Snap-Cards: A Dynamic Data Construct of Rapid Information Gathering and Integration for C2 Effectiveness in Homeland Security

Qiuming Zhu
Department of Computer Science, University of Nebraska at Omaha, Omaha, NE 68182
Zhuq@unomaha.edu

Jeffrey D. Hicks, Richard Flanagan, Alexander Stoyen
21st Century Systems, Inc., 6825 Pine Street, Suite 101, Omaha, NE 68106
Jeff@21csi.com

March 24, 2004

Snap-Cards: A Dynamic Data Construct of Rapid Information Gathering and Integration for C2 Effectiveness in Homeland Security

Qiuming Zhu
Department of Computer Science, University of Nebraska at Omaha, Omaha, NE 68182

Jeffrey D. Hicks, Richard Flanagan, Alexander Stoyen
21st Century Systems, Inc., 6825 Pine Street, Suite 101, Omaha, NE 68106

ABSTRACT

A dynamic data construct built on the basis of a snap-card paradigm is designed to meet the needs of information empowerment, formation flexibility, and representation versatility for rapid interchange and effective integration of information in complex environment. It facilitates the prompt reporting, gathering, tracking, and analysis of information from wide spread of heterogeneous resources, and effective decision making and event responses. The construct is especially designed to operate in wireless communication environment where the data amount contained in the transmission must be tightly packed. The technique is suitable for applications particularly in C2 situation awareness of battle space and security surveillances of airports, military installations, and occasions of large public events.

Key words and phrases: Snap cards, Dynamic data structure, Hyper-links, Rapid information exchanges, Wireless communication, Data fusion.

I. INTRODUCTION

The contemporary battlespace of military operations, particularly with the emergent homeland security operations, requires a means of modern information technology that can carry and deliver the time-critical information to war fighters promptly. The demand of providing the right information to right place at right time and in right format so that it is understandable easily and accurately requires the information to be flexibly coded, concisely presented, and quickly transferred. The information transferred among different units, places, and operational aspects need also be quickly integrated. Most current information systems are designed on the bases of allowing large amount of data storage, sharing, and transition, but not aimed at also facilitating the integration of information from multiple heterogeneous sources. For example, the data structure in most of the existing systems does not allow for effective integration of varying information contents and formats, and open for diversity of information resources. The rapid development and deposition of wireless communication networks also require a means of dynamic, adaptive, hyper-linked, open data structure that can flexibly encode information in various formats (multi-media, many formats, text, graph, image, audio video, etc.) concisely and deliver them efficiently.

A dynamic data construct built on the basis of snap-card paradigm (and technique) is designed in our research to meet the needs of information empowerment, format flexibility, and representation versatility for the rapid interchange and effective integration of information in complex environment. The concise and adaptive data structure facilitates the rapid reporting, gathering, tracking, and analysis of information from wide spread of heterogeneous resources, and prompt fast decision making and event responses [GS92, DT99]. The construct is especially designed to facilitate information fusion on data from heterogeneous resources. The incremental-complexity structure has particular advantage for operating in wireless communication environment where the data amount contained in the transmission must be tightly packed, and is suitable for applications particularly in C2 situation awareness of battle space and security surveillances of airports, military installations, and in occasions of large public gathering events.

A system of the dynamic data construct consists of four main functional blocks: (1) A networked environment where human operators (watch standers, security guards, war fighters), automated sensors, and intelligent software agents serve as the main functional entities to participating in observing, gathering, reporting, and analyzing information in each of their surrounding world [HSZ01]; (2) Snap cards that are created, transmitted, integrated, and responded through the software agent operations (for example, a PDA worn by a watch stander) taking place in the networked environment; (3) A situation Awareness Processor (SAP) that collects the snap cards issued and transmitted in the network, analyzes the contents of the snap cards, and fuses the snap-card information through the use of an Activity Assessment Matrix (AAM) [Zhu94]; and (4) System control agents that form a hierarchical structure of functional blocks for coordinating the snap-card transactions, archiving, processing, and determining proper responses to arising situations. There are different types of software agents, each responds to a specific task, such as access control, event inference, parameter adaptation, etc. in the dynamic system construct. For example, an event control agent in the system agent assembly is in charge of setting the surveillance priorities and conducting preliminary processing of the security-concerned events. .

The dynamic data construct bases its functionality on the concise and flexible snap-card information exchange and transmission format and protocol [AMS96]. The contents of snap cards are organized in relation to the pre-defined activities to be monitored in surveillance or real time events to be reported. It consists of a few basic entities plus a number of optional and dynamically adjustable entries. The snap-card based dynamic data system construct enhances the autonomous capability of intelligent agents [CZC01]. The basic entries of a snap-card include (1) Card identifier that is a unique code containing the information about the source of the card (location of a watch stander or automated sensor) and the priority of the card or its issuer for tracing the specific card in a networked environment. (2) Card suite that defines the severity and emergency levels of the events the card is reporting. (3) Card values that represent the specific threat activity in certain pre-defined event types and zoning areas. (4) Card links that connects relevant data and interactions together. In the optional and extendible part, a snap-card can include multimedia pieces of information to give a more detailed description of the reporting event, for example, a snap shot of the suspect or the scene of the event. The snap cards are interoperable on heterogeneous computing platforms.

