International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

REAL-TIME SYSTEM USING THE BEHAVIORAL PATTERNS ANALYSIS (BPA) Approach: The Train Traffic Management System (TTMS)

Dr. Assem El-Ansary, CEO
Emergent Technologies USA, Inc.
USA

Ass.Prof. M.M.Elgazzar
Faculty of Computing & Information
EGYPT

Abstract: -- This paper illustrates the event-oriented Behavioral Pattern Analysis (BPA) modeling approach using the Train Traffic Management System. In BPA, events are considered the primary objects of the world model. Events are more effective alternatives to use cases in modeling and understanding the functional requirements. The Event defined in BPA is a real-life conceptual entity that is unrelated to any implementation. The BPA Behavioral Patterns are temporally ordered according to the sequence of the real world events.

The major contributions of this research are:

- ***The Behavioral Pattern Analysis (BPA) modeling methodology.***
- ***Validation of the hypothesis that the Behavioral Pattern Analysis (BPA) modeling methodology is a more effective alternative to Use Case Analysis (UCA) in modeling the functional requirements of Human-Machine Safety-Critical Real-time Systems.***
- ***The development of an interactive software tool (DECISION), which is based on a combination of the Analytic Hierarchy Process (AHP) and the ELECTRE Multi-Criteria Decision Making (MCDM) methods. The DECISION software tool was used to process the assessment results of the case studies.***

Key words: analysis, modeling methodology, software modeling, event-oriented, behavioral pattern, use cases.

1. INTRODUCTION

Experience reports problems with Use Cases such as [GRAHAM 95]:

1. *The lack of a Use Case formal specification has led to a proliferation of modeling methodologies all calling themselves 'use case' based.*
2. *Lack of atomicity has been the reason for generating hundreds of use cases for some simple applications.*
3. *The lack of a notion of atomicity has made the measurement of a project's task complexity, by counting the use cases, unreal.*
4. *The absence of the notion of triggering events and business goals in determining use cases is a serious deficiency.*
5. *There is a problem with the phrase use case itself.*

A major problem in the use case approach is its tendency to focus on the solution rather than the problem. Jacobson defined use case as "a behaviorally related sequence of transactions in a dialogue with the system" [JACOBSON 91]. The processing of transactions, or operations, or use cases is what the machine does. It is part of the solution, not part of the problem [JACKSON 95].

The concluding statement of the "Question Time! About Use Cases" Panel of the OOPSLA'98 Conference by Ian Graham [OOPSOLA 98] was "There is a need for another modeling methodology with a sound theoretical basis and a precise definition." This need is what this research problem area is about.

In addition to the problems with the use cases [OOPSLA 98][JACKSON 95] that were described briefly above, several additional problems were identified during this research [ELANSARY 02, 05]. The following is a discussion of these problems:

- *The types of interactions are: interactions among users, interactions between users and the system, and interactions among the different components of the system. Yet, use cases describe only the users' interaction with the system. This is just one type of interaction.*
- *Because use cases are used to identify the objects, if use cases do not describe all of the interactions, the resulting object (class) model may be incomplete.*
- *As a result of this class model incompleteness there will then be an incomplete description of the interactions, and so the sequence diagram may be incomplete.*

- *Using natural language in use cases description, with the absence of any semantic structure such as alternation or repetition, increases the risks of ambiguity, incompleteness, and inconsistency.*
- *Use case approach, as well as most of the popular approaches such as Booch, OMT, UML, etc., are using the state diagrams to complement the system behavior modeling. One may argue that any missing interaction description may be captured via the state diagrams. However a missing object or interaction is unlikely to be captured and explicitly represented in these diagrams. Also, a state diagram describes an individual object's response to specific events rather than objects interaction. Hence, objects interaction must be reconstructed from the analysis of groups of diagrams. Such a task is complex and error-prone.*

In conclusion, if the analyst misinterpreted or neglected some structural or behavioral aspects, the resulting conceptual model will not be a good representation or understanding of the real world. The resulting software solution system built from the model may not demonstrate the correct behavior or may ungracefully terminate. The end result might be the loss of opportunities in using business systems, serious damages in embedded systems, or the loss of lives in using a safety-critical system.

This paper reports on Behavioral Pattern Analysis (BPA), which is a more effective alternative to use cases in modeling and understanding the functional requirements [ELANSARY 02]. BPA is an event-oriented modeling methodology in which events are considered the primary objects of the world model. While the term Event is used in UML, and in almost all of the other modeling methodologies, to mean an occurrence of stimulus that can trigger a state transition, the Event defined in BPA is a real-life conceptual entity that is unrelated to any implementation. In the BPA modeling methodology, the BPA Behavioral Pattern, which is the template that one uses to model and describe an event, takes the place of the Use Case in the UML Use Case View. The BPA Behavioral Patterns are temporally ordered according to the sequence of the real world events.

