

Managing Complex Information in Reactive Applications Using an Active Temporal XML Database Approach

Essam Mansour, Kudakwashe Dube, Bing Wu

Abstract

It is every organization's desire to incorporate best practice into its enterprise. This incorporation gives rise to the need to maintain information that could be viewed as complex. Managing this complex information poses a major challenge to the area of information management. This paper focuses on applications that take the form of reactive applications. The paper presents a generic approach and framework, called CIM, for managing such complex information. The approach focuses on the management of the complex information in a systematic and unified way such that the complex information could still be specified, executed, manipulated and queried within the same underlying framework. The complex information is based on a model that incorporates static and dynamic aspects of the application domain. A high level declarative language, called AIM, is utilized with this model to support the management framework. Active database, a temporal mechanism, and XML technologies are combined together as an implementation method for CIM. AIMS, prototype system, is being implemented as a complex information management system.

Introduction

Domains, such as patient care practice and securities trading order management, involve activities that require constant monitoring of some vital signals or essential happenings within the domain. Best practice exists in several forms, such as recommendations, expertise, and methodologies. Incorporating the application best practice into these activities produces Complex Information (CI), which includes dynamic and static aspects as follows:

- the reactive behaviour derived from the best practice with respect to the user's preferences or situations;
- the monitored information or data items;
- description for the complex information; and
- the evolution history of it.

Usually, the complex information exists in application domains as distinct entity that demands a computerized management support. Consider as examples, the customer order in securities trading and the medical patient plan in health care.

A major challenge in reactive applications is to provide a unified framework in which the complex information can be managed as one body in such a way that the formalisation, specification, execution, manipulation and query dimensions are explicitly provided as first class elements under one roof within the framework.

We are focusing on managing the Complex Information of the reactive applications, which have the ability to process events of interest to domain entities, and can respond to changing situations by issuing alerts, reminders, requests, and/or observations to users.

Related Work

The related work to this research could be classified into two categories:

A) Approaches

the approaches adopted in incorporating best practice into application activities are such as a) the active database and business rules[1]; b) event-based workflow [2]; c) AI and decision making. At a general level, existing works 1) provide little or no support for comprehensive unified management of the information and processes derived from best practice in domain applications. 2) separate between the dynamic and static aspects of the Complex Information

B) Technologies Supporting Complex Information

Several research efforts have utilized a combination of an event-driven mechanism with XML technologies [3] and /or temporal database [4] as the main technologies to support the Complex Information Management.

CIM: A Generic Approach and Framework to Computerizing the Complex Information

The CIM approach emphasizes the management of the Complex Information (CI) as one distinct entity. The approach's unique features include:

- managing the dynamic and static aspects of the CI as one object, not as a separated objects;
- a high level declarative language, AIM, for specifying, manipulating, and querying the CI;
- utilizing an event-driven mechanism incorporated with XML technologies, and temporal DB as an implementation method;
- support for the decentralized management of the CI.

The CIM framework defines unified plane-based support structure for managing the complex information as one distinct entity. As illustrated in Figure 1, the CIM framework consists of the three planes: specification plane; instantiation and customization plane; and management plane with the DBMS and XML technologies as the integrating factor and communication among the three planes.

Figure 1

The Complex Information Management (CIM) framework

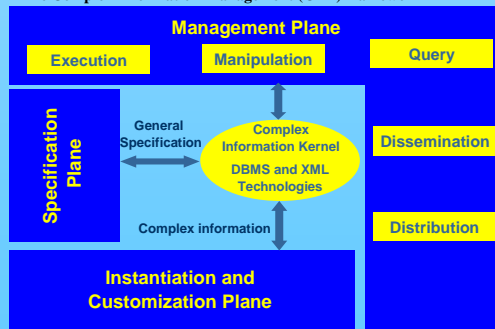
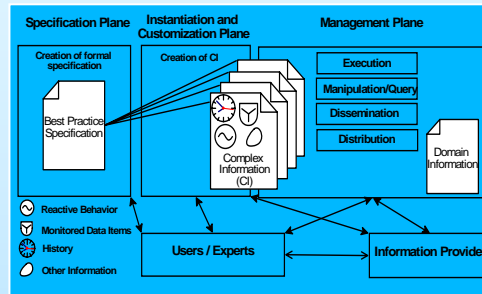


