Mastering Situation Awareness:

The Next Big Challenge?

Dieter Gawlick Eric S. Chan Adel Ghoneimy Zhen Hua Liu
Oracle Corporation
500 Oracle Parkway Redwood Shores, California, USA
{dieter.gawlick, eric.s.chan, adel.ghoneimy, zhen.liu}@oracle.com

ABSTRACT

John Boyd recognized in the 1960's the importance of **situation awareness** for military operations and introduced the notion of the OODA loop (Observe, Orient, Decide, and Act). Today we realize that many applications have to deal with situation awareness: Customer Relationship Management, Human Capital Management, Supply Chain Management, patient care, power grid management, and cloud services management, as well as any IoT (Internet of Things) related application; the list seems to be endless. Situation awareness requires applications to support the management of data, knowledge, processes, and other services such as social networking in an integrated way. These applications additionally require high personalization as well as rapid and continuous evolution. They must provide a wide variety of operational and functional requirements, including real time processing.

Handcrafting these applications is an almost impossible task requiring exhaustive resources for development and maintenance. Due to the resources and time involved in their development, these applications typically fall way short of the desired functionality, operational characteristics, and ease and speed of evolution. We – the authors – have developed a model enabling the development and maintenance of situation-aware applications in a declarative and therefore economical manner; we call this model **KIDS** – Knowledge Intensive Data-processing System.

1. INTRODUCTION

We published a formal definition of the KIDS model in [4] which deals with the situation awareness as documented in [16][21]. An effective support of situation awarness is at the core of many applications. This has been attempted by developing appropriate data structures, procedural code, and processes. To simplify the application development, the IT community has for years abstracted out these three important aspects; with varying degree of success and level of maturity. Data management was the first technology to be abstracted out; we could not write modern and mission critical applications without the existing database technology. Process management in the form of workflow systems and knowledge management in the form of rules, analytics, etc., have significantly simplified the management of the other two aspects of the applications.

However, data, knowledge, and process management are treated as three orthogonal and independent aspects of applications and are therefore separately managed by three independent technology platforms. If we try to use these technologies to solve more complex problems, such as situation awareness, we find that data, knowledge and process management are increasingly intertwined. Unfortunately, there is no high-level model managing these three aspects in a consistent way. To deal with this challenge we propose a model that helps applications to manage data, knowledge, and processes in a synergistic, consistent, and well structured way. This model is based on the observation that humans typically solve problems using a loop where they capture facts, condense the facts by applying knowledge, reason about the findings, and act. More specifically, we capture *quantitative* facts, classify the facts to derive compact *qualitative* perceptions, assess the perceptions to arrive at one or more hypotheses, and use these hypotheses to formulate directives, unless we decide that nothing should or could be done. The resulting directives will be acted upon to create new facts – see Figure 1.

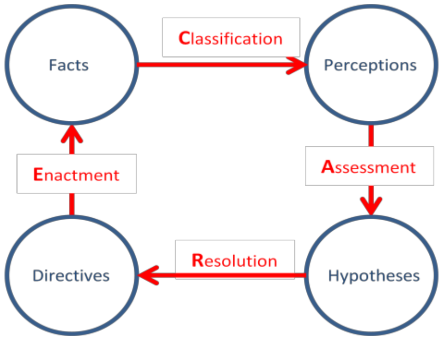

Figure 1: The KIDS CARE-loop

When we look at the CARE-loop from the perspective of processing we see the **CARE** (**C**lassification, **A**ssessment, **R**esolution, and **E**nactment) loop consisting of four distinct categories of knowledge acting on and producing specific categories of data. If we look at the same loop from the data perspective we see **F**acts, **P**erceptions, **H**ypotheses, and **D**irectives representing four distinctive types of data. The CARE-loop represents a normalized workflow; Facts, Perceptions, Hypotheses, and Directives of a CARE-loop instance represents a **situation**.