2. ILLUSTRATING BPA THROUGH THE TRAIN TRAFFIC MANAGEMENT MANAGMENT SYSTEM (TTMS)

The two main functions of the TTMS [BOOCH 07] are to route trains, and monitor trains. The following describe the Functional requirements:

- *Route Train: Establish a train plan that defines the travel route for a particular train.*
- *Plan Traffic: Establish a traffic plan that provides guidance in development of train plans for a time frame and geographic region.*
- *Monitor Train Systems: Monitor the onboard train systems for proper functioning.*
- *Predict Failure: Perform an analysis of train systems' condition to predict probabilities of failure relative to the train plan.*
- *Track Train Location: Monitor all train traffic using TIMS resource and Navstar Global Positioning System (GPS).*
- *Monitor Traffic: Monitor all train traffic within a geographic region.*
- *Avoid Collision: Provide the means, both automatic and manual to avoid train collisions.*
- *Log Maintenance: Provide the means to log maintenance performed on trains.*

The Nonfunctional requirements are:

- *Safely transport passengers and cargo.*
- *Support train speeds up to 250 miles/hr.*
- *Interoperate with the traffic management systems of operators at the TIMS boundary.*
- *Ensure maximum reuse of and compatibility with existing equipment.*
- *Provide a system availability level of 99.99%.*
- *Provide complete functional redundancy of TIMS capabilities.*
- *Provide accuracy of train speed within 1.5 miles per hour.*
- *Respond to operator inputs within 1.0 seconds.*
- *Have a designed-in capability to maintain and evolve TIMS.*

The Constraints are:
- *Meet national standards, both government and industry.*
- *Maximize use of commercial-off-the-shelf (COTS) hardware and software.*

3. RESEARCH THESIS

The specific thesis is that the proposed Behavioral Pattern Analysis (BPA) approach is more effective than the Use Case Analysis (UCA) approach at modeling the functional requirements of Interactive Safety-Critical Real-time Systems. To validate that the

BPA approach is more effective than the Use Case approach, sixteen Subject Matter Experts were given two case studies that are modeled using the two approaches and were asked to evaluate the models using the Safety, Repeatability, Unambiguity, Completeness, Consistency, Modifiability, and Traceability as the effectiveness criteria. The following subsection presents a summary of the research approach.

4. THE BPA REQUIREMENTS DEVELOPMENT PROCEDURE

The following is an outline of the BPA functional requirements development procedure:

I. Identify the problem at the highest level of abstraction (e.g. The Mission Statement and Operating Requirements).

II. Identify the scope of the requirements (problem) from the Originating Requirements.

III. Analyze the Originating Requirements to identify the Critical Constraints (e.g. Safety) and/or the Utility Requirements.

IV. Decompose the scoped problem (from step II) into Main Events based on the Mission and Operating Requirements (Step I).

V. Using the identified Main Events, draw the High Level Event Hierarchy Diagram (Figure 3).

VI. Decompose these identified Main Events into smaller and simpler events represented as Episodes (Composite Events) with clear boundaries[1]. An Episode Boundary at this stage may be marked with Location / Loci of Control and Effect. Add additional levels to the Event Hierarchy Diagram (Event Hierarchy Sub-Diagrams). For complex problems, it is often helpful to extract these sub-diagrams and analyze them. Detailed level event hierarchy diagrams are drawn as necessary.
   - The Event Decomposition Heuristics at this stage is ‘ One Agent and One Location’

VII. For each identified main event (from step IV) draw an Event Thread Diagram (Figure 4)
   - Starting with the Main Events, as initial composite events, recursively decompose the composite events into Basic Events
   - The Event Decomposition Heuristics at this stage is ‘ One Agent, One Location, One Motion Direction, and One Time Interval’.
   - Group Basic Events by their Location / Loci of Control and Effect. Draw a frame box around these Basic Events

VIII. Refine and transform the above Basic Events into their corresponding BPA Behavioral Patterns (Figure 5 represents a Behavioral Pattern sample).

IX. Using the Event Thread Diagrams from step VIII, draw the Temporal/Causal Constraint Diagrams by adding the temporal constraints alongside the associations and identifying the enable/causal relationships in each corresponding Event Thread Diagram (Figure 8).

X. Using the Critical Constraints (e.g. Safety), identify the critical events, identify all possible ways of each critical event’s failure, and draw the Critical Event Analysis Diagram (Figure 9).

XI. Using the BPA Event Patterns and the Critical Event Analysis Diagrams, identify any missing requirements that are necessary to satisfy the critical constraints. If these missing requirements are not in the Originating Requirements document, develop a Derived Requirements document and get users approval on this document.

XII. Using the Missing Requirements (from step XI), refine the Event Hierarchy Diagram (from step VI), the Thread Diagrams (from step VII), and the Temporal Constraint Diagram (from step IX) as necessary. Draw additional Event Thread Diagrams for identified critical events as necessary. The figure (Figure 1) below illustrates the iterative and incremental development process that is used in the BPA approach.

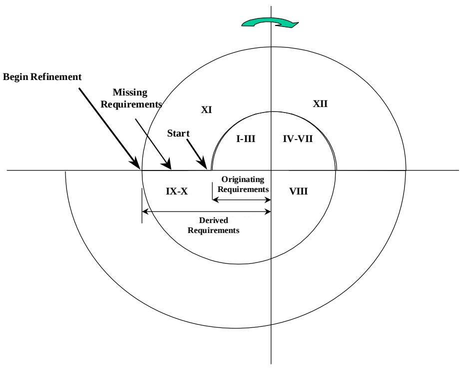


Figure 1 The BPA Modeling Process

XIII. Using the BPA Event Patterns (from step VIII), identify the candidate Classes from the Event Roles (Participants) and Instrument. Draw the Class Diagram (figure10).

XIV. To illustrate the relationship between Events and States, optionally, using the BPA Behavioral Patterns, draw the Event/State History Chart (Optional – not shown) that includes the States before and after each Event for each identified Class whose instance is a participant in that Event. This chart helps in developing the state model during the design stage. The above procedure illustrates the BPA functional requirements development procedure Figure 2 depicts the flow of the modeling activities for the BPA procedure.

