The Dhammic Framework for Understanding the Cause of System Project Failures from Buddhist Insights

Pita Jarupunphol
Department of Informatics
Faculty of Science and Technology
Phuket Rajabhat University
Phuket, Thailand, 83000
Email: p.jarupunphol@pkru.ac.th

Clark Thomborson
Department of Computer Science
University of Auckland
Auckland, New Zealand, 1010
Email: c.thomborson@auckland.ac.nz

Abstract—**Cognitive Informatics (CI) is a transdisciplinary approach to the cognitive and information sciences, emphasising the informational aspects of cognitive processes, with applications in the engineering of complex systems. Human cognition is a transcultural phenomeon, however to date all contributions to CI have been based on Western philosophy and science. In this article, we indicate how some of the fundamental concepts in Buddhist epistemology may be modeled in the CI framework. In particular: we develop a logical specification, in the Z notation, of cognitive processes which occur at levels 1 through 4 of the Layered Reference Model of the Brain (LRMB). We call these processes the Dhammic Framework. As with any axiomatic system, the validity of the Dhammic Framework cannot be proved by experimentation; but it could be invalidated if any of its implications were either logically inconsistent or in disagreement with experimental observation. Our formal statement of the Dhammic Framework will allow its axioms to be tested, scientifically, for contradiction within the framework of cognitive informatics. To this end, we propose a testable hypothesis about a way to avoid failures in systems engineering.**

Keywords*—*Cognitive Informatics (CI), Dhammic Framework, Layered Reference Model of the Brain (LRMB), Theoretical Framework of CI

I. Introduction

According to Wang [1], CI is “the transdisciplinary enquiry of cognitive and information sciences that investigates into the internal information processing mechanisms and processes of the brain and natural intelligence, and their engineering applications via an interdisciplinary approach”. Bo Zhang, in a recent position statement at a major CI conference, argued that “CI should benefit from the multidisciplinary research among information science, cognitive science and brain science, etc.” [2, p.18].

The emergent CI framework is a logically-coherent synthesis of well-accepted theories from psychology, cognitive science, and psychology. Constructs in these theories represent how information is processed as objects, attributes, and relations [3] in a layered reference model of the brain (LRMB) [4] with thirty-seven cognitive processes at six levels: Sensation, Memory, Perception, Action, Meta-Cognition, and Higher Cognition. The cognitive processes are not fully specified, as yet, in the CI framework.

In this article, we formally specify some cognitive processes. We derive our specifications from written records of the theory of cognition and perception developed more than twenty-five hundred years ago by Siddhartha Gautama (Lord Buddha). This theory posits some very specific connections between sensation, memory, perception, action, and meta-cognition. Hundreds of millions of practicing Buddhists have, subsequently, found useful guidance and no contradiction in these teachings – so the primary axioms in these teachings are demonstrably valid, at least for all practicing Buddhists, and possibly for all naturally intelligent individuals.

The *Paticca Samuppada*, one of the central tenets in Buddhism, is widely used to explain that all things arise in dependence upon multiple causes and conditions, in particular, condition genesis, dependent origination, dependent arising, dependent co-arising, and interdependent arising. A number of conditions in this doctrine have also been used to describe how any individual’s mind reacts to any perceived system that is always changing, according to Buddhist phenomenology. Anyone who attempts to apply this doctrine to understand, dispassionately, their own mental processes is attempting to engage in what is called ‘mindfulness meditation’: they are recognising their own thoughts and actions, as soon they occur, without judging them. This is a higher cognitive process, occurring at level 6 in the LRMB. We have not yet extended our Dhammic Framework to include a specification of mindfulness meditation. Instead, our current Dhammic Framework specifies only the processes occurring at levels 1 through 5 in the LRMB, as an individual perceives a changing system. Our goal is to develop a model – congruent with Buddhist philosophy and testable scientifically – of the cognitive process called “attachment” in Buddhism. Our working hypothesis is that a condition of attachment generally leads to undesirable outcomes in systems engineering. This article describes a theory which will guide future experimental studies which are aimed at validating it.

In order to avoid ambiguity, we express our Dhammic Framework as schemata in the Z notation [5], [6], [7]. Z is a formal specification language for Zermelo–Fraenkel set theory. These schemata express important aspects of the Dhammic theory of perception, from initial contact to more complex cognitions and emotions. Our Dhammic framework could be used, in a future cognitive computing system, to model how the perceptions of a collection of human users evolve as they gain experience with a collection of systems – where a "system" is defined to be any aggregation of matter, information and energy that some human perceives to be a persistent entity that is distinguishable from other systems.

II. The Dhammic Framework

In Buddhism, mind is everything. All problems are rooted in perception to a phenomenon. A system has no meaning if it is not perceived. According to Buddha [8, p.117], "the beginning lies in the recognition that the external world is only a manifestation of the activities of the mind itself, and that the mind grasps it as an external world simply because of its habit of discrimination and false-reasoning…the world has no self-nature, that it is unborn, that it is like a passing cloud, like the moon reflected in the ocean, like a vision, a mirage, a dream". *Citta* (heart, mind) is mentioned in various Buddhist doctrines. In mindfulness meditation, the meaning of citta can be very specific. In our framework, citta simply means a perceptual and cognitive element that represents an individual's mindset, or state of mind. In this case, the Dhammic framework will be used to represent how a system is perceived and cognised by a user. The Dhammic framework is based on two major schemata: 1) state schemata (the state schema provides a static view of a state of mind that includes components (or state variables) of the condition) and 2) operation schemata (the operation schema specifies how the state schema can be changed by taking an instance of the state schema and producing a new instance).

A. Sanna

Sanna (cognition) is a condition in mind that is related to short-term and long-term memory of knowledge and experiences regarding a perceived phenomenon. In this case, the state schema *SANNA* will provide a static view of all psychological conditions, including *phassa*, *rupa*, *nama*, *vedana*, *tanha*, *upadana*, *cetana*, *kamma*. The *SANNA* state schema is used to declare psychological conditions. The upper part of the schema consists of variable declarations of the mind. *SYSTEM* and *USER* are two basic type definitions where *system* is a variable of $\mathbb{P}\, SYSTEM$ and *user* is a variable of $\mathbb{P}\, USER$. Please note that 'self' in the Dhammic context is not specifically classified into subject positions, but is generally reserved for anyone, who is capable of perceiving a phenomenon. In the meantime, a perceived object is not classified into different names, but is generally regarded as a perceived phenomenon. As such, a user in this context is applicable to anyone capable of perceiving a system. A system in this context is a perceived object that can be any aggregation of matter, information, and energy.

