, opportunities for BiST, positive UiCs, and negative UiCs of SystemZ. If a team member does not know something, then TestEngineer explains it to the member. 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 2 h.

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

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

6. **Prioritized lists.** The Moderator and the SystemsEngineer consolidate and edit the five databases to create five prioritized [Botta and Bahill, 2007] lists.

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

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

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

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

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

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 they will do with their lists. This

tests **FR3-9**. TestEngineer documents these meetings.

7. PAL. 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. Each of the other owners will know what to do with his list.

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 completing 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. Update PAL. 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 and the Program Manager. 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.

6. DESIRABLE FEATURES FOR A UIC PROCESS

Table II compares features that are desirable in a process to be used for finding unintended consequences to alternative well-known systems engineering techniques. A capital X indicates a strong implementation, a lowercase x indicates some implementation, and a blank indicates little implementation.

Our analysis has shown that the following features are desirable in a process to be used for finding unintended consequences.

Looks Forward in Time: The UiC process must look forward in time. Existing tools can be modified to look forward or backward in time. But for tools that have been in use for decades, where there is substantial literature and knowledge, switching would be confusing.

Table II. Desirable Features for a UiC Process

Features	Alternative techniques								
	Diogenes	Problem Statement	Requirements Discovery	Risk Analyses	Tradeoff Studies	Formal Inspections	Sensitivity Analyses	Test Plan Generator	Casual Analysis and Resolution
Looks Forward in Time		X	X	X	x			X	
Looks Outside	X			x	x				X
Security	X	X	X	X	X	X	X	X	X
Built-in Self-test	X		x		X		x	x	
Uses Existing Entities	X	X	x	x	X	X	x	X	X
Reusability	X		x	x	X	x	X		
Described as a system	X		x			x	x		
Defined Inputs	X	X	x			x	x		x
Defined Outputs	X		x	x		x	x		
Defined States	X		x	x					x
Generate Test Plans	X							X	
Levels	X	X	X	X	X	x	X	x	X
Identifies Root Causes						x			X
Stakeholders	X	X	X	x					
Primary lifecycle phase when it should be used	M	I	I	I	SD	SD	SD	I&T	ORR

Code for When: I = Inception, SD = System Design, M = Manufacturing, I&T = Integration and Test, and ORR = Operations, Retirement and Replacement.

Looks Outside: The UiC process must look at how the system being designed (SystemZ) will affect *other* systems.

Security: The customer's designs must be kept secure. Security must be a prime consideration from the very beginning of the design. This means, for example, that the customers' designs must not be put on the Internet.

Built-in Self-Test: Ease of implementing of a Built-in Self-Test (BiST) is important. BiST could be passive, where the BiST process monitors outputs and checkpoints and displays status or BiST could be active, where the BiST system generates signals and applies them to the system inputs.

Uses Existing Entities: The UiC process should use existing data and people: It should not require additional organizational structure or funding channels.

Reusability: Nondevelopmental products must 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 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, (1) each UiC alternative process will be given points for use of Commercial off the Shelf (COTS) products and will be deducted points for use of custom-made products and (2) each alternative will be given points if its analysis and results are likely to be reused in other systems.

Described as a System: The UiC process should be described as a system, which means describing the inputs, outputs, states, next state function, and readout function [Wymore, 1993].

Defined Inputs: The UiC process should accept formal designs in the form of functional flow block diagrams (Figs. 2 and 3), use cases (Sec. 3.5), SysML diagrams (Fig. 6), block diagrams (Fig. 7), HHMs (Fig. 8), UML diagrams (Figs. 5, 9, and 10), and other common systems engineering tools [Bahill et al., 1998]. It should accept both visual and verbal input.

Defined Outputs: The UiC process should produce the five work-products: defects, risks, opportunities for BiST, positive UiCs, and negative UiCs.

Defined States: The behavior of the process should be described with state machine diagrams (Figs. 9 and 10) and activity diagrams (Fig. 6) [Botta, Bahill, and Bahill, 2006].