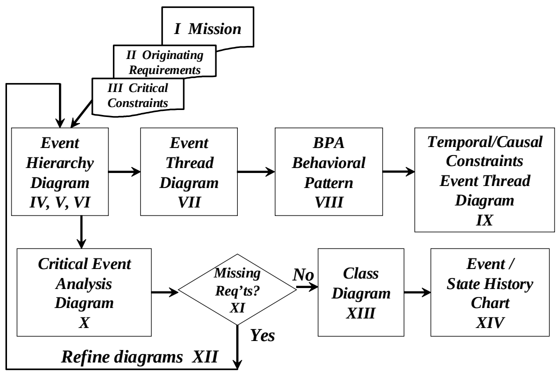


Figure 2 Requirements Development Procedure

4.1. Event Hierarchy Diagram (EHD)

Because there are many levels of requirements details analysts need techniques to structure the excessive amount of requirements information that surfaces. Event Hierarchy is used to model the events at different levels of abstraction (event decomposition). As per steps IV, V, and VI in the BPA procedure, a general problem with decomposition is when to stop the decomposition. To overcome this problem, **the decomposition heuristic** used in an Event Hierarchy Diagram (EHD) is one agent and one location. Using this heuristic, the leaf events in an Event Hierarchy are usually Simple Sequence Events. In other words, a leaf event is usually a set of Basic Events (atomic events) sequenced into episode[1]. The episode is marked with a location boundary. The following is the MCS detailed Event Hierarchy Diagram:

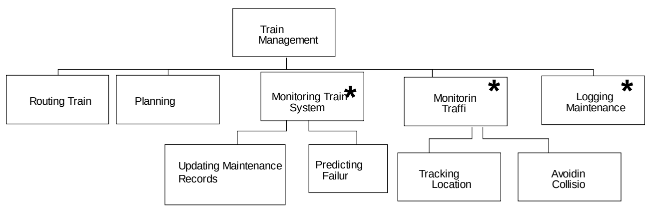


Figure 3 Event Hierarchy – Train Traffic Mgmt

Using the identified main events, the high level EHD diagram (or the first level in a detailed EHD diagram) is drawn. Each main event is then decomposed further until one arrives at leaf events, each of which has one location or one locus of effect and control and one agent.

In order to model the sequence of events (and show the location / loci of control and effect view, or the temporal / causal constraints), one uses the event thread diagrams as shown in the next subsections.

4.2. Event Thread Diagram (ETD)

In BPA, as per step VII, an Event Thread Diagram (ETD) is drawn for each main event, and optionally drawn for any other event, subordinate to main event, depending on its complexity or its critical nature.

A Basic Event is defined as an event that cannot be decomposed into another set of events (atomic event). The heuristic used in decomposing an event into its basic events is one agent, one location, one time interval, and one motion direction if the event involves any motion. The ETD, which one draws for an event, shows the sequence of the basic events of that event.

International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

Event Thread Diagram: 'Train Traffic Management'

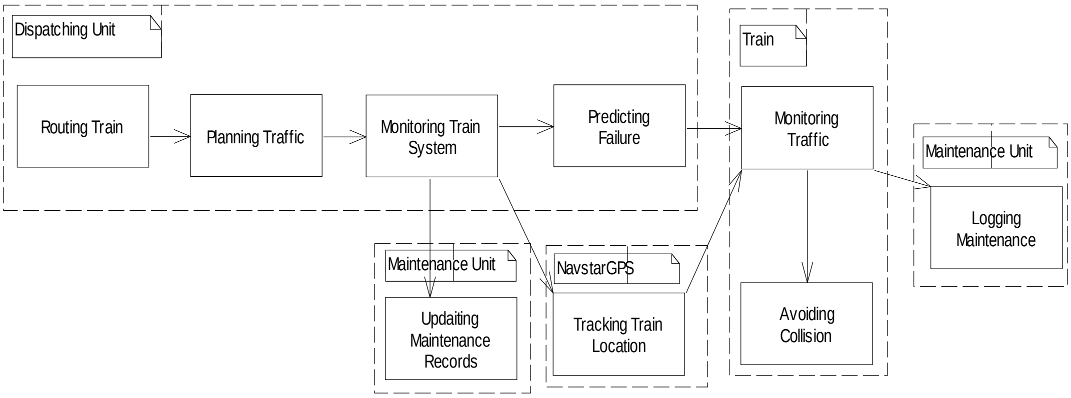


Figure 4 Event Thread – Train Traffic Management System

4.3. Behavioral Patterns

As explained earlier in step VIII, the research goal is to develop a requirements definition mechanism (BPA Pattern) that describes the What, Who, How, When, Where and Why.