Since the framework is based on a user's perception to a system at a particular state, we use *perception* as a partial function from *USER* to *SYSTEM* to represent a relationship that maps each element of *USER* to, at most, one element of *SYSTEM*. Other functions of psychological conditions are also defined in the same manner as the partial function *perception*. The lower part of the schema consists of logical statements which define the perception. The set *user* is to be a subset equal to the domain of the *perception* function and the set *system* is to be a subset equal to the range of the *perception* function. All perceptual functions are defined as subsets equal to the partial function *perception*.

$$
\begin{array}{|l}
\multicolumn{1}{l}{SANNA} \\
\hline
user : \mathbb{P}\, USER \\
system : \mathbb{P}\, SYSTEM \\
perception : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
phassa : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
rupa : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
nama : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
vedana : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
tanha : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
upadana : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
cetana : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
kamma : USER \mathrel{+\!\!\!\!\rightarrow} SYSTEM \\
\hline
user \subseteq \mathrm{dom}\, perception \\
system \subseteq \mathrm{ran}\, perception \\
phassa \subseteq perception \\
rupa \subseteq perception \\
nama \subseteq perception \\
vedana \subseteq perception \\
tanha \subseteq perception \\
upadana \subseteq perception \\
cetana \subseteq perception \\
kamma \subseteq perception \\
\hline
\end{array}
$$

The initialisation schema *INIT* is created to set an initial stage before a user and a system are contacted. Two variables, including *user* and *system* are set at $\emptyset$ in the first stage when a user has not yet perceived a system. There is no user and no system in the first state. In other words, there is no cognition about the system at the beginning.

$$
\begin{array}{|l}
\multicolumn{1}{l}{INIT} \\
\hline
\Delta SANNA \\
\hline
user = \emptyset \\
system = \emptyset \\
\hline
\end{array}
$$

B. Phassa

Phassa (contact) refers to an initial state of mind when a human and a phenomenon make contact. The *PHASSA* is an operation schema that includes the state schema $\Delta SANNA$. This means the schema *SANNA* will be used with both its declarations and predicates. The operation schema *PHASSA* updates the component *phassa* in the schema *SANNA*. The upper part of the schema consists of variable declarations in which $u?$ (defined as *USER* type) and $s?$ (defined as *SYSTEM* type) are two emerging elements (or input arguments) in *user* and *system* respectively. The lower part of the schema, *PHASSA*, consists of logical statements of phassa. The $u? \notin \mathrm{dom}\, phassa$ predicate is a pre-condition; it defines conditions which must hold when the operation starts. The pre-condition is used to represent the inconstant state of a phenomenon that a new system state is always perceived.

The predicate $phassa' = phassa \cup u? \mapsto s?$ is post-condition of phassa, used to represent an element $u?$ has been updated in *user* ($user' = user \cup u?$) and an element $s?$ has been updated in *system* ($system' = system \cup s?$) by *phassa*. Please note that any similar predicate to what we have explained in this section will not be explained further for other states of mind.

$$
\begin{array}{|l}
\hline
\textit{PHASSA} \\
\Delta \textit{SANNA} \\
u? : \textit{USER} \\
s? : \textit{SYSTEM} \\
\hline
u? \notin \operatorname{dom} \textit{phassa} \\
\textit{phassa}' = \textit{phassa} \cup \{u? \mapsto s?\} \\
\hline
\end{array}
$$

C. Rupa

Rupa (form) is a condition when external properties of a phenomenon are perceived by a human. It refers to a state of mind to the form of a perceived phenomenon, such as its physical properties that are based on the combination of four elements, including solidity, fluidity, motion and heat (understood by Buddhism as soil, water, wind, and fire). If there is no contact between a human and a phenomenon, no meaning can be perceived. *Rupa* can be described by primary qualities in Western philosophy. These are physical properties, such as those noted above, and logical construction of physical properties which are inseparable from all matter [9]. *Rupa* is an observable property of a system that directly interacts with a user.

The *RUPA* is an operation schema that includes the state schema $\Delta SANNA$. The operation schema *RUPA* updates the component *rupa* in the schema *SANNA*. The predicate $u? \mapsto s? \in phassa$ in the lower part is a pre-condition; it defines conditions, that a system must be contacted by a user when external properties of a system are perceived. For example, a user can perceive external properties of a system only after the user and the system are contacted($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \in rupa \Rightarrow u? \mapsto s? \in phassa$). In the meantime, if the user and the system are not contacted, then the user cannot perceive external properties of a system($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin phassa \Rightarrow u? \mapsto s? \notin rupa$)

$$
\begin{array}{|l}
\hline
\textit{RUPA} \\
\Delta \textit{SANNA} \\
u? : \textit{USER} \\
s? : \textit{SYSTEM} \\
\hline
u? \mapsto s? \in \textit{phassa} \\
u? \notin \operatorname{dom} \textit{rupa} \\
\textit{rupa}' = \textit{rupa} \cup \{u? \mapsto s?\} \\
\hline
\end{array}
$$

D. Nama

Nama (formlessness, name) is a condition when qualities of a phenomenon are perceived by a human that arise after perceived external properties. If a phenomenon is not perceived, there is no meaning that can be perceived from it. *Nama* can be described by secondary qualities in Western philosophy that are dispositions to produce sensory experiences under appropriate conditions. The secondary qualities are intrinsic features of objects that are represented in perception based on sensory aspects such as smell, taste, sound, colour, and warmth or cold. Any plausible qualities that can be perceived directly from external properties of a system are *nama*. *Nama* is comparable to qualities that are directly influenced by external properties of a system. Perceived qualities are not stand-alone, but dependent upon perceived external properties. A number of perceived qualities have been defined by system engineers in different factors (e.g., usability, security, reliability, etc.).

The *NAMA* is an operational schema that includes the state schema $\Delta SANNA$. The operation schema *NAMA* updates the component *nama* in the schema *SANNA*. The predicate $u? \mapsto s? \in rupa$ in the lower part is a pre-condition; it defines conditions, that external properties of a system are perceived when the qualities of a system are perceived. For example, a user can perceive qualities from a system only after the user perceives its external properties ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \in nama \Rightarrow u? \mapsto s? \in rupa$). In the meantime, if a user does not perceive the external properties of a new system state, then they cannot perceive its qualities ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin rupa \Rightarrow u? \mapsto s? \notin nama$).

$$
\begin{array}{|l}
\hline
\textit{NAMA} \\
\Delta \textit{SANNA} \\
u? : \textit{USER} \\
s? : \textit{SYSTEM} \\
\hline
u? \mapsto s? \in \textit{rupa} \\
u? \notin \operatorname{dom} \textit{nama} \\
\textit{nama}' = \textit{nama} \cup \{u? \mapsto s?\} \\
\hline
\end{array}
$$

E. Vedana

Vedana (feeling, sensation) is a condition when attitudes toward a phenomenon that immediately respond to perceived qualities of a phenomenon. If a quality of a phenomenon is not perceived, there is no attitude to a phenomenon. *Vedana* is just an initial level of attitude that can immediately noticed by a user. It refers to three plausible attitudes to a quality perceived from a phenomenon, including *sukha* (positive feelings such as like, satisfactoriness, pleasantness), *dukkha* (negative feelings such as dislike, unsatisfactoriness, unpleasantness), and *uppekha* (neutral, neither sukha nor dukkha) [10], [11]. All these attitudes will not arise without being influenced by perceived qualities.