The snap-card based dynamic system construct enhances the autonomous capability of intelligent agents [SZ96, Dawi99]. It makes the agents more flexible to make re-configurations and to adapt to different environments since knowledge and inference rules are not hard-coded in the agents but incorporated in the snap-card paradigm. The construct aligns with an open system concept that allows interchangeable hardware and anywhere executable software, thus leverages the computational power of the agents maximally. By collaborating on the snap-card platform, the agents will be much less replying on a central processor for the distribution and coordination of tasks, while keep the tasks they are executing synchronized [KHBM96]. The system can also be conveniently migrating between different applications, by incorporating different program codes with the snap-card instances.

The dynamic system construct is prototyped and tested in a Surveillance System Concept (SSC) named Sentinel Net for Force Protection (FP), developed in a SBIR project. The Sentinel Net offers naval forces advanced Detection, Indications and Warnings (DIW) of terrorist/asymmetric attack against U.S. naval vessels docked at a foreign port, ships at anchor, naval facilities, and forward-deployed forces (Naval and Marine). The system detects and tracks anomalous and/or suspicious activities, maintains a level of security while reducing manning requirements. It has proven that the snap-card based system construct is a viable means for rapid information integration and effective detection of suspicious and threatening events to build a complete picture of situation awareness for force protection and security operations.

II. THE SNAP-CARD CONSTRUCT

II.1. Motivation

Traditional data structures of computing and communication are designed mainly as data storage or transmission units used for holding static information. The size (volume) of the structure is usually a constant (fixed) when it is placed in use. As the applications of computing and data communication expanding from traditional numeric domain to symbolic domain, including multi-media, semantic networks, interactive human-machine interfacing, intelligent and autonomous systems, group collaborations, distributive decision making, process control and automation, the static data structure become less and less adequate for meeting the complexity and diversity of computations in these advanced applications. For example, the static data structure does not help much to meet the increasing demands of the tasks in data fusion and integration that are necessary for most hybrid dynamic systems in those advanced computer application domains.

A Snap-Card paradigm and technique is designed to meet the needs of data structure empowerment, flexibility, and versatility for the fundamental data representation, rapid interchange, and effective processing in modern information systems and technologies. Simply put, the Snap-Card data structure is featured with (1) dynamics, (2) self-organizing, (3) variable complexity, and (4) incremental construction, and (5) action embedment; and is suitable for a diverse set of applications of advanced computing systems that are intelligent, autonomous, and self-adaptive.

II.2. Overview

An information technological functional view of the Snap-Card technique is that it is an efficient and prominent information carrier (a smart media) for information sharing and collaboration. In the idea of smart media for computing, a data/information holder is no longer simply acting as a carrier of the information, but also playing a role as a participant, an executor, a controller, a mediator, and a driving force of certain events taking place in the surrounding environment. The Snap-Card data structure is to facilitate the information reporting, gathering, tracking, analysis and integration from wide spread heterogeneous resources instantly, thus to facilitate the situation awareness, knowledge dissemination, and decision making in military C2 and homeland security applications.

The Snap-card paradigm has the following main characteristics:

- Snap-Card is purposefully designed to present itself in the computation in incremental levels of complexity and sophistication.
- A Snap-Card can be created and activated automatically at different levels in terms of information contents it contains and the complexity of user interface.
- The operation levels are selectable and adjustable dynamically and autonomously: starting with lower, simple level to progress to more complex and sophisticated levels that contains more information and information with more diverse formats such as audio and video.
- In creating cards from simple level to more completed levels, the card automatically gets into the category of A, B, C, D, E, and F (details described in the following). That is, when information is entered into the structure, the corresponding types of cards are generated automatically (through proper software design).

Each entry of a Snap-Card can have different attributes. A Snap-Card can also contain a piece of software component (e.g., codes executable on heterogeneous computing platforms). This ability increases the flexibility and security of the agent activities. For example, when an agent A needs the collaboration of another agent B to perform certain task or assistance in certain task, the agent B does not need to have the executable code pre-stored in it. The agent A can send to agent B not only the request of task and necessary data but also the necessary program code (or give the address of the data and code) for the agent B to perform the task. Since the system does not need to store executable codes in every place of the agents, it reduces the risk that the codes may become rotten or incompatible because of lacking maintenance or under hostile attack. Meanwhile, since the activities of an agent can be controlled by what program code it receives, the agents are flexible to play different roles and act adaptively under different situations.