Figure 2

The Process Management of the Complex Information



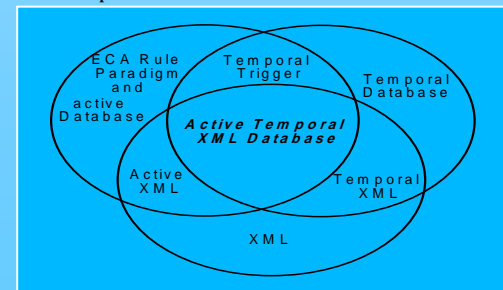
The Process Management of the Complex Information, as illustrated in Figure 2, spans the three planes of the framework. The best practice of an application domain, which represents the general form of the CI reactive behaviour, is formally specified. The domain experts furnish this process with the best practice. A formal specification language is required to support the specification process. With respect to specific domain scenario, the best practice specification is customized and instantiated to produce the CI. The user and the information provider participate in the instantiation and customization process by providing her/his interesting data items and information about these data items respectively. This process required a formal model for the CI that makes it easy for the complex information to be executed, manipulated, queried, disseminated, and distributed. These management aspects are provided in the management plane.

AIM Language facilitates the management of the Complex Information though its process management. AIM language consists of specification component (AIM-SL) and query/manipulation component (AIM-QL). AIM-QL provides advanced query capabilities, such as reply the evolution history of the Complex Information.

The Active Temporal XML approach utilizes the combined application of the Event-Condition-Action (ECA) rule paradigm, a temporal mechanism, advanced DBMS features and XML technologies to support the CIM approach and framework, and implement the AIM language. A map of the domains involved in this approach is shown in Figure 3.

Figure 3

Active Temporal XML Database

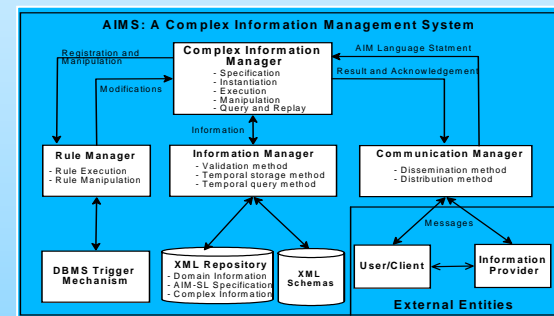


AIMS: A Complex Information Management System

AIMS utilizes the AIM language to manage the complex information according to the CIM framework. A conceptual architecture of AIMS is presented in Figure 4. The Complex Information (CI) manager provides the high level management for the complex information. CI manager provides support to specify the application best practice, generate, execute, manipulate, and query CI documents. The rule manager provides for the execution of CI rules, their manipulation and extension of the active mechanism to support temporal rules. The information manager extends an XML database system to provide temporal support and utilizes an XML DBMS to validate and store the AIM-SL specification and the CI documents. The communication manager supports the remote access and distributed management to the AIM-SL specification and CI documents. AIMS is being implemented by using DB2, java, and XML technologies, such as XQuery and Web services.

Figure 4

A Conceptual Architecture of AIMS



Conclusion

This paper has presented the Complex Information Management (CIM) framework and approach, which consists of three planes for specifying, customizing and managing the complex information. The AIM language that is developed to support the functionalities of the CIM framework planes has been briefly outlined. The paper has also presented a conceptual architecture for the AIMS system, which utilizes the AIM language in managing the complex information by following the CIM framework.

References

- Widom, Jennifer and Stefano Ceri (1996). Active Database Systems: Triggers and Rules For Advanced Database Processing, Morgan Kaufmann.
- Hong, Hyeon-Seok, Bong-Seok Lee, et al. A Web-Based Transactional Workflow Monitoring System. In Proceedings of WISE'2000: IEEE Computer Society.
- Bonifati, Angela, Daniele Braga, et al. (2002). Active XQuery. Proceedings of the 19th International Conference on Data Engineering ICDE.
- Gao, Dengfeng and Richard Snodgrass. Temporal Slicing in the Evaluation of XML Queries. in VLDB. 2003

For Further Information

Please contact essam.mansour@comp.dit.ie. More information on Essam's PhD project and his research can be obtained at <http://www.comp.dit.ie/emansour>.