BPA BEHAVIORAL PATTERN - EXAMPLE
Event (WHAT?) ***TrackingTrainSystem***
Actions
1. **2.**
3. **4.**
5. **6.**
Agent a: DispatchingUnit
- Initial State: tracking
- Final State: monitoring

Affected p: Train
- Initial State: tracked
- Final State: monitored

Modality (HOW?)
Instrument
i: NavastarGPS
Circumstances
Manner
m:Critical
Condition
c1: Tracked c2:
Effect
f1: Monitored f1:
Date/Time (WHEN?)
t: After EstablishingTrafficPlan
Place (WHERE?)
Location
l: DispatchingUnit
Path
Motion
m: _
Direction
d: _
Rationale (WHY?)
Goal g: Tracking Train
Mental State bdi:
Caused-By ***e': DispatchingUnit***
End;

Figure 5 BPA Pattern – Train Traffic Management

International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

4.4. Introducing Time

The key intuitions motivating the introduction of time are:

- Events take time. Yet, in most of the popular Object-Oriented Modeling methodologies such as OMT and UML, time is neglected in the event definition.
- Multiple events may occur at the same time, and could be unrelated, cooperating, or interfering with each other.
- Events may have temporal constraints. They may overlap, start or finish together, occur together, or disable (disjoint) each other. BPA uses the time intervals' relations that are described in the Interval Algebra framework [ALLEN 83] to model the temporal relationships between events. In this Interval Algebra framework, seven basic relations can hold between time intervals. Figure 6 illustrates these basic relations for arbitrary events x and y.
- Figure 6 illustrates the Interval Algebra Relations.

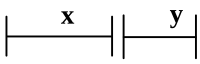
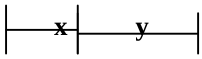
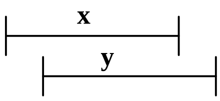
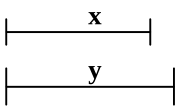
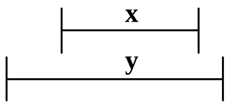
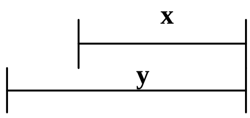
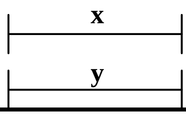
REL	SYM	MEANING
x before y	**b**	
x meets y	**m**	
x overlaps y	**o**	
x starts y	**s**	
x during y	**d**	
x finishes y	**f**	
x equals y	**eq**	

Figure 6 Time Interval Algebra – Temporal Relations

4.5. Introducing Enable / Cause Relationships

The introduction of the Enable[1] / Cause relationships between events will enable the analyst to do cause effect analysis and reason about any possible failure of the system.

[1] 'Enable' is defined in the American Heritage Dictionary as: "..To supply with the means, knowledge, or opportunity; make able: *a hole in the fence that enabled us to watch; techniques that enable surgeons to open and repair the heart."*

International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

Temporal Relations

b Before
m Meets
o Overlaps
d During
s Starts
f Finishes
eq Equals
i Inverse

Figure 7 Time Interval Algebra - Temporal Relations Notation

In the Temporal Constraint Diagram, as described in steps IX, and X, the temporal relations, that are displayed in Figure 8, are written alongside the sequence relationships between the events to represent the possible timing at which these events can occur.

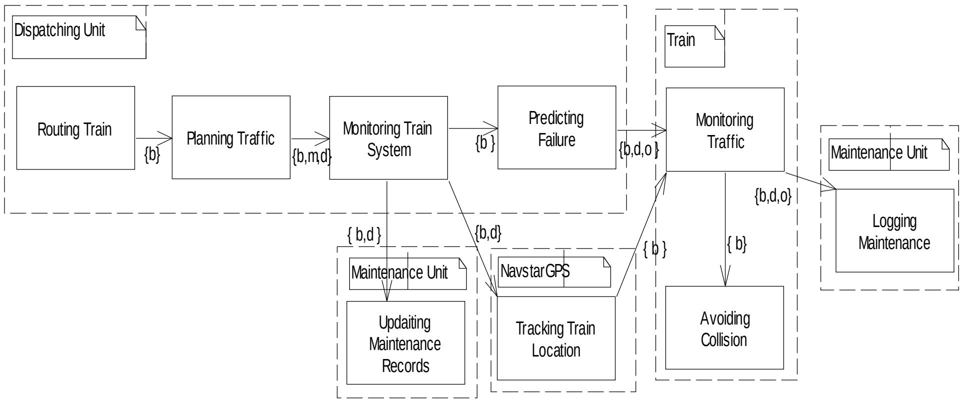


Figure 8 Temporal Constraint Diagram – Train Traffic Management

4.6. Failure Issues

The following is a list of reasons of possible failures in responding to events:

- Occurrence of a relevant event which the system does not handle
- Event rate exceeding the system's capacity
- Unsuccessful detection and acquisition of all events including manually captured events
- Non-capturing of all information triggered by event
- Failure across man-machine interface
- Failure of Software, Hardware, or Human.

The ability to provide requirements specification for safe behavior is very limited using the current modeling methodologies. Neither a safety analysis (anterior analysis) nor accident analysis (posterior analysis) can be achieved efficiently without event analysis. As will be explained below, the BPA modeling methodology provides the Critical Event Analysis (defined below) as an efficient solution to this problem.

4.6.1. Critical Events Analysis

The requirements should correctly reflect the critical properties of the environment in which software is to work. In order to gain as much confidence as possible in the software for a critical system, the analyst should perform a 'Critical Event Analysis'.