II.3. Definitions

The contents of snap cards are organized in relation to the pre-defined activities to be executed or real time events to be reported. Each card includes a set of primary entries such as

(1) Card identifier – that is a unique code for tracing the specific card in a networked environment and for processing the specific card.
(2) Card category – that distinguishes the specific types of a card, for example, whether it is a (a) data card, (b) event report, (c) action request, and (d) operation response.
(3) Card face value – depending on the card category, different contents can be included.

(4) Card links - a sequence of hyper-links to indicate the locations of relevant data and execution agent options.
(5) Card data – The main body that accommodates variety of data types and contents and is flexible in organization.

More detailed description of the card entries, as examples, is presented in the following.

II.4. Classifications

A.1. The **Abstract** Card structure — A-CARD

The A-CARD contains a fundamental data member - Card ID. The card analyzer uses the Card ID to access the location, holder, and other Card-related information that is useful for the data analysis. A set of virtual functions for fundamental card operations, such as the Card Set, Reset, Read, Write, Modify, and Transmit are defined. The A-CARD is not necessary to be a physical card. It can serve as a virtual card structure to be instantiated/inherited by other card types. Once it is instantiated and filled in values, it becomes a B, C, D, E, or F card that is a physical card to be actually transmitted and processed.

A.2 The **Basic** Card structure — B-CARD

A B-CARD can have basic suit types, such as the set {♣ ♦ ♥ ♠}, and background in colors or graphical patterns in addition to the fundamental members of the A-CARD. A B-CARD can also have face values in types of integer or real numbers. Other possible information in B-card includes card format specifics and menu selected options.

A.3 The **Commentary** Card structure — C-CARD

A C-CARD data member includes Text Message, a set of fields specifications in forms of word-value pairs, and event description in user defined language, in addition to all fields inherited from the B-CARD.

A.4 The **Data-gram** Card structure — D-CARD

A D-CARD can accommodate Tabular report with a set of buttons called Persistent Tabular Buttons – PTB to set the table values, in addition to the fields contained in the C-CARD type. An example of the possible data field can be shown on the table at the right.

PTB	Value setting
Event to report	An unidentified object
Object parameters	Distance, speed, direction, …
Emergency level	Within 1000m
Certainty level	Visually observable
Exception	Exception condition?
Response/reaction	Warning? / Destroy?
Hyper-links	Links to other event reports

A.5 The **Extended** Card structure — E-CARD

An E-CARD crosses Multimedia – images and voices are part of the card content. The Card can contain buttons of several kinds, text fields of varying lengths, and color or black-and-white graphics; in addition to any field included in the D-CARD.

A.6 The **Function** Card structure — F-CARD

The F-CARD accommodates the inclusion of software functions – a piece of program or a segment of executable code that, once received by the destination station, can be executed automatically or under the receiver's control and discretion.

Figure 1 below shows the inclusiveness relations of snap-card construct classifications. The transitional relations of the snap-card construction are shown in Figure 2.

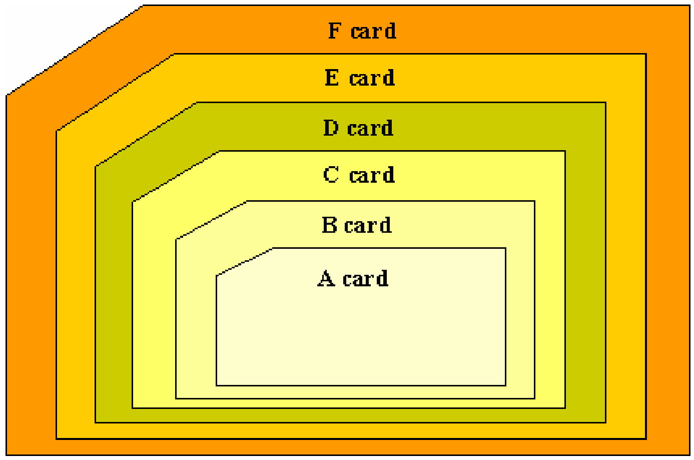

Figure 1. The inclusiveness relations of snap-card construct classifications.

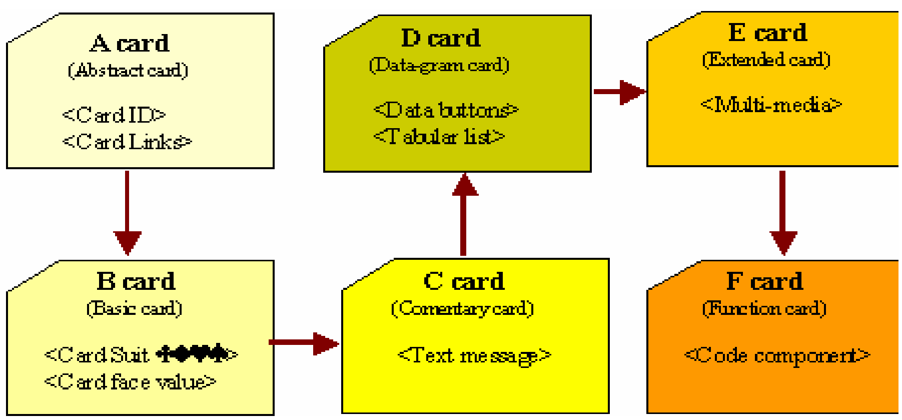

Figure 2. The transitional relations of the snap-card construction.