In this case, *VEDANA* is a state schema that includes the state schema $\Delta SANNA$. As *sukha*, *dukkha*, and *uppekha* are types of *vedana*, their functions are also defined in the same manner as the partial function *vedana* in the schema *SANNA*. The lower part indicates that all three types of vedana are subsets equal to *vedana*. A user can feel positive, negative, or neutral to a system at a time of each perception ($\forall u? : user \mid \exists s? : system \bullet u? \mapsto s? \in sukha \vee u? \mapsto s? \in dukkha \vee u? \mapsto s? \in uppekha \bullet sukha \cap dukkha \cap uppekha = \emptyset$).

$$
\begin{array}{|l}
\hline
\textit{VEDANA} \\
\Delta \textit{SANNA} \\
\textit{sukha} : \textit{USER} \nrightarrow \textit{SYSTEM} \\
\textit{dukkha} : \textit{USER} \nrightarrow \textit{SYSTEM} \\
\textit{uppekha} : \textit{USER} \nrightarrow \textit{SYSTEM} \\
\hline
\textit{sukha} \subseteq \textit{vedana} \\
\textit{dukkha} \subseteq \textit{vedana} \\
\textit{uppekha} \subseteq \textit{vedana} \\
\hline
\end{array}
$$

1) Sukha: *Sukha* is a condition when attitude toward a system is positive. The *SUKHA* is an operational schema that includes the state schema $\Delta VEDANA$. The operation schema *SUKHA* updates the component *sukha* in the schema *VEDANA*. The predicate $u? \mapsto s? \in nama$ in the lower part is a pre-condition; it defines conditions that a user must have perceived in a system quality when the *sukha* operation arises in state of mind. For example, a user's positive attitude toward a system arises only after the user perceives qualities of a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \in sukha \Rightarrow u? \mapsto s? \in nama$). In the meantime, if the user's positive attitude toward a system does not arise, then the user does not perceive qualities of a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin nama \Rightarrow u? \mapsto s? \notin sukha$).

$$
\begin{array}{|l}
\multicolumn{1}{l}{SUKHA} \\
\hline
\Delta VEDANA \\
u? : USER \\
s? : SYSTEM \\
\hline
u? \mapsto s? \in nama \\
u? \notin \mathrm{dom}\, sukha \\
u? \notin \mathrm{dom}\, dukkha \\
u? \notin \mathrm{dom}\, uppekha \\
sukha' = sukha \cup \{u? \mapsto s?\} \\
\hline
\end{array}
$$

2) Dukkha: *Dukkha* is a condition when attitude toward a system is negative. *DUKKHA* is an operational schema that includes the state schema $\Delta VEDANA$. The operation schema *DUKKHA* updates the component *dukkha* in the schema *VEDANA*. The predicate $u? \mapsto s? \in nama$ in the lower part is a pre-condition; it defines conditions that a user must have perceived in a system quality when the *dukkha* operation arises in state of mind. For example, a user's negative attitude toward a system arises only after the user perceives qualities of a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u \mapsto s? \in dukkha \Rightarrow u \mapsto s? \in nama$). In the meantime, if the user's negative attitude toward a system does not arise, then the user does not perceive qualities of a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin nama \Rightarrow u? \mapsto s? \notin dukkha$).

$$
\begin{array}{|l}
\multicolumn{1}{l}{DUKKHA} \\
\hline
\Delta VEDANA \\
u? : USER \\
s? : SYSTEM \\
\hline
u? \mapsto s? \in nama \\
u? \notin \mathrm{dom}\, sukha \\
u? \notin \mathrm{dom}\, dukkha \\
u? \notin \mathrm{dom}\, uppekha \\
dukkha' = dukkha \cup \{u? \mapsto s?\} \\
\hline
\end{array}
$$

3) Uppekha: *Uppekha* is a condition of *vedana* when attitude toward a system is neutral that cannot be expressed as positive or negative. *UPPEKHA* is an operational schema that includes the state schema $\Delta VEDANA$. The operation schema *UPPEKHA* updates the component *uppekha* in the schema *VEDANA*. The predicate $u? \mapsto s? \in nama$ in the lower part is a pre-condition; it defines conditions that a user must have perceived in a system quality when the *uppekha* operation arises in state of mind. For example, a user's neutral attitude toward a system arises only after the user perceives qualities of a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \in uppekha \Rightarrow u? \mapsto s? \in nama$). In the meantime, if the user's neutral attitude toward a system does not arise, then the user does not perceive qualities of a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin nama \Rightarrow u? \mapsto s? \notin uppekha$).

$$
\begin{array}{|l}
\multicolumn{1}{l}{UPPEKHA} \\
\hline
\Delta VEDANA \\
u? : USER \\
s? : SYSTEM \\
\hline
u? \mapsto s? \in nama \\
u? \notin \mathrm{dom}\, sukha \\
u? \notin \mathrm{dom}\, dukkha \\
u? \notin \mathrm{dom}\, uppekha \\
uppekha' = uppekha \cup \{u? \mapsto s?\} \\
\hline
\end{array}
$$

F. Tanha

Tanha (craving, desire) is a condition of intensified degree of attitudes toward a phenomenon that arise after attitudes toward a phenomenon. If there is no attitude toward a phenomenon, an intensified degree of attitude to a phenomenon will not arise. *Tanha* is categorised into *kama-tanha* (desire or craving for sensuality), *vibhava-tanha* (aversion or craving for non-being), and *bhava-tanha* (craving for neutral or for not to decline).

When attitude toward a system is positive, a system is 'craved for sensuality'. When a system is negative, a system is 'craved for non-being'. In this case, *uppekha* is the only type of *vedana* that will not trigger *tanha* conditions. A user's intensified positive or negative attitudes toward a system will not arise if a user feel neutral to a system ($\forall u? \mapsto s? \in vedana \bullet u? \mapsto s? \in uppekha \Rightarrow u? \mapsto s? \notin kamatanha \wedge u? \mapsto s? \notin vibhavatanha$). When a system is either craved for sensuality or craved for non-being of sensuality, it is always 'craved for not to decline'.