The Critical Event Analysis procedure includes the following steps:

- Identify Critical Events
- For each critical event, identify all possible ways in which it may fail
- Capture these possible failure modes using the undesired event notation

- Study each undesired related state to find out how to achieve protection against such possible failure

The following diagram (Figure 9) illustrates the critical event analysis in BPA as described in step XI:

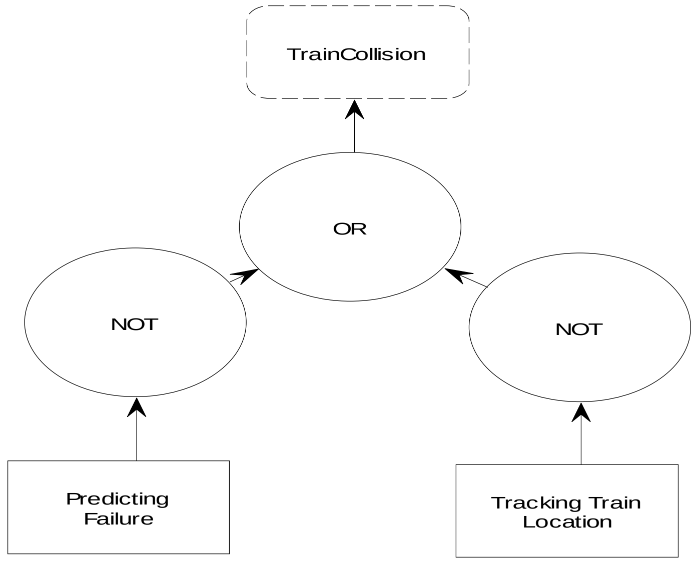


Figure 9 Critical Analysis Diagram – Train Traffic Management

In the Critical Analysis Diagram, the round ended rectangles represent the states of the critical events. The dashed rounded ended rectangles represent the failure that occurs due to these states.

5. MISSING REQUIREMENTS

There were no missing requirements that required generating a Derived Requirement Document.

XII. TTMS Class Diagram

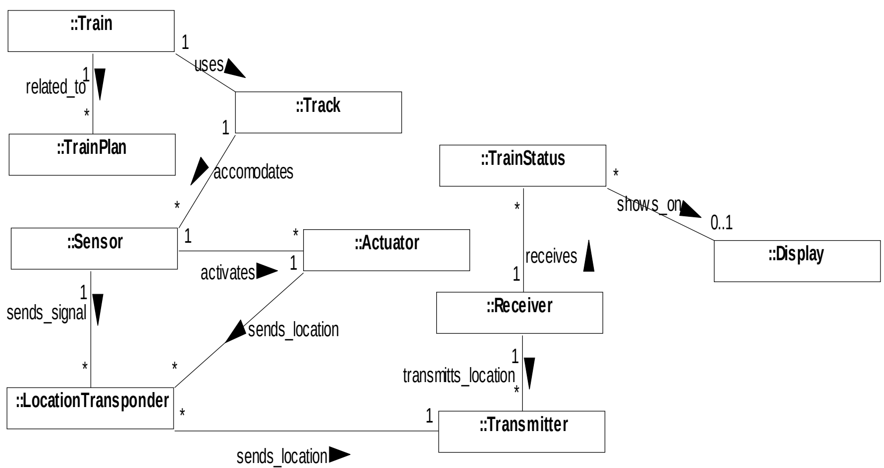


Figure 10 Class Diagram - TTMS

6. EVALUATION OF THE EFFECTIVENESS OF THE BPA MODELING METHODOLOGY AND THE UCA MODELING METHODOLOGY

In this research, three real-life applications were used to illustrate the effectiveness of the new BPA modeling methodology in handling safety-critical real-time systems development:

(1) The Therac-25 Medical Device System [LEVESON 96]

(2) The Production Cell System [LEWERENTZ 95]
(3) The Railroad Crossing System [HEITMEYER 96].

The UCA and the BPA modeling methodologies were used to define the requirements and model these systems. The first application was used, as a proof of concept, in a pilot case study. The last two applications were distributed as part of the case studies material to compare the UCA versus the BPA modeling methodologies using the pre-mentioned effectiveness criteria.

7. THE EFFECTIVENESS METRICS

The effectiveness metrics categories used in this research include:

1. System Effectiveness represented by safety
2. Requirements Engineering Process Effectiveness represented by the CMM and CMMI repeatability
3. Definition of Requirements Effectiveness represented by the ANSI (NIST) / IEEE standards for systems specifications.

7.1 System Effectiveness

Because the focus of this research is on safety-critical systems, Safety is selected as the measure of system effectiveness. 'Safety' is defined in the American Heritage dictionary as the condition of being safe, or as the freedom from danger, risk, or injury.

7.2 Requirements Engineering Process Effectiveness

'Repeatability' is defined in the American Heritage dictionary as following the same procedure in doing or expressing something. Repeatability is an indicator of the effectiveness of software development process. It represents the second level of the SEI Capability Maturity Model (CMM) [HUMPHREY 89]. The Capability Maturity Model (CMM and the new CMMI) is a framework for assessing software process maturity in software development organizations.

7.3 Definition of Requirements Effectiveness

The following is a compiled list, from the ANSI/IEEE Std 830-1984 [IEEE 84], of the particular defined characteristics that were used to compare the effectiveness of the BPA modeling methodology versus the Use Case Analysis modeling methodology:

- Unambiguous
  Requirements definition is unambiguous if – and only if – every requirement stated therein has only one interpretation. To reduce the ambiguity inherent in natural languages, formal requirement specification languages and/or graphical modeling techniques are used to define the requirements.
- Complete
  Requirements definition is complete if it possesses the following qualities:
  - Inclusion of all significant requirements and constraints
  - Specification of responses to valid and invalid input values.
- Consistent
  Requirements Definition is inconsistent if and only if no set of individual requirements described in it conflict. Likely conflict types are:
  - Naming
  - Characteristics specification
  - Temporal or Logical.
- Modifiable
  Requirements Definition is modifiable if its structure and style enable changes to be made easily, completely, and consistently. Modifiability generally requires:
  - Ease of use
  - No redundancy.
- Traceable
  Requirements Definition is traceable if the origin of each requirement is clear and if it facilitates the referencing of each requirement in the next development stages. The forward traceability is especially important when requirements change. It is essential to be able to identify all the requirements that may be affected by these modifications. Unique names or reference numbers are required for this purpose.
- Usability
  Usability means that Requirements Definition is usable during design, implementation, and maintenance phases.