III. THE SNAP-CARD SYSTEM CONCEPT AND FUNCTIONS

We discuss the system concept of the snap-card in this section. The Snap-Cards are interoperable on heterogeneous computing platforms. On one hand, the operation of a Snap-Card system operates resembles the message-passing scheme for intelligent system cooperation. On the other hand, it contains more functions than the message passing. The Snap-Card system aligns with an open system concept that allows interchangeable hardware and anywhere executable software, thus leverages the computational power of the agents maximally. The Snap-Card system can be conveniently migrated between different applications, by incorporating different program codes with the card instances. It also makes the agents more flexible to make re-configurations and to adapt to different environments since knowledge and inference rules are not hard-coded in the agents but incorporated in the Snap-Cards.

III.1 The snap-card system architecture

The snap-card system construct consists of four main functional blocks:

(1) Network environment – The snap cards system runs in an agent network environment. That is, the set-up, activation, access, and utilization of the cards system will all be done automatically through the card operation agents active in a distributed network.

(2) Snap-card stacks – The snap cards are organized in stacks according to event categorizations in the network environment. The cards can be created, organized, accessed, and modified through operations that can be performed either remotely (via network) or locally.

(3) System control agents – The system control agents coordinate the creation, modification, access, and activation of the snap-card system. There are different types of agents, each responds to a specific task, such as access control, event inference, parameter adaptation, situation assessment, etc.

(4) Card base/repository – The snap cards are processed in a card queue. Snap cards are archived in a card base/repository. The repository organizes cards according to types of event reports and other information from multiple sensors, observers, and other information sources. There is an event control agent in the system agent assembly in charge of setting the event processing priorities and preliminary processing of the events reported in snap cards.

III.2 Snap-card operation functions

To make effective use of the snap cards, a set of functions for card handling methods are designed. Some basic operations of the snap cards are:

(1) Card creation – This operation is invoked when a situation arises that either an information sharing or a collaboration of agents must take place.

(2) Card posting – This operation deal with the questions of to whom (agents) the card should be sent, or where (in what scope) the card should be cast.

(3) Card retrieving – One of the snap-card features is that once a card is posted, it can be retrieved and executed autonomously by responding agents. This operation allows agents to identify the cards that are relevant to their roles and functionalities, and actively participating in the collaboration.

(4) Card execution – Once a card is received (or retrieved), the card category entry identifies the activity to be carried out. The receiving agent is responsible for processing the data or executing the program specified in the card entries.

(5) Card archiving – There is a central tracker (an agent) in the snap-card system that performs the collection, sorting, and archiving operation on the cards ever issues and processed in the system. The cards are sorted according to the categories before they are stored.

(6) Card modification – Occasionally an agent receiving the card will need to make certain modifications to the card, for example, adding certain information or marking certain fields, and then post the card again.

The snap cards are organized in stacks in the card tracker and maintenance database. Each stack corresponds to a specific type of event or certain task classifications. The cards in each stack are not necessary to be stored in the same physical location. They can be distributed in different locations, such as located in both the central command office and local site (e.g., the agents). The cards can be formatted in the same way as a simple Web page that can be remotely accessed and processed.

III.3 Snap-card transmission - Communication modes

Communication is the most important function for the snap-card system construct. The snap-card can be transmitted using either one of the following communication schemes:

(1) Uni-cast – a point-to-point transmission from a source to a destination, where both the source and the destination can be a card agent, card tracker or a human user,

(2) Multi-cast – a one-to-many transmission from a source to multiple destinations registered in the system, and

(3) Broad-cast – a one-to-all transmission from a source to all valid card destinations designated by the system.

All Uni-cast transmissions should have closed loop acknowledgement. Multicast should have an available periodic status checking to verify that transmissions are not lost. No feedback checking is necessary for the broadcasting transmission but a flagging to signal the card transmission is enabled.

In the case of broadcasting, the execution of the card is controlled autonomously by the receiving agents. Once the snap-card transactions start taking place, the collaboration process begins. Collaboration also takes place between the users (humans) and the computer system (agents) operating the snap cards, and between the users and the central tracker. The central tracker of the snap-card system oversees the activities and operations of the snap cards to ensure that each snap-card is properly processed. The central tracker is also agent-based. The agent will attempt to use the quickest method to try to converge on a decision whether activity is carried normally. A number of ways for judging whether an event is anomalous enough to warrant interference should be adapted.