Generate Test Plans: The process should produce preliminary test plans and test procedures. This will force consideration of test into early phases of design.

Levels: The process should be appropriate for high-level artifacts and also for low-level detailed artifacts [Bahill et al., 2008]. Mappings should be defined between levels.

Identifies Root Causes: The process could identify the root cause of failure.

Stakeholders: The process should encourage discussion of the needs of a wide variety of stakeholders.

Primary life cycle phase when it should be used: All system design processes are iterative. But there is usually a phase of the system life cycle when each process consumes the most resources.

Once again with reference to Table II, we will consider the following well-known alternative techniques.

1. Diogenes, the process presented in this paper
2. Problem statement portion of the SIMILAR process [Bahill and Gissing, 1998], brainstorming [Brassard and Ritter, 1994], the five whys [Ohno, 1988], concept mapping [Novak and Cañas, 2008], and affinity diagrams
3. Requirements discovery [Bahill and Dean, 1999, 2009]
4. Risk analyses [Arnauld and Nicole, 1662; Bahill and Smith, 2009; Haimes, 2009; Chaves and Bahill, 2011], including failure modes and effects analyses (FMEAs) [Carbone and Tippett, 2004; Clausen, and Frey, 2005]
5. Tradeoff studies [Daniels, Werner, and Bahill, 2001], including commercial products such as Kepner-Tregoe®
6. Formal inspections, such as Fagan inspections [Fagan, 2011]
7. Sensitivity analyses [Hsu, Bahill, and Stark, 1976; Karnavas, Sanchez, and Bahill, 1993; Smith et al., 2008]
8. Test plan generator [Bahill, 2010, 2011]
9. Casual analysis and resolution, including cause and effect analyses [Juran, 1989], fishbone diagrams [Ishikawa, 1990], and root cause analysis.

Not surprisingly, the features that are desirable in a process depend on the purpose of the process. Table II shows how well existing processes satisfy the purpose of discovering UiCs. We found that we could not use an existing process to search for UiCs. However, we reused components of other process in order to create Diogenes. Table II shows features and alternative techniques that were helpful in designing a process to search for UiCs.

7. SUMMARY

The output of Diogenes is the following five prioritized lists: (1) the list of defects in development documents that will be given to the Author/Designer for resolution; (2) the list of risks that could adversely affect SystemZ that will be given to the director of Risk Management; (3) the list of opportunities for Built-in Self-Test (BiST) that will be given to the director of Test Engineering; (4) the list of positive UiCs that will be given to the director of Marketing; (5) the list of negative UiCs that will be given to the director of Management and to the company lawyers. Diogenes is not intended to fix any of these problems. It passes the fixing task to these five directors.

The take-home message of this paper is that the systems engineer is responsible for discovering UiCs of systems that are being designed. This is important because negative UiCs can be serious. However, UiCs can be anticipated. Diogenes can help discover defects and risks and at the same time help identify opportunities for BiST, positive UiCs, and negative UiCs. Therefore, it will not cost extra money to use Diogenes to search for negative UiCs.

ACKNOWLEDGMENTS

We thank the Systems Engineering Brain Trust, George Dolan, Bob Sklar, Brad Sowers, Bruce Gissing, Al Chin, and Gary Lingle for working through the development of Diogenes, INCOSE Fellows Mark Maier and Scott Jackson for valuable comments on the manuscript, John Hayhurst of IBM for suggesting the consideration of moonlight in the BIMS project, and Walt Zaharchuk of Lutron Electronics for financial support. We thank Sparx Systems for a site license for Enterprise Architect.

REFERENCES

A. Arnauld and P. Nicole, Logic, or, the art of thinking: Containing, besides common rules, several new observations appropriate for forming judgment, 5th edition, translated from French in 1996 by Jill Vance Buroker, Cambridge University Press, Cambridge, 1662.

A.T. Bahill, Design and testing of an illuminance management system, ITEA J 31(1) (2010), 63–89.