However, there is no evidence that Usability is an independent characteristic. It can mean Unambiguous, Complete, or Modifiable. Because we have included all of the pre-mentioned characteristics, Usability was taken out.

8. THE CASE STUDIES

A 'Case Study' is defined in the American Heritage Dictionary as an exemplary or cautionary model. Case studies continue to be used extensively in the evaluation research [YIN 94]. The case study method is illustrated in the following figure (modified version of Yin [YIN 94]):

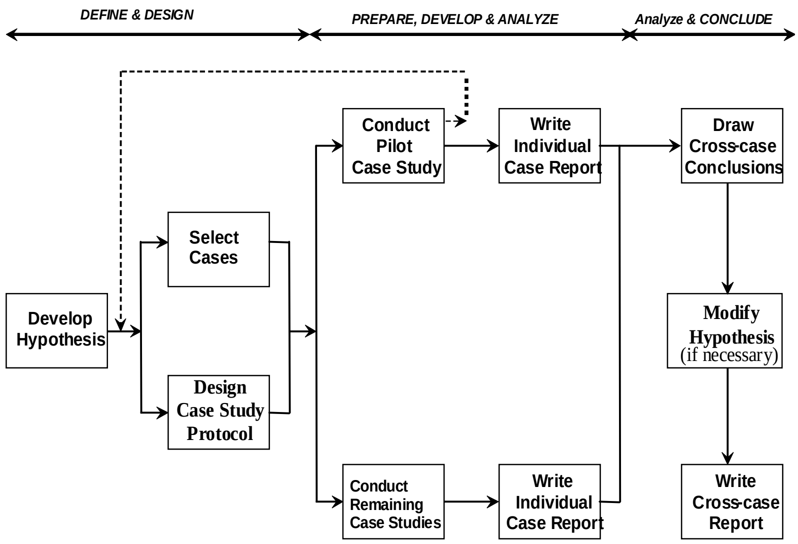


Figure 11 Case Study Research Method

In this research, three real-life applications were used to illustrate the effectiveness of the new BPA modeling methodology in handling safety-critical real-time systems development:
(1) The Therac-25 Medical Device System [LEVESON 96]
(2) The Production Cell System [LEWERENTZ 95]
(3) The Railroad Crossing System [HEITMEYER 96].
The UCA and the BPA modeling methodologies were used to define the requirements and model these systems. The first application was used, as a proof of concept, in a pilot case study. The last two applications were distributed as part of the case studies material to compare the UCA versus the BPA modeling methodologies using the pre-mentioned effectiveness criteria.

9. THE PAIRWISE COMPARISON METHOD

A Multi-Criteria Decision Making (MCDM) Tool, named as DECISION, was developed by this researcher to evaluate the assessment results. The Decision tool uses a combination of the Analytic Hierarchy Process (AHP) and the ELECTRE Pairwise Comparison approaches. Pairwise Comparisons is the process in which experts rate a set of objects, events, or criteria, by comparing only two at a time. Most people are reliable estimators using pairwise comparisons because they only have to consider two things at a time [SAATY 82]. The selected approaches, AHP and ELECTRE, are popular and have strong theoretical basis [MEYER 91], [BUI 87].

10. CASE STUDY PROTOCOL DESIGN

The protocol design stage is composed of two main tasks:

a. Determine the required skills of the subject matter experts (SMEs)
   - The required skills to evaluate the UCA vs. the BPA modeling methodology were determined as follows:
     - The level of Structured Analysis Experience
     - The level of UML / UCA Experience
   - In order to determine the level of experience, a questionnaire was sent to the subject matter experts. The details are presented in the Subject Matter Expert subsection.

b. Develop and review the protocol
   - A good guideline for doing case studies is to conduct the research so that another researcher (or an auditor) could repeat the procedures and arrive at the same results. This may be achieved by thorough documentation of the procedures to be followed and the questions to be asked. The following subsection presents the used case study material in which the documentation guidelines of the case study protocol were followed.

International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

11. THE CASE STUDY MATERIAL

Each SME was provided with a case study kit that contains:

1. A cover letter
2. A consent form
3. The instructions
4. An application
5. A brief overview and a step by step procedure describing how to analyze and model requirements using the UCA and BPA modeling methodologies
6. Two analyses of the given application; one using the UCA modeling modeling methodology and the other using the BPA modeling modeling methodology
7. Explanation of the evaluation method (Pairwise Comparison) and the effectiveness criteria

The set of questions presented clearly in a table format (Evaluation Forms).

