An Architectural Framework for an Adverse Drug Event Surveillance System
Young Ho Lee¹, Ji Eun Choi², Ga Eul Cha², Kyung Jin Lee³, Hee Joung Hwang¹, Byung-Mun Lee¹, Rae Woong Park, MD³
(1)Gachon University of Medicine and Science, Inchon
(2)Regional Clinical Trial Center, Ajou University Hospital, Suwon
(3)Department of Medical Informatics, School of Medicine, Ajou University, Suwon, Republic of Korea

ABSTRACT
Much effort has focused on detecting as many adverse drug events (ADEs) as possible, as soon possible. An ADE surveillance system (ADESS) is a computerized surveillance system that detects ADEs automatically, by analyzing medication orders, laboratory results, and medical records. We propose a new ADESS architectural framework using the object-oriented component-based development (OOCBD) methodology to extract and analyze ADEs automatically, with a minimal server-side workload.

The architectural model was expressed using different views to represent various aspects of the ADESS. Although there are many ways to represent the software architecture, such as 4+1 View, the OOCBD method was used to develop this model.

The architectural model consists of business architecture, technical architecture (TA), data architecture, application architecture, and security architecture. The business architecture was expressed using a structured use-case model, which was defined in the requirement analysis process. The TA constructs the technical environment used to execute individual attributes defined in the business architecture. The TA also constitutes the execution environment for the developed ADE components. The data architecture provides a view to realize a reference knowledge database for ADE detection. The application architecture extracts possible ADE signal data from a hospital information system (HIS) and analyzes it to detect ADEs. The security architecture protects the privacy and confidentiality of patient information in the ADESS.

Most existing ADESSs use a ‘triggering’ function for the database management system (DBMS) as an active data-monitoring method to extract ADEs from the database. This triggering function of the DBMS is a simple way to detect various events in a database. Nevertheless, the triggering method increases the DBMS workload and may sometimes slow the working of the HIS when an abrupt increase in the number of triggering actions occurs.

The proposed ADE architectural framework uses an information warehouse to solve the workload increase in the event-driven triggering method. The proposed ADE information warehouse framework consists of an ADE monitoring system, ADE inference engine, ADE result delivery system, and ADE feedback. Based on the proposed ADE architectural framework, we are now developing an ADESS using the Extraction, Transformation and Loading (ETL) technology, which enables the system to infer and predict data events.