Since *bhava-tanha* is always associated with *kama-tanha* and *vibhava-tanha*, it will not be formalised in this section. The *TANHA* is a state schema that includes the state schema $\Delta SANNA$. As *kama-tanha* and *vibhava-tanha* are types of *tanha*, their functions are also defined in the same manner as the partial function *tanha* in the schema *SANNA*. The lower part indicates that all types of *tanha* are subsets equal to *tanha*.

$$
\begin{array}{|l}
\multicolumn{1}{l}{TANHA} \\
\hline
\Delta SANNA \\
kamatanha : USER \nrightarrow SYSTEM \\
vibhavatanha : USER \nrightarrow SYSTEM \\
\hline
kamatanha \subseteq tanha \\
vibhavatanha \subseteq tanha \\
\hline
\end{array}
$$

1) Kama-Tanha: *Kama-tanha* is a type of *tanha* arising after *sukha* of *vedana*. It is a condition of an intensified degree of positive attitude toward a system in accordance with positive attitude. *KAMATANHA* is an operational schema that includes the state schema $\Delta TANHA$. The operation schema *KAMATANHA* updates the component *kamatanha* in the schema *TANHA*. The predicate $u? \mapsto s? \in sukha$ in the lower part is a pre-condition; it defines the condition that a user attitude toward using a system must have been positive when the *kama-tanha* operation arises in state of mind. For example, a user's intensified positive attitude toward a system arises only after the user's positive attitude toward a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \in kamatanha \Rightarrow u? \mapsto s? \in sukha$). In the meantime, if the user's positive attitude toward a system does not arise, then the user's intensified positive attitude toward a system will not arise ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin sukha \Rightarrow u? \mapsto s? \notin kamatanha$).

$$\begin{array}{|l}
\textit{KAMATANHA} \\
\hline
\Delta \textit{TANHA} \\
u? : \textit{USER} \\
s? : \textit{SYSTEM} \\
\hline
u? \mapsto s? \in sukha \\
u? \notin \operatorname{dom} kamatanha \\
kamatanha' = kamatanha \cup \{u? \mapsto s?\} \\
\hline
\end{array}$$

2) Vibhava-Tanha: Vibhava-tanha is a condition of *tanha* arising after *dukkha*. It is a condition of an intensified degree of negative attitude toward a system in accordance with negative attitude. *VIBHAVATANHA* is an operational schema that includes the state schema $\Delta TANHA$. The operation schema *VIBHAVATANHA* updates the component *vibhavatanha* in the schema *TANHA*. The predicate $u? \mapsto s? \in dukkha$ in the lower part is a pre-condition; it defines the condition that a user attitude toward using a system must be negative when aversion toward a system arises. For example, a user's intensified negative attitude toward a system arises only after the user's negative attitude toward a system ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \in vibhavatanha \Rightarrow u? \mapsto s? \in dukkha$). In the meantime, if the user's negative attitude toward a system does not arise, then the user's intensified negative attitude toward a system will not arise ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin dukkha \Rightarrow u? \mapsto s? \notin vibhavatanha$).

$$\begin{array}{|l}
\textit{VIBHAVATANHA} \\
\hline
\Delta \textit{TANHA} \\
u? : \textit{USER} \\
s? : \textit{SYSTEM} \\
\hline
u? \mapsto s? \in dukkha \\
u? \notin \operatorname{dom} vibhavatanha \\
vibhavatanha' = vibhavatanha \cup \{u? \mapsto s?\} \\
\hline
\end{array}$$

G. Upadana

Upadana (attachment, clinging) is a condition when a system is attached and cognised by a human after intensified degree of attitudes toward a system. In other words, an attached system is driven by an intensified degree of attitude. A system can be attached for sensuality (or secure attachment in psychology) or for non-being (insecure attachment in psychology) [12], [13], [14]. A system, which is emotionally attached for security or insecurity will influence the user's view attachment to the system. If the user feels secure attachment to the system, the user's view to the system will be attached by the user as right. In contrast, the user's view to the system will be attached by the user as wrong if the user feels insecure attachment to the system.

Since a condition of attachment is preconditioned by different psychological conditions starting from perceived external properties, any change in external properties can result in influencing the entire chain of attached cognition. All sufferings are because underlying states of mind of an attached phenomenon are triggered when a perceived phenomenon is not the same as an attached phenomenon.

The *UPADANA* is an operational schema that includes the state schema $\Delta SANNA$. The operation schema *UPADANA* updates the component *upadana* in the schema *SANNA*. The predicate $u? \mapsto s? \in tanha$ in the lower part is a pre-condition; it defines conditions, that external properties toward a system are perceived when the qualities of a system are perceived. For example, a user's attachment toward a system arises only after the user's intensified attitude toward a system. A system is attached in accordance with craving for sensuality ($\forall u : user, s : system \mid \exists u?, s? \bullet u \mapsto s? \in upadana \Rightarrow u \mapsto s? \in kamatanha$) or craving for non-being ($\forall u : user, s : system \mid \exists u?, s? \bullet u? \mapsto s? \notin upadana \Rightarrow u? \mapsto s? \in vibhavatanha$). If the user's intensified positive attitude toward a system does not arise, then the user's attachment toward a system for sensuality will not arise ($\forall u : user, s : system \mid \exists u?, s? \bullet \neg u? \mapsto s? \notin kamatanha \Rightarrow \neg u? \mapsto s? \in upadana$). In the meantime, if the user's intensified negative attitude toward a system does not arise, then the user's attachment toward a system for non-being will not arise ($\forall u : user, s : system \mid \exists u?, s? \bullet \neg u? \mapsto s? \notin vibhavatanha \Rightarrow \neg u? \mapsto s? \notin upadana$).

We use $upadana' = upadana \cup \{u? \mapsto s?\}$ to represent a condition when a user attaches to a system and $upadana' = upadana \setminus \{u? \mapsto s?\}$ to represent a condition when a user attaches to a system.

$$\begin{array}{|l}
\textit{UPADANA} \\
\hline
\Delta \textit{SANNA} \\
u? : \textit{USER} \\
s? : \textit{SYSTEM} \\
\hline
u? \mapsto s? \in tanha \\
u? \notin \operatorname{dom} upadana \\
u? \mapsto s? \in kamatanha \bullet \\
upadana' = upadana \cup \{u? \mapsto s?\} \\
u? \mapsto s? \notin vibhavatanha \bullet \\
upadana' = upadana \setminus \{u? \mapsto s?\} \\
\hline
\end{array}$$

H. Cetana

Cetana (intention) is an intention toward using a system, which is situated in *bhava* (becoming). Cetana is influenced by a condition of attachment. It is a condition that forms a pattern in order to take action in accordance with a condition of attachment. The implication of *cetana* as bhava is also mentioned in cetana sutta [15] that "What one intends, what one arranges, and what one obsesses about...When that consciousness lands and grows, there is the production of renewed becoming in the future".

The*CETANA* is an operational schema that includes the state schema $\Delta SANNA$. The operation schema *CETANA* updates the component *cetana* in the schema *SANNA*. The predicate $u? \mapsto s? \in upadana \vee u? \mapsto s? \notin upadana$ in the lower part is a pre-condition; it defines conditions that a condition of attachment must have arisen when the *cetana* operation arises in state of mind. For example, a user's intention toward a system arises in response to the user's attachment toward a system, which can be adopting a system ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet u? \mapsto s? \in \mathit{cetana} \Rightarrow u? \mapsto s? \in \mathit{upadana}$) or rejecting a system ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet u? \mapsto s? \notin \mathit{cetana} \Rightarrow u? \mapsto s? \notin \mathit{upadana}$). If the user's attachment toward a system for sensuality does not arise, then the user's intention to adopt a system will not arise ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet \neg u? \mapsto s? \in \mathit{upadana} \Rightarrow \neg u? \mapsto s? \in \mathit{cetana}$). In the meantime, if the user's attachment toward a system for non-being does not arise, then the user's intention to reject a system will not arise ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet \neg u? \mapsto s? \notin \mathit{upadana} \Rightarrow \neg u? \mapsto s? \notin \mathit{cetana}$).