12. SUBJECT MATTER EXPERTS' SELECTION

The SMEs were selected from two sets of software engineering professionals. The first set was composed of software engineering professionals who happen to be graduate students in the Information Technology (INFT) School and attended one or more software engineering courses. The second set was composed of working professionals at Lockheed Martin Company and Federal Government. Email and surface mail letters were sent to more than 100 of these graduate students and working software engineering professionals. Sixteen software engineering professionals, with the required experience to carry out the case studies, were selected out of these two sets of professionals that showed interest in participating. Questionnaires were sent to these SMEs to classify them according to their Structured Analysis (SA) methods and UML / UCA knowledge. The questions used in the SMEs selection are shown in the next section. As shown in Figure 12, from each set, two with the same kind of experience out of each group were selected to receive one of the two case studies' applications.

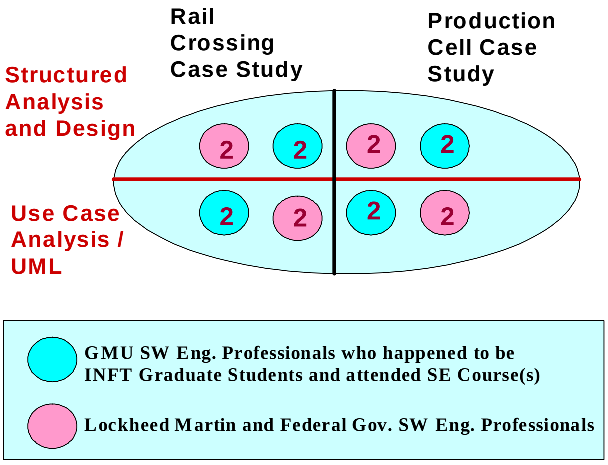


Figure 12 Subject Matter Experts (SMEs) Selection

It is worth mentioning that more than 60% of the SMEs were persons previously unfamiliar with the researcher conducting the assessment.

13. THE SUBJECT MATTER EXPERTS

The number of SMEs depends on the number of the controlled variables. The controlled variables are:

- The applications.
- The set of the SMEs.
- The SMEs' software engineering experience:
  - Structured Analysis and traditional software engineering modeling techniques
  - Use Case Analysis / UML.

If two subjects are used for each control variable, the total required number of subjects would be 16 (2 X 2 X 2 X 2). Table 1 illustrates this break down by SME experience and application:

Table 1. SME Experience to Application Assignment Matrix

Experience/Application	Production Cell Case Study	Rail Crossing Case Study
Structured Analysis and Design	4	4
Use Case Analysis / UML	4	4

International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

14. CASE STUDIES' RESULTS

14.1 Case Studies Results

14.1.1 AHP Results

The summary of the assessment results using AHP is illustrated in Figure 13 in a column chart format.

	BPA	UCA
SME 5C1	0.89	0.11
SME 4C1	0.75	0.25
SME 1C1	0.57	0.43
SME 8C2	0.75	0.25
SME 7C2	0.66	0.34
SME 3C2	0.8	0.2
SME 2C2	0.57	0.43
SME 6C1	0.47	0.53
SME9C1	0.73	0.27
SME10C1	0.79	0.21
SME11C1	0.76	0.24
SME12C1	0.85	0.15
SME13C2	0.77	0.23
SME 14C2	0.84	0.16
SME 15C2	0.81	0.19
SME 16C2	0.88	0.12

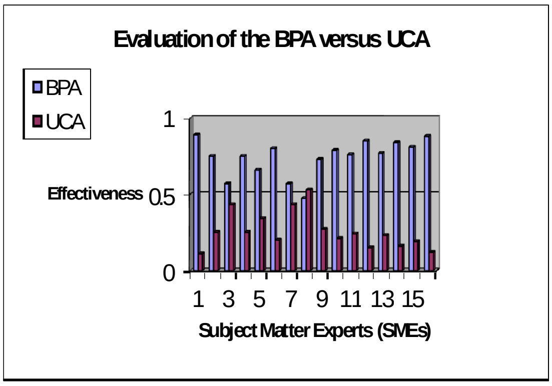


Figure 13 Effectiveness Evaluation of BPA versus UCA – Results using AHP

The above results show that:

- Fifteen SMEs out of sixteen evaluated BPA as a more effective alternative to UCA in defining the requirements. This result gives an indication of about 93.8 % approval rate for the thesis hypothesis.
- One SMEs out of sixteen evaluated UCA as more effective alternative to BPA in defining the requirements.
- The average of the overall priorities for BPA (total of the overall priorities of the SMEs divided by the total number of SMEs, which is 11.89/16) is about 0.74.
- The average of the overall priorities for UCA (total of the overall priorities of the SMEs divided by the total number of SMEs, which is 4.11/16) is about 0.26.

The above results give an indication of about 93.8 % approval rate for the thesis hypothesis with about three times overall effectiveness for BPA over UCA on the average.

14.1.2 ELECTRE Results

The following is a collective summary by number of SMEs using ELECTRE:

- Fourteen SMEs out of sixteen evaluated BPA as a more effective alternative to UCA in defining the requirements.
- Two SMEs out of sixteen evaluated UCA as more effective alternative to BPA in defining the requirements.

The above results give an indication of about 87% approval rate for the thesis hypothesis. Figure 14 is a pie chart representation of the results' summary by the number of SMEs. In summary, the ELECTRE method gave a less approval rate (87%) than the AHP method (93.8). To be on the conservative side, we can safely conclude that the approval rate of the hypothesis is 87%.