III.4 Wild card and wild matches.

Wild card and wild card matches should be allowed and activated in the snap-card system construct – i.e., a (wild) snap-card may match several snap-card integration slots in the automatic assessment processor (AAP). By wild snap card, we mean that the card contains some field that is not (or cannot be) clearly specified, such as the event type reported (e.g., just known a serious threat from air is presenting). Only a few designated filed of a card can be set wild. Not every filed of a card can be set wild. A wild card match slot in AAP may be matched with several different snap cards.

A wild card is kind of different from the incomplete card where some card entries (fields) are missing because of lacking the information. Sometimes, an incomplete card may be treated as a wild card in terms of the field it misses (if it just fit into the definition fields of the corresponding snap-card type.).

IV. SNAP-CARDS IN C2 AND HOMELAND SECURITY APPLICATIONS

IV.1 Snap-card applications in general

The snap-card technique can be used in many computing and communication systems. Some of the application examples are as the following.

1. The technology can be used among a group of collaborators (engineers, architects, Subject Matter Experts, etc.) to share data/information, to work on drafts/plan/blueprints, to cooperate on problem solutions.
2. The technology can be used for carrying/transmission multi-source, geologically distributed intelligence (e.g., from a sensor grid) in command and control situations.
3. The technology can be used by a group of decision makers to lay out, assess, revise, finalize decision plans on a common (everybody seeing the same thing) visualization platform.
4. The technology can be used by managerial and military commanding officers to issue, track, and inspect working plan, to check schedule and assignment within a group of personnel at different geological locations, and to coordinate plan executions.

An application of Snap-Card system method to an event management in business administration or military commanding situation, including task scheduling and control of an agent group, can be illustrated by the following example.

1. When a task comes up – may be initiated by an agent "A" in a computing grid (e.g., a sensor agent detects a hostile object approaching), the agent "A" sends an alert and request-for-service (ROS), with descriptions of the observed situation, to every other agents (broadcasting) in the grid, through an issuing of a particular typed Snap-Card – i.e., a card is *created*.
2. The request-for-service (ROS) is visible to all agents, and could be responded by any of the agents in the grid. The agent who has the capability to answer the ROS, has available (computing) resource, is to pick up the hyper card. The Snap-Card is marked immediately once it is picked up by one agent, say agent "B" and a hyper link between the agents "A" and "B" is *created*.
3. The agent "B" now is to respond to the request specified in the hyper card, and supposedly to carry out the specified task. If the agent "B" is capable of completing the task, it completes the task, mark the snap card, and send the marked card to the "task tracking agent" for archiving (reporting, sorting and storing). We say the Snap-Card is *signed off*.
4. In most cases, the task requires other agents' collaboration. The agent "B" will then create one or more "ROS" snap cards, according to its analysis and judgment, to the agent grid. These cards have hyper-links to the previous card, which is now a "master card." The new cards are broadcasted to the grid and processed in the same way as the master card.

5. If a ROS snap-card is not picked up (no activity agent in the grid respond to the ROS), a default agent – "card tracker agent" will pick up the card. The card tracker agent is not going to execute the task specified on the card, but to find a way to re-distribute the card to a specific agent or a group of specific agents. There could be more than one car tracker agent in the grid.
6. The card tracker agent will in charge of analyze the card, and identify corresponding agents that are capable of process the ROS according to the type of the card or the task specified on the card. When no capable agent existing in the grid, the card tracker agent will search the agent registrar in the grid or hand off the task to a coordinator agent. The coordinator agent could designate an agent who is idle to alter its role (if possible) for the designated task, and send the program code (retrieved from a code bank/base) along with the card and requested task. The coordinator agent may choose to re-schedule the tasks that are carrying out by the identified agents in terms of the priorities of the current task and the task on the card. The process sequence of the above is shown graphically in figure 3.

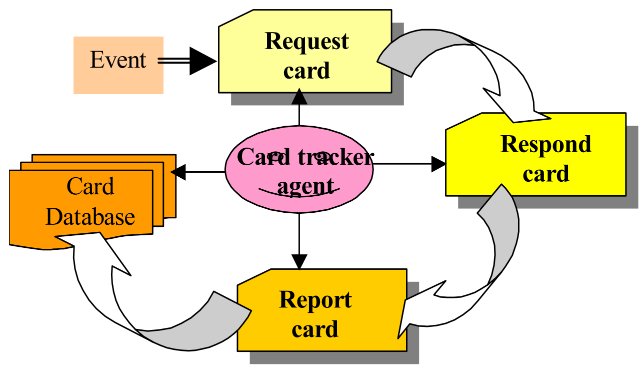

Figure 3. Snap-card event transaction cycle.