We use $\mathit{cetana}' = \mathit{cetana} \cup \{u? \mapsto s?\}$ to represent a condition when a user intends to adopt a system and $\mathit{cetana}' = \mathit{cetana} \setminus \{u? \mapsto s?\}$ to represent a condition when a user intends to reject a system.

$$\begin{array}{|l}
\mathit{CETANA} \\
\hline
\Delta \mathit{SANNA} \\
u? : \mathit{USER} \\
s? : \mathit{SYSTEM} \\
\hline
u? \mapsto s? \in \mathit{upadana} \lor u? \mapsto s? \notin \mathit{upadana} \\
u? \notin \operatorname{dom} \mathit{cetana} \\
u? \mapsto s? \in \mathit{upadana} \bullet \\
\mathit{cetana}' = \mathit{cetana} \cup \{u? \mapsto s?\} \\
u? \mapsto s? \notin \mathit{upadana} \bullet \\
\mathit{cetana}' = \mathit{cetana} \setminus \{u? \mapsto s?\} \\
\hline
\end{array}$$

I. Kamma

Kamma (action) is comparable to an action toward using a system situated in *jati* (birth, status of coming forth), which is pre-conditioned by cetana in bhava. *Cetana* is a condition that is strongly connected to *kamma* (action, deed), which is referred to intention and the result of intention [16]. It is a condition that what is intended comes forth. *Cetana* is an intention toward a system, which is situated in *bhava*; while kamma is an action toward a system, which is situated in *jati*. In other words, *kamma* is conditioned by *cetana*. At this stage, a system can be adopted or rejected by a user. A system will be adopted when a user intention to use a system is strong. In the meantime, a system will be rejected when a user intention to reject a system strong.

The *KAMMA* is an operational schema that includes the state schema $\Delta \mathit{SANNA}$. The operation schema *KAMMA* updates the component *kamma* in the schema *SANNA*. The predicate $u? \mapsto s? \in \mathit{cetana} \lor u? \mapsto s? \notin \mathit{cetana}$ in the lower part is a pre-condition; it defines conditions that a user must have intention to adopt or reject a system when the *kamma* operation arises. For example, a user's action toward a system arises in response to the user's intention toward a system, which can be adopting a system ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet u? \mapsto s? \in \mathit{cetana} \Rightarrow u? \mapsto s? \in \mathit{kamma}$) or rejecting a system ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet u? \mapsto s? \notin \mathit{cetana} \Rightarrow u? \mapsto s? \notin \mathit{kamma}$). If the user's intention to adopt a system does not arise, then the user's action to adopt a system will not arise ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet \neg u? \mapsto s? \in \mathit{cetana} \Rightarrow \neg u? \mapsto s? \in \mathit{kamma}$). In the meantime, if the user's intention to reject a system does not arise, then the user's action to reject a system will not arise ($\forall u : \mathit{user}, s : \mathit{system} \mid \exists u?, s? \bullet \neg u \mapsto s? \notin \mathit{cetana} \Rightarrow \neg u \mapsto s? \notin \mathit{kamma}$).

We use $\mathit{kamma}' = \mathit{kamma} \cup \{u? \mapsto s?\}$ to represent a condition when a system is adopted by a user and $\mathit{kamma}' = \mathit{kamma} \setminus \{u? \mapsto s?\}$ to represent a condition when a system is rejected by a user.

$$\begin{array}{|l}
\mathit{KAMMA} \\
\hline
\Delta \mathit{SANNA} \\
u? : \mathit{USER} \\
s? : \mathit{SYSTEM} \\
\hline
u? \mapsto s? \in \mathit{cetana} \lor u? \mapsto s? \notin \mathit{cetana} \\
u? \notin \operatorname{dom} \mathit{kamma} \\
u? \mapsto s? \in \mathit{cetana} \bullet \\
\mathit{kamma}' = \mathit{kamma} \cup \{u? \mapsto s?\} \\
u? \mapsto s? \notin \mathit{cetana} \bullet \\
\mathit{kamma}' = \mathit{kamma} \setminus \{u? \mapsto s?\} \\
\hline
\end{array}$$

III. Undesirable Symptoms of Upadana

Although there are different views in systems engineering, they share the same goal – that the system is finally adopted by the user after the user and the system are contacted ($\forall u \mapsto s \in \mathit{phassa} \mid \exists u?, s? \bullet u? \mapsto s? \in \mathit{kamma}$). At the same time, they also share the goal that the system will not be rejected by the user ($\forall u \mapsto s \in \mathit{phassa} \mid \neg \exists u?, s? \bullet u? \mapsto s? \notin \mathit{kamma}$). In most cases, systems engineers will design a system in such a way that the user will have positive attitude toward the system and adopt the system after the user and the system are contacted ($\forall u \mapsto s \in \mathit{phassa} \mid \exists u?, s? \bullet u? \mapsto s? \in \mathit{sukha} \Rightarrow u? \mapsto s? \in \mathit{kamma}$). At the same time, they will also design a system in such a way that the user will not have negative attitude toward the system and reject the system after the user and the system are contacted ($\forall u \mapsto s \in \mathit{phassa} \mid \neg \exists u?, s? \bullet u? \mapsto s? \in \mathit{dukkha} \Rightarrow u? \mapsto s? \notin \mathit{kamma}$).

When a system is perceived by a user, however, the system is immediately cognised in each psychological condition. When the system is attached by the user and the user perceives the system again (despite the same object), there are two system states, including a recognised system (s) and a perceived system ($s?$). The most desirable condition of attachment for systems engineers is when the perceived system is attached for sensuality ($u \mapsto s? \in \mathit{upadana}$) and the cognised system is attached for non-being ($u \mapsto s \notin \mathit{upadana}$). In constrast, the most undesirable condition of attachment is when the cognised system is attached for sensuality ($u \mapsto s \in \mathit{upadana}$) and the perceived system is attached for non-being ($u \mapsto s? \notin \mathit{upadana}$). When the perceived system is attached for non-being in order to sustain the cognised system. It is the stage when the perceived system state is automatically attached in the user's mind as an undesirable feature of the cognised system. The user's mind is attached that rejecting the perceived system will sustain the cognised system ($\forall u \mapsto s \in \mathit{upadana} \mid \exists s? : \mathit{system} \bullet u \mapsto s? \notin \mathit{kamma} \Rightarrow u \mapsto s \in \mathit{upadana}$).