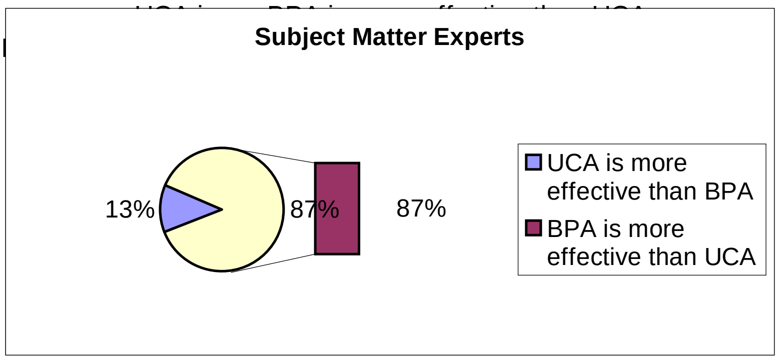


Figure 14 Evaluation Results' Summary by Number of Subject Matter Experts (SMEs)

15. RESEARCH CONTRIBUTION

The major contributions of this research are:

- The Behavioral Pattern Analysis (BPA) modeling methodology.

- Validation of the hypothesis that the Behavioral Pattern Analysis (BPA) modeling methodology is a more effective alternative to Use Case Analysis (UCA) in modeling the functional requirements of Human-Machine Safety-Critical Real-time Systems.
- Another contribution of this research was the development of an interactive software tool (DECISION) that is based on a combination of the Analytic Hierarchy Process (AHP) and the ELECTRE Multi-Criteria Decision Making (MCDM) methods. The DECISION software tool was used to process the assessment results of the case studies.

16. WHY THIS WORK IS IMPORTANT

16.1 Real-time Systems

In most of the popular object-oriented development modeling methodologies state diagrams are used to model the behavior. By using state diagrams, one is focusing on an individual object's response to specific events rather than objects interaction. Hence, objects interaction must be reconstructed from the analysis of groups of diagrams. Such a task is at least complex and error-prone. By describing the requirements in terms of events, represented by the behavioral patterns, this perceived problem is reduced.

16.2 Multi-agent Systems

There is a need for a multi-agent systems analysis and design method that is powerful enough to model interaction patterns involving autonomous agents as well as the knowledge structure that they need to manipulate. In a future research, the author is planning to show that the BPA modeling methodology can be used to model multi-agent systems effectively.

16.3 Safety-critical Systems

In these systems, analysts should perform a 'Safety Analysis'. Using BPA, one identifies and documents the critical events during the requirements definition stage. GOD says [KORAN][TORAH], " … Whoever rescues a single life earns as much merit as though he had rescued the entire world." If the use of the BPA Modeling methodology may save one life, the significance of this modeling methodology is immeasurable.

Bibliography

[ALLEN 83] Allen, J. F., Maintaining Knowledge about Temporal Intervals, *Communications of ACM*, 26, 1983, pp 832-843

[BELL 76] Bell, T. and Thayer, T., Software Requirements: are they really a problem?, Second International Conference on Software Engineering, 1976.

[BOOCH 99] Booch, G.,Jacobson, I., and Rumbaugh, J., The Unified Modeling Language User Guide, Addison Wesley, Reading, Massachusetts, 1999.

[BUI 87] Bui, Tung X., Co-oP, A Group Decision Support System for Cooperative Multiple Criteria Group Decision Making, Springer-Verlag, 1987.

[DAVIDSON 80] Davidson, D., Essays on Actions and Events, Oxford University Press, New York, 1980.

[ELANSARY 02] El-Ansary, Assem I., Behavioral Pattern Analysis: Towards a New Representation of Systems Requirements Based on Actions and Events, in *Proceedings of the 2002 ACM Symposium on Applied Computing*, ACM, New York, NY, 2002.

[ELANSARY 05] El-Ansary, Assem I., Behavioral Pattern Analysis: Towards a New Representation of Systems Requirements Based on Actions and Events, Doctoral Thesis, George Mason University, 2005.

[GOLDSMITH 95] Sylvia Goldsmith, A Practical Guide to Real-Time Systems Development, Prentice Hall International, London, UK, 1995.

[GRAHAM 95] Graham, Ian, Migrating to Object Technology, Addison-Wesley, Reading, Massachusetts, 1995.

[HEITMEYER 96] Heitmeyer, Constance and Mandrioli, Dino, Formal Methods for Real-Time Computing: An Overview, in Formal Methods for Real-Time Computing, John Wiley & Sons, Inc., NY, 1996.

[IEEE 84] ANSI and IEEE, ANSI/IEEE Std 830-1984, IEEE Guide to Software Requirements Specification, in System and Software Requirements Engineering, IEEE Computer Society Press, Los Alamitos, California, 1990, pp 170-192.

[JACKSON 95] Jackson, Michael, Software Requirements & Specification, A Lexicon of Practice, Principles and Prejudices, ACM Press, Addison-Wesley, Reading, Massachusetts, 1995.

[JACOBSON 92] Jacobson, I., Christeron, M., and Overgaard, Object-Oriented Software Engineering: A Use Case Driven Approach, Addison-Wesley, Reading, Massachusetts, 1992.

[JACOBSON 95] Jacobson, I., The Use Case Construct in Object-Oriented Software Engineering, in Carroll, J., Scenario-Based Design, John Wiley & Sons, Inc., NY, 1995.

[LEWERENTZ 95] Lewerentz, Claus, and Lindner, Thomas, Formal Development of Reactive Systems, Springer-Verlag, NY, 1995.

[OOPSLA 98] Fowler, Martin and Cockburn, Question Time! About Use Cases, OOPSLA'98 Proceedings, ACM Press, New York, NY, 1998.