IV.2 Sentinel Net – A snap-card applications in force protection

In the final, we are going to illustrate an application of snap-card to military force protection. The "Sentinel Net" is a project we conducted to provide force protection for Navy ships anchoring in a foreign port, where a scenario is shown in figure 4. The system concept can also be applied to many other military and civilian situations. There are several agents designed in the Sentinel Net system, for example, the sensory and information management agents, the in the system, and the decision support agent for deciding whether a Warning, Caution, or Advisory should be issued according to the Activity Assessment Processor (AAP) – a snap-card integration and analysis agent. Key functional agents for the snap-card processing include (1) Card Tracker for packing snap cards according to source, destination, card flow and transmissions, (2) Card Analyzer for content extraction, event-type identification, timing, situation assessment (the main component of card analyzer is the Activity Assessment Processor), and (3) Card Organizer that is responsible for card archiving, user management, system access control, and security measures of the system.

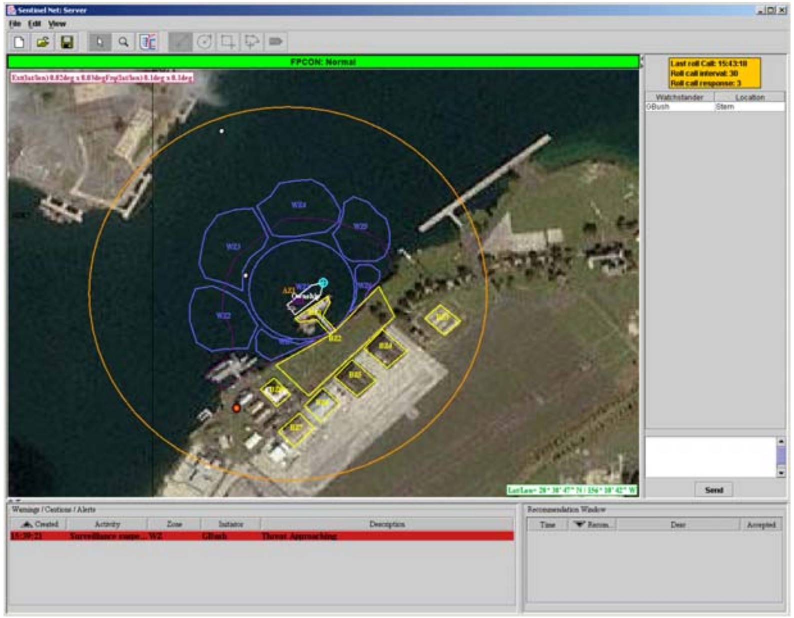

Figure 4. A force protection scenario for Navy ships anchoring in a foreign port

The agents communicate with each other using snap card. In addition, snap cards are enabled on a number of PDA (Personal Digital Assistant) devices carried by watch standers located at different part of the ship and the port. The watch standers are to guard the ship, watch any activities happening surrounding the ship and at any place in the water, air, and port areas, and reporting any anomalous events using the snap-card enabled on the PDA. Examples of the snap-card on PDAs are shown in figure 5 at the right.

Figure 5. Snap cards on PDA for rapid information exchange in Sentinel Net.

The Activity Assessment Processor of the snap-card system in the Sentinel Net makes use of an *Activity Assessment Matrix (AAM)* for correlating, integrating, and inferring information reported on snap-card related to ship security and safety. Each designated sentinel zone of the watching area has an Activity Assessment Matrix (AAM) designed for detecting and identifying possible threats. AAM specifies the activities to monitor/watch, (what kind of activity? Does any card information match the watching activity defined in the AAM?) The content of AAM is established off-line before the deployment of PDAs. AAM resides in the Activity Assessment Processor, and is to be manipulated by the Card tracker, card analyzer, and card organizer agents. The AAM structure/scheme facilitates the needs for system expansion, extension, and scalability. There will be multiple AAMs, each for a specific zone (W, A, B, U, S) or for specific type of threat (Chemical, Energy, Explosive, Biological, …). To add more threat detection types, activities, and intelligent sources, just add one more AAM. – The capacity of the AAP can be dynamically increased easily, as the desire, and as the need of field commander and situations. A schematic illustration of the AAM is shown in figure 6.

8♠ R1	3♥ H2	5♠ R3	8♣ B3	4♥ S4	7♦ A6
8♣ D1	8♦ F4	4♥ B5	5♥ S5	1♣ B2	8♠ R4
2♦ C2	8♦ F4	4♥ B5	8♣ B3	8♥ S2	3♣ H3
8♣ D1	8♦ F4	8♥ B3	5♠ A4	6♣ S5	7♠ W3
8♠ R1	3♥ H2	6♣ S5	8♣ B3	7♠ W3	8♥ S2

Figure 6. A schematic illustration of an AAM for snap-card information fusion.