When the user has reached these symptoms of attachment, it is also the most undesirable condition for all systems engineers. The systems engineer needs to spend considerable effort to make the user adopt the system. In many cases, it does not assure that the user will eventually adopt the system

$(\forall\, u \mapsto s? \notin \mathit{upadana} \mid \exists\, s? : \mathit{system} \bullet u \mapsto s \in \mathit{upadana} \Rightarrow u \mapsto s? \notin \mathit{kamma})$. A number of cyclic attempts at addressing users' persistent attachment is an undesirable condition in systems engineering. It is understood in Buddhism as *samsara* (continuous movement, endless suffering, cyclic existence, perpetual wandering, transmigration, wheel of suffering).

IV. Middle Path in Systems Engineering

We have learnt there are two extreme paths leading to desirable or undesirable outcomes in systems engineering. Either positive or negative attitude toward a system will lead to a condition of attachment. A system will be adopted or rejected depending upon a this condition. When an adopted system and a rejected system are 'attached' by a system user, this will also affects the system engineer, who can be deceived into *samsara* which leads to system project failures.

In contrast to extreme paths, Middle Path (or 'MP') has long been studied in different bodies of literature as a way of reaching a compromise between contradictory extreme views. For example, Doctrine of the Mean (or Golden Mean) of Aristotle [17], [18] was proposed for identifying the desirable middle between two extremes in a definition. Doctrine of the Mean was also mentioned in Neo-Confucianism of Chinese philosophy [19] as a way to maintain balance and harmony in extremes. All these concepts are based on belief the true virtue is situated in the space that is not extreme or far to one side.

In many applications of systems engineering, objective measurements are often used to illustrate MP concepts. For example, Kevin Burton, founder of Web Crawler, described in his personal blog[1] that MP was another way of engineering an optimal system. He cited the words of Dalai Lama. Managing SWAP files in Linux was used to illustrate that MP is an optimal solution due to the significant trade-off of qualities between virtual memory and actual memory. Another relevant example referred to Taguchi philosophy [20], [21], in which there are three criteria (Higher is Better (HB), Norminal is Best (NB) and Lower is Better (LB)) for evaluating system quality, since not all qualities have the same characteristics for quality engineers. While these MP theories were proposed as a compromise between extremes by identifying a middle path of a perceived phenomenon, and they may be applicable for observable attributes, they may not be applicable for more complicated scenarios that are not easily measured. They are inadequate to provide rigorous concepts of how to handle more complicated extremism associated with same types of perceptions. Identifying MP can become problematic and undesirable, since it can be just another extreme view in a conflict that requires yet another MP as a compromise.

In addition to identifying a path between two extreme poles, the notion of MP discussed in Eisley and Tang [22, p.254] is taken from the Art of War of Sun Wu (or 'Sun Tzu') which means either of two extreme poles, depending on context. The authors state that "*All or Nothing* are two paths in systems engineering that IT companies are experiencing. MP is a path of flexibility which winds between two radically different poles ... The Middle Path is the path of gradually reconstruction". The authors claimed that MP is the key principle of Sun Wu's art of war and can be effectively used in IT adoption.

Majjhima Patipada or MP in Buddhism is about "avoiding the extremes in our life and finding happiness, joy, and inner peace through the Middle Way" [23, p. 4]. MP is derived from having the 'right view' of a phenomenon. MP in Buddhism does not mean to identify a middle path between contradictory extreme poles, but be mindful to extreme poles that may lead to a condition of attachment, which is the root cause of sufferings. MP is a view that is detached from the identity of a phenomenon. It is a path that is situated between two extreme poles, which is recommended by Buddha to his disciples as a true axiom of all things that protect all humans from being deluded into a condition of attachment. MP is generally used by Buddha to explain the true nature of all things that cannot be defined by either existence or non-existence.

We learnt that the perceived system will be neither craved for sensuality nor craved for non-being if user's attitude to a new system is neutral. In order for the perceived system to integrate with the cognised system, user's neutral attitude towards a system can be another goal of systems engineering. MP can provide a way for a system engineer to work without making precommitment to optimism or pessimism, and without inappropriately prioritising concern for feasibility over desirability or vice versa. In addition, the essence from Buddhist MP can be used to avoid a condition of attachment and also avoid affecting that condition of other people. In order to illustrate how MP should be considered as another goal of systems engineering, we start from when one system has been attached by a user $(\forall\, u : \mathit{user} \mid \exists\, s : \mathit{system} \bullet u \mapsto s \in \mathit{upadana})$. We now describe how an MP can benefit systems engineering using six scenarios below.

1) A user who has attached to one system feels neutral to a perceived system $(\forall\, u \mapsto s \in \mathit{upadana} \mid \exists\, s? : \mathit{system} \bullet u \mapsto s? \in \mathit{uppekha})$.
2) The perceived system will not be craved for sensuality or non-being by the user because the user does not have positive or negative attitude toward the perceived system $(\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \neg\, \exists\, s? : \mathit{system} \bullet u \mapsto s? \in \mathit{kamatanha} \wedge u \mapsto s? \in \mathit{vibhavatanha})$.
3) The perceived system will not be attached for security or insecurity by the user because the perceived system is not craved for sensuality or non-being by the user $(\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \neg\, \exists\, s? : \mathit{system} \bullet u \mapsto s? \in \mathit{upadana} \wedge u \mapsto s? \notin \mathit{upadana})$.
4) There is no intention to adopt or reject the perceived system because the perceived system is not attached for security or insecurity by the user $(\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \neg\, \exists\, s? : \mathit{system} \bullet u \mapsto s? \in \mathit{cetana} \wedge u \mapsto s? \notin \mathit{cetana})$.
5) There is no action to adopt or reject the perceived system because there is no intention to adopt or reject the perceived system $(\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \neg\, \exists\, s? : \mathit{system} \bullet u \mapsto s? \in \mathit{kamma} \wedge u \mapsto s? \notin \mathit{kamma})$.
6) The perceived system state is amalgamated with the attached system $(\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \exists\, s, s? : \mathit{system} \bullet u \mapsto [s/s?] \in \mathit{upadana})$.

[1] The Middle Path and the Solution to Linux Swap, http://feedblog.org/2009/02/14/the-middle-path-and-the-solution-to-linux-swap/

7) The perceived system amalgamated with the attached system will migrate to the user's intention ($\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \exists\, s, s? : \mathit{system} \bullet u \mapsto [s/s?] \in \mathit{upadana} \Rightarrow u \mapsto [s/s?] \in \mathit{cetana}$).

8) The perceived system amalgamated with the attached system will migrate to the user's action ($\forall\, u \mapsto s \in \mathit{upadana};\ u \mapsto s? \in \mathit{uppekha} \mid \exists\, s, s? : \mathit{system} \bullet u \mapsto [s/s?] \in \mathit{upadana} \Rightarrow u \mapsto [s/s?] \in \mathit{kamma}$).