Data describes things that are typically stored in modern files or databases without any distinction. Formal knowledge is stored in articles, books, application code, workflow, case management systems, or decision support systems without any distinction about which knowledge is best used at what time and how everything interacts. To make up for the deficiencies, the CARE-loop provides a much needed structure of interaction between data, knowledge, and processes management in applications with the goal of providing a framework for applications dealing with situation awareness.

2. RELATED WORK

Boyd developed a model to deal with situation awareness by proposing the OODA loop (Observe, Orient, Decide, and Act) [2] [15]. The OODA loop was originally applied to military operations. Boyd conjectured that any operational system faced with rapidly changing and incomplete knowledge of the reality can thrive by rapid iteration using the OODA loop by continuously interacting with the environment to assess and adapt to the changes. Our proposed KIDS CARE-loop is fundamentally an enhanced computerized version of the OODA loop.

Paper [13] by Horvitz and Mitchell presented the need for transforming large data (facts) into insights and decisions support for evidence-based decision making process. Paper [21], motivated by Barwise, Perry, and Devlin, defined the situation theory ontology that materialized a situation as an object like any other physical or conceptual objects. In KIDS model [4], a situation is a quadruple of fact, perception, hypothesis, and directive objects. Paper [6] articulates a Data-Action-Model loop that has some conceptual similarity to the CARE-loop of KIDS.

3. THE KIDS MODEL

KIDS provides a model to support situation awareness by structuring applications according to the CARE-loop (see figure 1) using three core technologies: Data, knowledge, and process managament. The management of data is a relatively mature technology, which is currently going through a rapid evolution. KIDS contributes to this evolution by defining and managing four distinct categories of data:

Facts are what we measure in the world around us. The number, rate, and quantity of facts make it in many environments impossible for the human cognitive system to deal with these measurements directly. Therefore, we have to compact these facts in an ever increasing way for human consumptions.

Perceptions are compact, temporal, and qualitative representations of facts. They are optimized for use by the human cognitive system; they represent the most important characteristics of evolving situations, with high focus on exceptions. Perceptions depend on the perspective of the user.

Hypotheses are descriptions of possible root causes explaining the facts and the perceptions leading to the directives.

Directives describe what needs to be done to react to a specific set of facts, perceptions, and hypotheses. Directives specify action plans often in the form of workflows or processes. Obviously, the directive will most likely influence the evolving facts and also determine which facts should be captured in the future.

Managing these data requires a wide variety of data types/structures, extensibility, declarative access across data types/structures, time travel, flexibility in evolving data structures (support for well structured data as well as data first/structure later or never), OLTP, analytic, and so forth. There is also a need for extended functionality such as (fine grain) security, and provenance. Last but not the least important operational characteristics are disaster recovery, high availability, reliability, scalability, performance, and rapid development tools. Obviously, the management of data requires such a broad functional and operational support that is only available in mature and widely used databases. The collection and management of facts does not require the classical transaction model and limited loss of data may be acceptable; provenance requirements will drive these specifications. Mature databases need to optimize the management of facts and provide support with significantly reduced resource consumption. This can be achieved with well understood technologies; unlimited scalability is very achievable as well.

Knowledge is divided into four categories as well: Classification, Assessment, Resolution, and Enactment. These categories are based on the mode of reasoning that is required to process each category of data. A substantial subset of each category of knowledge can be automated by suitable computation models.

The **Classification** knowledge - transforming facts into perceptions - is primarily represented by **deductive** reasoning. Some Classification knowledge that produces prediction or norm may involve inductive reasoning as well. The computation model for classification includes Support Vector Machines [1], naïve-Bayes classifier [22], Multivariate State Estimation Technique [5], Clustering, Association Rules, Decision Trees, Cognitive computing, etc.

The **Assessment** knowledge - transforming perceptions into hypotheses - is typically implemented by **abductive** reasoning that derives the *Hypotheses* from *Perceptions*. The computation model for assessment includes Bayesian Belief Network [22] and Least-Squares Optimization or Regression of solutions for inverse problem [9].