A.T. Bahill, SIE-454/554a, The Systems Engineering Process, http://www.sie.arizona.edu/sysengr/sie554/index.html, accessed August 2011.

A.T. Bahill and R. Botta, Fundamental principles of good system design, Eng Management J 20(4) (2008), 9–17.

A.T. Bahill and F.F. Dean, "Discovering system requirements," Handbook of systems engineering and management, A.P. Sage and W.B. Rouse (Editors), Wiley, Hoboken, NJ, 1st edition, 1999, pp. 175–220, 2nd edition, 2009, pp. 205–266.

A.T. Bahill and B. Gissing, Re-evaluating systems engineering concepts using systems thinking, IEEE Trans Syst Man Cybernet C Appl Rev 28(4) (1998), 516–527.

A.T. Bahill and E.D. Smith, An industry standard risk analysis technique, Eng Management J 21(4) (2009), 16–29.

A.T. Bahill , M. Alford, K. Bharathan, J. Clymer, D.L. Dean, J. Duke, G. Hill, E. LaBudde, E. Taipale, and A.W. Wymore, The design methods comparison project, IEEE Trans Syst Man Cybernet C Appl Rev 28(1) (1998), 80–103.

A.T. Bahill, F. Szidarovszky, R. Botta, and E.D. Smith, Valid models require defined levels, Int Gen Syst 37(5) (2008), 553–571.

Y. Bar-Yam, A mathematical theory of strong emergence using multiscale variety, Complexity, 9(6) (2004), 15–24, www.necsi.edu/research/multiscale/MultiscaleEmergence.pdf.

R. Botta, Z. Bahill, and A.T. Bahill, When are observable states necessary? Syst Eng 9(3) (2006), 228–240.

R. Botta and A.T. Bahill, A prioritization process, Eng Management J 19(4) (2007), 20–27.

M. Brassard and D. Ritter, The Memory Jogger II, A pocket guide of tools for continuous improvement & effective planning, GOAL/QPC, Salem, NH, 1994.

T.A. Carbone and D.D. Tippett, Project risk management using the project risk FMEA, Eng Management J 16(4) (2004), 28–35.

A. Chaves and A.T. Bahill, Risk analysis for incorporating photovoltaic solar panels into a commercial electric power grid, University of Arizona, 2011.

D. Clausen and D.D. Frey, Improving system reliability by failure-mode avoidance including four concept design strategies, Syst Eng 8(3) (2005), 245–261.

A. Cockburn, Writing effective use cases, Addison-Wesley, Reading, MA, 2001.

J. Daniels and A.T. Bahill, The hybrid process combines traditional requirements and use cases, Syst Eng 7(4) (2004), 303–319.

J. Daniels, P.W. Werner, and A.T. Bahill, Quantitative methods for tradeoff analyses, Syst Eng 4(3), (2001), 190–212 [correction: Syst Eng 8(1) (2005), 93].

M. Fagan, Fagan inspections and continuous process improvement, http://www.mfagan.com/, last accessed January 11, 2011.

Y.Y. Haimes, Risk modeling, assessment, and management, 3rd edition, Wiley, Hoboken, NJ, 2009.

F.K. Hsu, A.T. Bahill, and L. Stark, Parametric sensitivity of a homeomorphic model for saccadic and vergence eye movements, Comput Prog Biomed 6(1976), 108–116.

K. Ishikawa, Introduction to quality control, Chapman & Hall, London, 1990 [Engl transl by J.H. Loftus of Dai-3-pan Hinshitsu Kanri Nyumon, 3rd edition, JUSE Press, Tokyo, Japan, 1990].

S. Jackson, Architecting resilient systems: Accident avoidance and survival and recovery from disruptions, Wiley, Hoboken, NJ, 2010.

I. Jacobson, Use cases in large-scale systems, road to the unified process, Cambridge, University Press, Cambridge, 2000.

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

W.J. Karnavas, P. Sanchez, and A.T. Bahill, Sensitivity analyses of continuous and discrete systems in the time and frequency domains, IEEE Trans Syst Man Cybernet SMC-23(2) (1993), 488–501.