A system engineer can avoid user's negative attitude toward the perceived system in such a way that the user's attached system state will not (or minimally) be affected, so the perceived system can amalgamate with the user's attached system. Any change to the user's attached system can be unnoticed or minimally noticed by the user despite there being a significant change to the user's attached system. It is the only path that avoids *samsara* by not affecting the user's attached system. In other words, Buddhist MP is a path that is aimed to avoid offending the user, who has already attached to a particular system.

V. Buddhist MP Systems Engineering

Buddhist MP systems engineering is recommended to a system project manager as the most desirable way for avoiding system project failures given that the goal of systems engineering is to get the system project accepted by the user. In Human Computer Interaction or HCI, a number of theories have been proposed from different disciplines (e.g., social sciences, cognitive psychology, etc.) in order to promote how a system should be engineered in accordance with user behaviour towards using the system. Some of them were intended to avoid offending users or affecting user attachments, which are coherent with the notion of Buddhist MP systems engineering.

A. Considerate Systems

The notion of considerate systems proposed by Selker [24] is another systems engineering that is strongly coherent to Buddhist MP systems engineering. The philosophy behind considerate systems is that social understanding must be integrated in systems engineering. According to Selker [24, p.1], "system must be adaptive and recognise and learn appropriate times and approaches to communicate a request or provide other feedback".

Disruptions and interruptions were distinguished by the author to explain how a system should be designed in accordance with these two. Interruption is used as positive term when there is an interrupting signal assisting and supporting a user to accomplish the task. An interrupting signal does not offend a user and help a user accomplish the task more effectively. In contrast, disruption is reserved for negative meaning when there is an interrupting signal changing the topic a user is attending to. The author claims that user negative perception to a system can be mitigated if a system is engineered to avoid disruption and pay attention to appropriate interruptions. In this case, the author asserts that feedback is an important element of social responses that validates the level of user understanding, while social responses integrated into a system must be modeled in a polite manner just as when we communicate in daily basis that has different degrees of politeness. Physical and emotional detections such as eyelid and voice recognition are examples of element of considerate systems that understand when and how the system should interact with a user. An interaction with appropriate behaviour is the key element of considerate systems. The author concludes that "considerate systems must hold up their side of an interaction with appropriate behaviour. A considerate system must be able to present itself with the adequate social aspects for the stage of an interaction it finds itself in" [24, p.9].

B. Polite Computing

This notion of Buddhist MP systems engineering can also be illustrated by the theory of polite computing proposed by Whitworth [25], which is a social control to computer human interaction. In ensuring polite computing, the users choices are respected, only useful choices are offered to the user and any important past choices are remembered. Politeness in computing means any practice in computing where all the other users are seriously taken into an account. It is a fundamentally accepted concept but it should be remembered that polite computing actions varies in different cultures. The globally accepted form of computer politeness is the goal it achieves and not necessarily the specific behaviors it identifies with as shown in the information system task concept [26]. Politeness is the ability to be in agreement with the rest when in any social gathering than in situations when you are not but it is different from being etiquette which tends to be more specific oriented. Therefore, the utmost rule of politeness is agreeing to another person's praise without compromise whatsoever.

Polite computing practice offers a choice to the computer users while impolite computing offers no choice. Polite computing therefore is the undesired action that shifts the mind of the computer user from the normal operation. For example, an abrupt pop up window that forces the computer user to look at a different thing from his or her main point of focus [27]. This means the user's choice to view the articles of his choice is denied. This in computer terminologies cannot be termed as illegal but it is considered impolite. Whitworth [25] asserted that many users do not prefer that their computer screens to be commanded by annoying pop-up windows that must be closed from time to time. Spam emails are also other forms of impolite computing because it occupies inbox space without the users consent [28]. Numerous computer users will attempt to fight back but if it persists, they may reduce the time they take when browsing or emailing. Any software that secretly retrieves that computer user's information without his or her knowledge and modifies it will be engaging in impolite computing. Impolite computing will drive away the users as they feel dictated to and may eventually lead to system project failures. The philosophy behind polite computing is that there are no globally agreed standards to guide it. It is up to the computer users to demonstrate behaviors that do not disturb any of the parties using the computer. Applying computer politeness by always considering the user's choices is the most important thing in all corporate businesses aspiring to grow [28].

VI. Applications of the Dhammic Framework

According to Davis and Venkatest [29], it is partially true from system engineer perspectives that a working prototype of a new system can reflect future usage behaviour because the prototype represents functionality toward user requirements. However, dealing with user experiences such as perceived ease of use and perceived usefulness is also valuable, since it can be much more accurate in predicting user behavioural intention and actual usage. According to Davis [30], [31], it is vital to explain and modernize the systems after identifying reasons why potential users accept or reject them. In this case, how a user actually responds to a system has been extensively measured, studied, and evaluated. Psychological conditions underlying human perceptions have been applied in different aspects of systems engineering and related disciplines. Scholars, researchers and organizations in different disciplines have been conducting studies on use and adoption of innovations. These researches aim at identifying constant constructs, factors determining an individual's decisions, satisfaction and intentions of individuals, and use and adoption of innovations [32].

In system acceptance theories, a number of models have been proposed to investigate and understand the factors affecting user acceptance and diffusion of innovations (e.g., Diffusion of Innovation, Technology Acceptance Model (TAM), Theory of Reasoned Action (TRA), Theory of Planned Behaviour (TPB), and Unified Theory of Acceptance and Use of technology (UTAUT)). Among those, TAM is the most widely referred model that is constructed from psychological conditions of individuals in response to a system [33]. TAM has been the only system acceptance model that has received the most attention in the information systems community [34]. TAM was pointed out to be more appropriate than other integrated models that may not have better predictive power for acceptance of innovations [35]. It has been considered as a useful theoretical model in explaining how a user interacts with a system. The relationship between psychological conditions are used to forecast the likelihood of a system project being accepted or rejected by users. TAM posits that an external property influences PEOU (perceived ease of use) and PU (perceived usefulness). ATU (Attitude Towards Behaviour) is influenced by both PEOU and PU. ITU (Intention to Use), a construct for predicting the system actual usage, is influenced by ATU.

Based on psychological constructs in the Dhammic framework, a condition of attachment from Buddhist insights should play a significant role in technology acceptance and adoption and can be considerable for enhancing predictability of TAM. In this case, Dhammic TAM (DTAM), which is based on TAM extended by a condition of attachment in Buddhism, has been proposed for empirical validations. A condition of attachment used in DTAM will follow the notion of Buddhist attachment, which is situated between intensified degree of attitude and intention. For the experiments of DTAM, attachment construct is positioned between users' attitude towards using and behavioural intention. Since a condition of attachment is central to human behaviours in Buddhism that strongly influences behavioural intention, a condition of attachment can be considered as a factor that strongly influences users' actual usage of a system. We have recently obtained ethics approval for conducting an initial round of experimentation which will be described in a future publication.