The **Resolution** knowledge - transforming hypotheses into directives - involves **making decisions** under the uncertainty of outcomes by considering the relative merit of the different outcomes and the associated payoffs/costs. The computation model for resolution includes Bayesian Belief Network extended with decision nodes and payoff/cost nodes, known as Influence Diagrams [14], Dempster-Shafer theory [7][25], Decision Trees, and Prognosis of Remaining Useful Life [5].

The **Enactment** knowledge - transforming directives into actions (and new facts) - involves **control structures** encoded in scripts, plans, schedules, BPEL workflows, and business processes in BPMN. The actions often include capturing of new facts.

Knowledge will be applied in the proper sequence as specified by the CARE-loop. In some cases not all steps of the CARE-loop need to be executed. Knowledge - including each version of it - has to be stored in databases to take advantage of the state management with declarative access and manipulation. Knowledge has to be applicable ad-hoc and in real time. An important ad-hoc use case is the ability to revisit data (especially facts and perception) with new knowledge to find out what has been missed and has been overrated.

Additionally, each category of formal knowledge is complemented by tacit knowledge. Applications may also require social networking services where we can profile the tacit knowledge [11] and social preferences of the actors in the system. This allows identifying the most qualified individuals or teams for a task by adjusting the tacit knowledge profiles based on recent activities to ensure that profiles are as up-to-date as possible.

An important requirement of applications is the ability for continuous improvements. Methods to enable continuous improvements are:

- Improve the rules, queries, models, and procedures leveraging insights from users and domain experts.
- Re-run the existing models using additional data or new algorithms.

Knowledge can be exchanged between experts of a field; this exchange should be as formal as possible. Papers should be considered as the equivalent of Venn Diagrams helping to understand intuitively models or whatever formalism is used. Obviously, any new knowledge has to be carefully reviewed before it is generally used. KIDS allows the use of evolving and existing knowledge concurrently and is able to show both results. The application of knowledge is driven by the CARE-loop, which is a standardized process management structure.

4. KIDS USE CASES

In the following sections we discuss three use cases, which are in various states of development.

4.1 Cloud Operation

A basic postulate of cloud computing is the economy of scale by consolidation and pooling of the physical resources and providing virtually unlimited resources by dynamic resource management. The control system needs to manage the dynamic entity model to provide an accurate awareness of the system which changes due to frequent new software releases, patches for bug fixes, hardware upgrades, capacity scale out, and dynamic resource allocation. To conform to service level agreements (SLA), the cloud operations need continuous monitoring of vital signs and predictive diagnosis capability to detect and circumvent the SLA violations. A typical cloud operation has to monitor millions of hardware and software components of the data centers. The reactive fault detection and manual diagnosis techniques of traditional IT operations are labor intensive, require extensive domain expertise, often too little or too late in responsiveness, often resulting in disproportionate responses involving restart of large parts of the system instead of isolating and fixing the faulty components, and obviously cannot scale out for the cloud.

Cloud operation can only thrive by rapid iteration of the CARE-Loops to get inside the dynamics of seasonal cycles, load trends, load spikes, system response characteristics, transient glitches, gradual degradations, aging, and performance drifts of millions of components in the environment. The framework must automatically transform facts from different entities at different points of time into a representation of comprehendible and actionable perceptions for effective human or automated decision making. The framework must also perpetually evolve the entity model by discovering new entities, new relationships, and new knowledge that transforms vital measures about entities into appropriate representations [21]. This involves managing terabytes of data combined with read-time analytics in Big Data systems [4] and large scale management of millions of CARE-Loop instances using Oracle bi-temporal database, expression filters, registered queries, and orchestration engine to integrate many inference engines.

The KIDS model is a viable framework to enable the information fusion and situation awareness at the level of the complexity and heterogeneity needed to support the effective sequence of facts, perceptions, hypotheses, and directives in CARE-Loops. For example, a classification model predicts a load surge in the next two months based on an annual seasonal trend model learned from abduction. An assessment model hypothesizes that the CPU and memory usage will exceed the capacity during the seasonal peak. It issues a directive to scale out the number of virtual machines for the next two months. The system response to such a change would be observable in the new facts.