Note that the structure of the element of an AAM resembles the structure of a snap-card. It is true and is intended. This is because of the basic matching operation to be performed between the snap-cards and the AAM elements (Card inference and Sentinel flag setting will be based on the match of the card issued by PDAs and other information sources with the elements of the AAM.) Each row of the AAM is a set of activity measurements that lead to a specific threat indication and warning consequence (WCA). These rows and the contents of AAM are predefined according to scenario specification. The elements of AAM are not necessary to be all the different (duplicated elements are permitted, since one activity ID may contribute to different WCAs). Some elements of an AAM could be empty (or serve as wild slots, see discussions later). Empty elements have a default color.

Two levels of pattern match take place at an AAM in the AAP: (1) Event detection level matching where each event is represented by a snap card, we also call it Card-level pattern match because it takes place between each individual card and each individual AAM slot, and (2) Situation awareness level matching that observes a combination of events through the integration and fusion of the card events, we also call it the Matrix-level match because it takes place between cards on a row of AAM and the backend of AAM. These two levels of pattern matching operations are corresponding (coincident) to two levels of information fusion – fusion level 2 and fusion level 3. The event matching helps to (contribute toward) build the situation awareness; while the building up of the situation awareness (through integration of multiple cards) helps to clear up the uncertainties at event level – can provide assistances in the handling/ proper reaction selection of the individual events (such as an anti-terror action/reaction). Figure 7 shows a diagram for illustrating the AAM pattern matching. Note that the Activity Characterization may be generated automatically by the AAP based upon the information transmitted in snap cards, and in coordination with the information fused from and integrated with other intelligence sources.

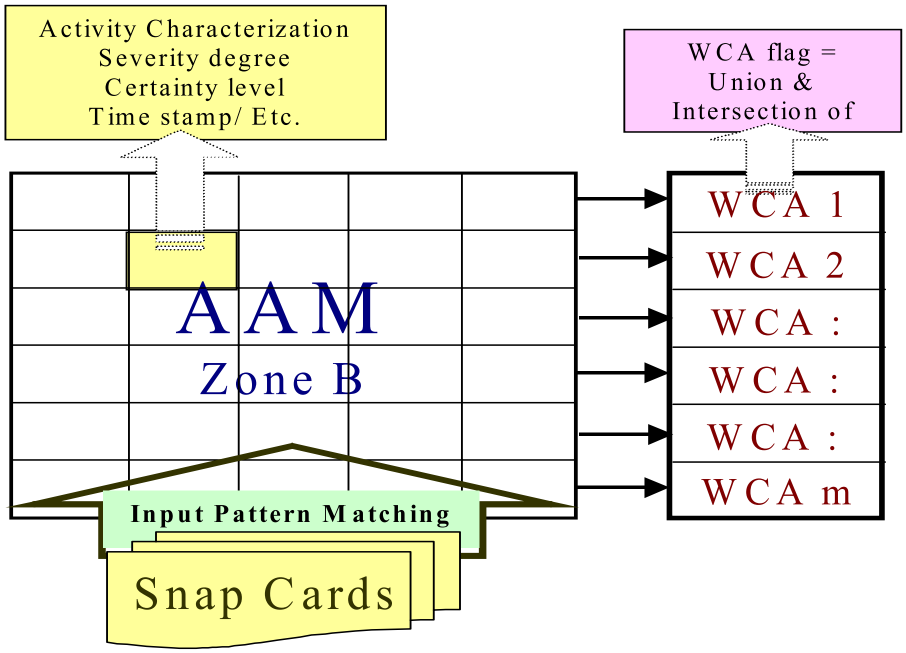

Figure 7. A diagram for illustrating the AAM pattern matching.

The snap-card operations on the PDA are activated by the watch standers in the following sequence:

1. To create a card, a watch stander lays out a card object using either a card-specific or a common background.
2. A template of a B-CARD must be already instantiated in the PDA when the PDA is deployed into service.
3. A security function such as user id/password, biometric, voice, and PIN types might also have been established through the PDA operations acting onto the card instance.
4. Once released by a watch stander or generated automatically by a sensor device, the snap-card flies from its source to destination autonomously.
5. Normally, cards go to the "Card Tracker" "Card Analyzer" and "Card Organizer," but certain cards may go to specific locations also.
6. The issuing of a card creates a thread of responses - A card induces a thread of activities. A sequence of actions on the system devices is triggered, including card communication, card interpretation, card tracker, information gathering, integration, assessment, warning/alert issuing, actuators, etc.
7. Functions for automatic / semiautomatic activation of card creation and issuing (transmission) upon certain events broken-up may be developed and installed.

