AIM: An XML-Based ECA Rule for Supporting a Framework for Managing Complex Information

Presented By
Essam Mansour
Over View

- Introduction
- AIM language
- AIM Specification Component
- The Complex Information Model in AIM
- AIM Query Component
- Summary
Introduction

- My Project is the second stage of an on-going research
- My Project Objectives:
  - Generalizing and enhancing the concepts and management framework done in the first stage;
  - Developing a high-level language for facilitating the management of the generalized framework;
  - Developing intermediate models for implementing the language using the available technologies, such as XML and DBMS;
  - Developing a proof-of-concept system;
  - Evaluating the work done in the second stage
Introduction

First Stage
- Clinical Guidelines
  - Medical Patient Plan
    - Healthcare Record
- Formalization And Modelling Language
- Main Research Focus
- Execution and Manipulation

Second Stage
- Stock Exchange Regulation
  - Customer Order
    - Stock Items
- Best Practice
- Complex Information
  - Enriching these Functionalities
    - Modelling Language
    - Maintenance Support
  - Enriching these Functionalities
- Domain Information

Examples
Research Focus
Not the Research Focus

Generalizing The Work Done

Examples:
- Clinical Guidelines
- Stock Exchange Regulation
- Best Practice
- Complex Information

Research Focus:
- Medical Patient Plan
- Customer Order
- Healthcare Record
- Stock Items

Not the Research Focus:
- Formalization And Modelling Language
- Main Research Focus
- Execution and Manipulation
Introduction

- The Complex Information is produced by incorporating the best practices into the daily management.
- The medical plans and customer orders are examples for Complex Information produced by incorporating clinical guidelines and stock regulation into the disease and order management, respectively.
- Managing Complex Information
- Integrating the Complex Information Management into DBMSs, which are utilized to manage the domain information
AIM Language

- Supporting the SEM Framework at three plans:
- AIM is an XML-Based language
- AIM Components

AIM Components:
- AIMSL
  - Support the Specification Process
- CI Model
  - Support instantiation and maintenance processes
- AIMQL
  - Support Manipulation and Information Retrieval
AIM Specification Component (AIMSL)

- The main concepts in AIMSL model are:
  - Protocol
  - Schedule
  - Rule
- Generic and Customization.
AIM Specification Component (AIMSL)

- AIMSL utilizes ECA Rules to represents the best practice.
- In AIMSL, ECA Rules are distinguished by:
  - The event, condition, and action are defined using domain application terms, ex:
    Rule 1: on two days after patient admission, order the blood test.
  - Temporal Events:
    - Absolute time or Relative time event
      on June 1, 2008
      on day 2 of patient admission
    - Once-off or Repetitive with before or after support
      on day 2 of patient admission
      every 10 hours before the operation time
The Complex Information Model in AIM

Complex Information Life-Cycle in AIM
The Complex Information Model in AIM

Complex Information Schema in AIM

Diagram showing the complex information model in AIM with nodes and relationships.
The Complex Information Model in AIM

Rule MAS1: ON day 2 of the patient admission, DO order the test albumin creatine ratio (ACR).

Rule MAS2: ON receiving the result of test ACR IF the ACR result is greater than 25 DO Add rule 3 and 4 to order ACR test at day number 6 and 38.

What is CI in Day 4?

Replay CI from Day 3 to 5.
AIM Query Component (AIMQL)

- **Requirements**
  - Move complexity from user/application code to high level declarative language
  - Changes Propagation
  - Declarativity
  - XQuery –based language
  - Convenient for human to read and write

- **XQuery Extensions:**
  - Manipulation Operations: add, remove, modify, activate, deactivate, terminate, and Fire.
  - Temporal Query Support for the replay functionality
AIM Query Component (AIMQL)

Natural language:
Replay the plans of category no CAT, which was working through out the past Y days.

AIMQL

REPLAY Complex Information CI
SHOW When, How, Why OF CI
Where CI.cast("day") >= Y
and CI.meets(NOW)
and CI[@catID=CAT]
Summary

- Our Research focuses on providing a comprehensive management for the complex information
- K-CAMP provides a generic approach and framework for Complex Information Management
- AIM language consists of:
  - AIM Specification Component
  - The Complex Information Model in AIM
  - AIM Query Component
Thank You