4.2 Software & Hardware Product Support

SW and HW support applications facilitate collaborative and iterative problem solving activities involving product support and customer's IT personnel. The objective of such activities is to minimize system unavailability by minimizing the time required to either, find and apply existing knowledge (tacit, explicit, or automated) to resolve known issues, or to discover remedy for new issues. And therefore, justifies the necessity for maximizing the automation of diagnosing and fixing known issues whenever economically possible, in order to free support and customer personnel to focus on newly emerging issues that require a great deal of collective human experience and intelligence. To achieve economical automation, it is essential to ensure that data and knowledge about product issues, encountered throughout the product lifecycle (in bug database, support tickets), are captured with precise articulation and provenance. This includes consistent terminology with accurate definitions, accurate and validated causality relationships, system configuration, and personnel's contributions. For example, computing a perception about invalid objects in the data dictionary with potential states (hi, lo, none) implies that we need to define the source where the fact (a query, a view, a diagnostic tool). A hypothesis such as database migration failure could be caused by invalid objects. And the resolution for such hypothesis could be an action plan that calls for recompilation the dictionary and run a query to evaluate the state of the dictionary after such recompilation, which will generate a new set of facts that will help determine whether our resolution worked or not. Such articulation and provenance enables accurate statistics for the likely recurrence of an issue and the degree of complexity in recognizing and resolving the issue automatically or semi-automatically. Next, it is essential to standardize the process for diagnosing issues by leveraging data collected for provenance. For example, diagnosing ORA-4031 Error (an oracle database memory allocation error), are most effectively done using facts from alert log and ORA-4031 trace. Such standardization aims at establishing standardized data collection, standardized parsing and interpretation of such data, standardized diagnoses, and standardized remediation methods as well as standardization of the entire process of issue resolution.

4.3 Patient Care

In patient care facts are representing various measurements and images such as vitals, blood tests, and ultra sound. Classification is used to express perceptions of concerns, such as a high and increasing risk for an impending heart attack. The assessment identifies the root cause such as fibrillations, which will be used as the working hypothesis. The resolution determines the directives: what medicine to take or procedure to perform or, if in doubt, what other facts need to be captured how often and when. Finally, the directives are acted upon, and new facts are captured to follow the progress of the patient; e.g., the perception will be re-computed to provide an updated high level view and the hypothesis will be re-evaluated. The hypothesis may entail prediction of expected future values. In this case the doctor can ask KIDS to inform him/her if the evolution of the patient is not as expected; the context determines the exact meaning of this sentence. The medical background of different doctors can lead to different – even conflicting – perceptions; this is not a problem for KIDS [10] [12].

Any new knowledge can be adapted immediately and not only be used to treat existing and future patients but also be used to review the history of former patients. Significant deviations in the perception and hypothesis based on new knowledge will be brought to the attention of the doctors [23].

5. ACTIVITIES DRIVING THE EVOLUTION OF KIDS

Within Oracle the KIDS model is actively used for the management of cloud services and customer care. In both cases the use of the KIDS model significantly helped the evolution of these projects. Other groups are in the process of evaluating KIDS.

A research project with IIT (Illinois Institute of Technology) is focused on providing unambiguous provenance in bi-temporal environments. Provenance allows the use of qualitative languages for perception without losing the precision that is inherent in facts [19][20]. Another research project with the University of Bonn (Germany) is focused on a detailed architecture for perceptions, with flight supervision as the primary use case. This use case requires significant advances in temporal spatial technology [18]. An additional research project with the University of Buffalo is focused on estimating the uncertainty of the results after applying knowledge to data. This uncertainty can among others be the result of imprecise, insufficient, or decaying data as well as imprecise knowledge [17].

While the above projects are funded by Oracle, there are other informal collaborations with universities without funding from Oracle; e.g. with Ken Baclawski from the Northeastern University [21].