Card transmission uses either Uni-cast or Multicast scheme. In Uni-cast, the message (card content) transmission should be in closed loop: Issuing a card from PDA ---> Card Tracker ---> PDA (acknowledge the successful transmission). Roll call is employed to verify network configuration status by checking each PDA through requesting a response of "here" from every PDA. The Card Tracker verifies that last roll call time is most current. Roll call would require that PDA return "here" message within a time interval to be valid. If PDA is capable of operation OTHER than Sentinel Net functions then a current operational mode indicator is needed to check what "state of mind" the user/PDA is in. A persistent transmission/reporting mode allows for an enhanced security measure. Once a snap-card (or a button) is set, the PDA will send out the card automatically and persistently unless a watch stander (PDA holder) cancels the action. This is to ensure that any substantial threat be reported without lost even under harsh situation, and enhance the reliability of card communication

V. CONCLUSION

Snap card paradigm is dynamic, adaptive, content-based, with incremental complexity, hyper-links, and is an open structure. It is a data format protocol, a data structure (logical view), and an efficient /prominent information carrier (physical view) - a smart media and protocol for information sharing and collaboration. An event driven, action embedded active data structure. It is designed to facilitate effective wireless communication and to facilitate information fusion. It also facilitates data exchange in multi-agents interaction and facilitates task management in complex systems.

The Main features of the snap-card technology can be summarized as the following.

1. A uniform, standardized, and versatile information representation, transmission, interchange, and sharing data structure which is also a format, a media, and a protocol in a networked multi-team and multi-user collaboration environment – a tool and protocol that integrates and converts heterogeneous information from multiple resources into a uniform and easy-to-manipulate representation.
2. An event driven, action embedded active data structure (data + functions) and pattern match mechanism that enables automatic reasoning, data fusion, and integration of uncertain and incomplete information, all carried out directly at the components (cards) level.
3. A task management tool that automates, or assists automating, the event and activity threading, tracking, auditing, and record keeping.

The overall characteristics of the snap-card system construct can be summarized as the following.

1. A dynamic data structure facilitating command and control of dynamic systems and process automation,
2. A smart-media/intelligent data structure facilitating data fusion/integration operations,
3. A rapid information exchange format facilitating instant wireless communication,
4. An incrementally constructible data facilitating user interface diversity,
5. A content-based autonomous data formatting paradigm,
6. A complexity variant communication protocol,
7. A volume variant and format adaptive multi-media data structure,
8. An action-driven, hyper-linked, multi-thread data representation scheme, and
9. An Open-structure data representation paradigm.

The snap-card system construct is designed to especially facilitate rapid and effective wireless communications where the data amount contained in the transmission must be highly volume-packed. It also presents as a new attempt of data structure design that particularly facilitates information fusion and integration.

Reference:

[Dawi99] E. Dawidowicz, "Intelligent Agent Technology in Command and Control Environment," Proceedings of 1999 Command and Control Research and Technology Symposium, Newport, Rhode Island, June 29-July 1, 1999.

[AMS96] C. Amaro, T. J. Marlowe, A. D. Stoyenko, "An Approach to Constructing Complex Evolving Systems Using Composition of Knowledge Domains," *21st IFAC/IFIP Workshop on Real-Time Programming,* November 1996.

[CZC01] M. Chen, Q. Zhu, and Z. Chen, "An Integrated Interactive Environment for Knowledge Discovery from Heterogeneous Data Resources," *Journal of Information & Software Technology*, Vol. 43, pp. 487-496, 2001.

[DT99] J. Doyle and R. H. Thomason (1999), "Background to qualitative decision theory," *AI Magazine*, Vol.20, No.2, pp. 55-68.

[GS92] P. A. Gibson, A. D. Stoyenko, "Development and Integration of a Concurrently Executing Interactive User Interface for the I-STAT Portable Clinical Analyzer: A Case Study in Real-Time Systems Integration," *Journal of Systems Integration*, pp. 349−388, Vol. 2, No. 4, October 1992.

[HSZ01] J. Hicks, A. Stoyen, Q. Zhu, "Intelligent Agent-based Software Architecture for Combat Performance under Overwhelming Information Inflow and Uncertainty," Seventh *IEEE International Conference on Engineering of Complex Computer Systems,* pp.200-210, June 2001.

[KHBM96] S. D. Kushner, C. H. Heithecker, J. A. Ballas, D. C. McFarlane, "Situation Assessment Through Collaborative Human-Computer Interaction," *Naval Engineers Journal*, 108(4), pp. 41-52, July 1996.

[Setal93] A. D. Stoyenko, T. J. Marlowe, W. A. Halang, M. Younis, "Enabling Efficient Schedulability Analysis through Conditional Linking and Program Transformations," *Control Engineering Practice*, Vol. 1, No. 1, pp. 85-105, January 1993.

[SZ96] K. Sycara and D. Zeng (1996), "Multi-Agent Integration of Information Gathering and Decision Support," *Proceedings of the 12th European Conference on Artificial Intelligence*, 1996.

[Zhu94] Q. Zhu, "Probabilistic Reasoning in an Augmented Fact-Proposition Space and its Applications," *Journal of Engineering Applications of Artificial Intelligence*, Vol. 7, No. 6, pp. 627-637, 1994.