J.A. List, M. Margolis, and D.E. Osgood, Is the endangered species act endangering species? Working Paper 12777, NBER Working Paper Series, National Bureau of Economic Research, Cambridge, MA, December 2006, http://www.nber.org/papers/w12777, for a free copy, http://graphics8.nytimes.com//images/blogs/freakonomics/pdf/FreakPDF4.pdf.

R.K. Merton, The unanticipated consequences of purposive social action, Amer Sociol Rev 1(6) (1936), 894–904.

R. Norton, "Unintended consequences," The Fortune encyclopedia of economics, David R. Henderson (Editor),Warner Books, New York, 1993, pp. 92–94, http://oll.libertyfund.org/title/1064 accessed March 27, 2011 [edited and reprinted at http://www.econlib.org/library/Enc/UnintendedConsequences.html].

J.D. Novak and A.J. Cañas, The Theory underlying concept maps and how to construct them, Technical Report IHMC CmapTools 2006-01, Rev 01-2008, Florida Institute for Human and Machine Cognition, Pensacola, FL, 2008, http://cmap.ihmc.us/Publications/ResearchPapers/TheoryUnderlyingConceptMaps.pdf.

T. Ohno, Toyota production system: Beyond large-scale production, Productivity Press, New York, 1988 [Transl of: Toyota seisan hõshiki, Diamond, Tokyo, 1978].

G. Santayana, "Flux and constancy in human nature," Reason in common sense, The Life of Reason, Vol. 1, Charles Scribner's Sons, New York, 1905, Chap. XII.

S. Sheard and A. Mostashari, Principles of complex systems for systems engineering, Syst Eng 12(4) (2009), 295–311.

A. Smith, An enquiry into the nature and causes of the wealth of nations, 2 vols., R.H. Campbell and A.S. Skinner (Editors), Oxford University Press, Oxford, 1975, IV.ii.9, 2: 456,1776.

E.D. Smith and A.T. Bahill, Attribute substitution in systems engineering, Syst Eng 13(2) (2010), 130–148.

E.D. Smith, Y.J. Son, M. Piattelli-Palmarini, and A.T. Bahill, Ameliorating mental mistakes in tradeoff studies, Syst Eng 10(3) (2007), 222–240.

E D. Smith, F. Szidarovszky, W.J. Karnavas, and A.T. Bahill, Sensitivity analysis, a powerful system validation technique, Open Cybernet Systemics J 2 (2008), 39–56, http://www.bentham.org/open/tocsj/openaccess2.htm, DOI: 10.2174/1874110X00802010039.

A.W. Wymore, Model-based systems engineering, CRC Press, Boca Raton, FL, 1993.

Terry Bahill is Professor Emeritus of Systems Engineering at the University of Arizona in Tucson. He received his Ph.D. in electrical engineering and computer science from the University of California, Berkeley, in 1975. Bahill has worked with BAE Systems in San Diego; Boeing Information, Space and Defense Systems in Kent, WA; Idaho National Laboratory in Idaho Falls; Hughes Missile Systems in Tucson, AZ; Lockheed Martin Tactical Defense Systems in Eagan, MN; Lutron Electronics in Coopersburg, PA; Raytheon Missile Systems in Tucson, AZ; and Sandia Laboratories in Albuquerque, NM. For these companies he presented seminars on systems engineering, worked on system development teams and helped them describe their systems engineering process. He holds a U.S. patent for the Bat Chooser, a system that computes the Ideal Bat Weight for individual baseball and softball batters. He received the Sandia National Laboratories Gold President's Quality Award. He is a Fellow of the Institute of Electrical and Electronics Engineers (IEEE), of Raytheon Missile Systems, of the International Council on Systems Engineering (INCOSE), and of the American Association for the Advance of Science (AAAS). He is the Founding Chair Emeritus of the INCOSE Fellows Selection Committee. His picture is in the Baseball Hall of Fame's exhibition "Baseball as America" [http://www.sie.arizona.edu/sysengr/].