6. OPPORTUNITIES & CHALLENGES

A very important value of databases is that they provide a declarative data management abstraction to manage persistent states. Furthermore, the success of databases is due to the underlying set-based abstract algebra. However, databases currently provide declarative abstraction for data only; they do not provide declarative abstraction to manage knowledge, processes, nor the interactions among data, knowledge and processes in the KIDS CARE-loops. Instead, these tasks have been procedurally coded in every application that needs KIDS semantics. We think it is time for databases to extend its declarative state management service to provide KIDS abstraction. There are a number of opportunities and challenges.

Theoretical Foundation: The set-based relational model is the theoretical corner stone for databases. To support KIDS semantics, however, we need set algebra that can embrace and reason about transformations among set elements and discover morphisms among transformations. Investigation of the KIDS abstraction using category theory [8] can lead to a theoretical foundation of KIDS and new principles of data, knowledge, and process management systems.

Declarative Language Support: Based on a theoretical foundation, we can extend the declarative languages such as DDL, DML and query component of SQL so that they can manage the facts, knowledge, processes, and CARE-loops. For example, a DDL extension can allow the applications to manage the life cycles of facts, knowledge, processes in a repository. A DML extension can support loading instances of KIDS elements into a repository, establishing the links among the KIDS elements in the CARE-loops, and navigating the CARE-loops stored in the repository, all with full integrity checks. It is essential to support automated CARE-loop execution. Finally, a comprehensive query language extension is needed to query not only facts, but also knowledge, processes, CARE-loops, and their interactions.

Paper [24] explored the database extensibility mechanism to accomplish flexible schema data management. We will continue to explore the database extensibility framework, which provides an engineering foundation, to naturally evolve databases to support KIDS abstraction. Certainly, designing a proper indexing and storage structure to provide efficient access of knowledge and process will require a lot of innovative techniques.

Time Dimension Support: KIDS model has both implicit and explicit dependency on tracking the valid time and transaction time dimensions for data and knowledge; KIDS semantics is based on the valid times of the data and knowledge when transforming them through the CARE-loops. The bi-temporal and provenance support of modern databases provides a valuable direction to understand data management in time. However, they need to be further enhanced to support the notion of multi-version of data with branching so as to support "what if" KIDS analysis. The goal is to eventually make the KIDS system behave like a time machine with a notion of the parallel universe of KIDS leveraging the idea of data version and branching.

Performance & Scalability: A DBMS supporting KIDS is expected to support the rapid growth of the number of KIDS instances, in particular, the growth of fact. While some of the facts can be discarded after a short time period, others may need to be kept and made easily accessible for an extended period of time. This requires a highly optimized data service providing data flexibility, durability, exactly once semantic, security, compression, compaction, time travel, provenance, along with extreme performance and scalability.

Practical KIDS Migration Strategy with Social Networking: Few applications have the luxury to start from scratch; therefore, a KIDS migration strategy is important. One idea is to keep the mature legacy applications running to support the existing functionality while creating a 'shadow' application based on the KIDS model. The shadow application can crawl the information of existing systems to automatically categorize the data, knowledge, process, and CARE-loop interactions so that application users can gradually gain the benefits of the KIDS model through the 'shadow' applications. Such a shadow KIDS application can be tuned based on the preferences of the application users by integrating with social networking services [3]. Like the Google Internet index, the shadow KIDS application provides the KIDS index for the applications data to let the community to search for matching problems and contribute solutions. Such a KIDS index can represent a situation aware service.

7. CONCLUSIONS

KIDS provides a framework to organize and support the applications to deal with situation awareness. KIDS integrates three technologies: data, knowledge, and process management. It does so by providing a high level model that normalizes the use of these technologies using the CARE-loop. By doing so, KIDS allows applications designers to develop applications with a new methodology leading to applications that are well structured and can evolve in real time. The evolution can be achieved through collaboration among many developers working on different applications with some overlap.

KIDS offers an incentive for the research communities of the three core technologies to work more closely to better leverage each other's work.