VII. Conclusions

The Dhammic framework is a completely original framework that represents psychological constructs from Buddhist insights in logical manner. The logical arguments were used to illustrate how a condition of attachment is derived and why and how this condition can contribute to systems failures. Since we live in the world of attachments, it is not surprising a systems engineer wants to build a system based on their own closely held views. However, that is not desirable if it affects what the user has become attached to in the way they have used the system already. While most systems engineers choose the extreme path in order for the system project adopted by the user, the framework suggests that MP is an alternative path that systems engineers can use to avoid *samsara* by not affecting the user's attachment.

The theory can be considered as scientific in the Hopperian sense of being falsifiable, since psychological constructs in the Dhammic framework can be tested empirically. Although this article has not provided clear technical contribution in form of clear definition of the framework to CI at this stage, we believe that the essence from logical arguments used to represent psychological constructs from Buddhist insights can be useful for the theoretical framework of CI.

References

[1] Y. Wang, "Editorial: Cognitive informatics: Exploring the theoretical foundations for natural intelligence, neural informatics, autonomic computing, and agent systems," *The International Journal of Cognitive Informatics and Natural Intelligence*, vol. 1, no. 1, pp. i–x, 2007.

[2] Y. Wang, B. Widrow, B. Zhang, W. Kinsner, K. Sugawara, F. Sun, T. Weise, Y. Zhong, and D. Zhang, "Perspectives on cognitive informatics and its future development: Summary of Plenary Panel II of IEEE ICCI'10," in *Ninth IEEE International Conference on Cognitive Informatics (ICCI)*, Jul. 2010, pp. 17–25.

[3] Y. Wang and Y. Wang, "Cognitive informatics models of the brain," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 36, no. 2, pp. 203–207, March 2006.

[4] Y. Wang, Y. Wang, S. Patel, and D. Patel, "A layered reference model of the brain (LRMB)," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 36, no. 2, pp. 124–133, March 2006.

[5] J. Jacky, *The Way of Z: Practical Programming with Formal Methods*. Cambridge: Cambridge University Press, 1997.

[6] J. M. Spivey, *The Z Notation: A Reference Manual*. University of Oxford, 1998.

[7] J. Woodcock and J. Davies, *Using Z Specification, Refinement, and Proof*. Oxford: Prentice Hall International Series in Computer Science, 1996.

[8] D. Goddard, Ed., *A Buddhist Bible*, 1st ed. Beacon Press, 1938, ch. VII – Self-Realisation.

[9] R. Audi, Ed., *The Cambridge Dictionary of Philosophy*, 2nd ed. Cambridge University Press, 1999.

[10] C. Y. Shen and G. Midgley, "Toward a Buddhist systems methodology 1: Comparisons between Buddhism and systems theory," *Systemic Practice and Action Research*, vol. 20, no. 3, pp. 167–194, June 2007.

[11] D. T. Win, "Ayatanas, sensors and sensing," *ABAC Journal*, vol. 27, no. 5, pp. 40–53, September–December 2007.

[12] M. D. Ainsworth and L. H. Ainsworth, *Measuring Security in Personal Adjustment*. Toronto: University of Toronto Press, 1958.

[13] M. D. Ainsworth, M. C. Blehar, E. Waters, and S. Wall, *Patterns of Attachment: A Psychological Study of the Strange Situation*. New Jersey: Lawrence Erlbaum Associates, 1978.

[14] J. J. Johnson, *What's So Wrong with Being Absolutely Right: The Dangerous Nature of Dogmatic Belief*, 1st ed. New York: Prometheus Books, January 2009.

[15] "Cetana Sutta: Intention (SN 12.38)," Access to Insight, June 2010.

[16] "Bhava Sutta: Becoming (AN 3.76)," Access to Insight, December 2011.

[17] Aristotle, *The Philosophy of Aristotle*, J. L. Creed and A. E. Wardman, Eds. New York: A Mentor Book, 1963.

[18] M. Deslauriers, *Aristotle on Definition*. Boston: Brill, 2007.

[19] R. Eno, *Confucius – A Selected Collection of the Doctrine of the Mean*. Indiana: Indiana University, 2005.

[20] G. Taguchi, *Introduction to quality engineering: Designing quality into products and processes*. Tokyo: Asian Productivity Organization, 1986.

[21] G. Taguchi, S. Chowdhury, and Y. Wu, *Taguchi's Quality Engineering Handbook*. John Wiley & Sons, 2005.

[22] N. Eisley and M. T. Tang, "The middle path: Dealing with transformation in Asia's information technology competitiveness experience," *International Journal of Information Management*, vol. 16, no. 4, pp. 253–276, 1996.

[23] L. Field, *Business and the Buddha: Doing well by doing good*. Wisdom Publications, 2007.

[24] T. Selker, "Understanding considerate systems - ucs (pronounced: You see us)." in *CTS*, W. W. Smari and W. K. McQuay, Eds. IEEE, 2010, pp. 1–12.

[25] B. Whitworth, "Polite computing," *Behaviour and Information Technology*, vol. 24, no. 5, pp. 353–363, 2005.

[26] S. Edgar, *Morality and Machines: Perspectives on Computer Ethics*. Sudbury, Massachusetts: Jones and Bartlett Publishers, 2003.

[27] L. Floridi, *The Blackwell Guide to the Philosophy of Computing and Information*. Malden, MA: Blackwell, 2004.

[28] K. Baldauf and R. Stair, *Succeeding with Technology: Computer System Concepts for Your Life*, ser. New Perspectives Series: Concepts Series. Boston, Massachusetts: Course Technology, 2010.

[29] F. D. Davis and V. Venkatest, "Toward preprototype user acceptance testing of new information systems: Implications for software project management," *IEEE Transactions on Engineering Management*, vol. 51, no. 1, pp. 31–46, February 2004.

[30] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Quarterly*, vol. 13, no. 3, pp. 319–340, 1989.

[31] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw, "User acceptance of computer technology: A comparison of two theoretical models," *Management Science*, vol. 35, no. 8, pp. 982–1003, 1989.

[32] P. M. Silva and A. Dias, "Theories about technology acceptance: why the users accept or reject the information?" *Brazilian Journal of Information Science*, vol. 2, no. 2, pp. 69–86, 2007.

[33] M. Y. Yi, J. D. Jackson, J. S. Park, and J. C. Probst, "Understanding information technology acceptance by individual professionals: Toward an integrative view," *Information and Management*, vol. 43, no. 3, pp. 350–363, 2006.

[34] M. Chuttur, "Overview of the technology acceptance model: Origins, developments and future directions," *Sprouts: Working Papers on Information Systems*, vol. 9, no. 37, pp. 1–20, 2009.

[35] Y. K. Chau and J. Hu, "Information technology acceptance by individual professionals: A model comparison approach*," *Decision Sciences*, vol. 32, no. 4, pp. 699–719, 2001.