8. ACKNOWLEDGMENTS

The authors extend their thanks to colleagues who helped with many constructive suggestions and critics. Especially, we thank Andrew Mendelsohn, Rafiul Ahad, and Vikas Arora for encouraging and supporting this work. We like to thank Ken Baclawski, Andreas Behrend, Boris Glavic, Ying Hu, and Oliver Kennedy for their many contributions to the refinement of the KIDS model.

9. REFERENCES

[1] Boser, B.E., Guyon, I.M., Vapnik, V.N., 1992, *A training algorithm for optimal margin classifiers*. Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, ACM Press.

[2] Boyd, J.R., 1976. *Destruction and Creation. U.S.* Army Command and General Staff College.

[3] Chan, E.S., Behrend, A., Gawlick, D., Ghoneimy, A., Liu, Z.H., 2012, *Towards a Synergistic Model for Managing Data, Knowledge, Processes, and Social Interaction, SDPS-2012*, Society for Design and Process Science.

[4] Chan E.S., Gawlick D., Ghoneimy A., and Liu Z.H., "*Situation Aware Computing for Big Data*," SemBIoT 2014

[5] Cheng, S., Pecht, M., 2007, *Multivariate State Estimation Technique for Remaining Useful Life Prediction of Electronic Products*, Association for the Advancement of Artificial Intelligence.

[6] Crankshaw, D., Bailis, P., Gonzalez, J., Li, H., Zhang, Z., Franklin, M., Ghodsi, A., Jordan, Mi: *The Missing Piece in Complex Analysis: Low Latency, Scalable Model Management and Serving with VELOX,* CIDR 2015.

[7] Dempster, A.P., 1968, A generalization of Bayesian Inference. *Journal of the Royal Statistical Society*.

[8] D.I. Spivak: "*Category Theory for the Scientists*," ISBN-13: 978-0262028134

[9] Fletcher, R., 1970, *Generalized Inverses for Nonlinear Equations and Optimization*. *Numerical Methods for Non-Linear Algebraic Equations*. Gordon and Breach, London.

[10] Gawlick, D., Ghoneimy, A., Liu, Z.H., 2011, *How to Build a Modern Patient Care Application*. HEALTHINF.

[11] Gilmour D.L, et al. 2003, *Automatic Management of Terms in a User Profile in a Knowledge Management System*. United States Patent 6,640,229.

[12] Guerra, D., Gawlick, U., Bizarro, P., Gawlick, D., 2011, *An Integrated Data Management Approach to Manage Health Care Data*. BTW 2011.

[13] Horvitz, E., Mitchell, T., 2010. *From Data to knowledge to Action: A Global Enabler for the 21st Century*. *Data Analytic Series*, Computing Community Consortium.

[14] Howard, R.A., Matheson, J.E., 1984, *Influence Diagrams*. *Readings on the Principles and Applications of Decision Analysis*, v.2. Strategic Decisions Group, Menlo Park, CA.

[15] http://en.wikipedia.org/wiki/OODA_loop

[16] http://en.wikipedia.org/wiki/Situation_awareness

[17] http://mjolnir.cse.buffalo.edu/

[18] https://www3.uni-bonn.de/idb/research/states

[19] http://www.cs.iit.edu/~dbgroup/research/gprom.php

[20] http://www.cs.iit.edu/~dbgroup/research/oracletprov.php

[21] Kokar, M.M., Matheus, C.J., Baclawski, K., (2009), *Ontology-based situation awareness*, Journal Information Fusion, Vol 10, Issue 1.

[22] Koller, D., Friedman, N., 2009, *Probabilistic Graphical Models: Principles and Techniques*, The MIT Press, Cambridge, Massachusetts.

[23] Liu, Z.H., Behrend, A., Chan, E., Gawlick, D., Ghoneimy A., *KIDS - A Model for Developing Evolutionary Database Applications*. DATA 2012: 129-134.

[24] Liu Z.H., Gawlick, D., *Management of Flexible Schema Data in RDBMSs*, CIDR 2015

[25] Shafer, G., 1976, *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, NJ.