

데이터베이스를 이용한 객체지향 유출해석

Object-Oriented Runoff Analysis Using DataBase

김 상 민* · 박 승 우(서울대)

Kim, Sang Min · Park, Seung Woo

Abstract

This paper presents a framework for developing an object-oriented system for runoff analysis. The objects include rainfall, meteorologic, watershed, reservoir, stream, DB management, and GUI. Data and method of each object were analyzed and defined. The database for runoff analysis were designed and DBMS MS-Access was chosen. The system design features and implementation are described, and an graphic user interface for flood runoff is presented

I. 서론

농업생산분야의 생산력 제고와 구조개선을 위해 '농어촌용수이용 합리화계획 자료정보 데이터베이스 구축 연구' 등 농업생산기반정비사업에 관련된 정보들을 처리할 수 있는 연구들이 수행되고 있으며, 관련된 데이터베이스가 광범위하게 구축되고 있다. 기존에 구축된 데이터베이스를 활용하여 수문분석에 이용한다면 보다 용이하게 수문관련 자료를 취득할 수 있을 것으로 기대되며, 이를 위해서는 데이터베이스와 연동되는 수문분석 프로그램의 개발이 요구된다.

객체지향기법은 객체라는 개념과 자료의 추상화를 이용하여 실제의 세계와 유사하게 프로그램을 구현할 수 있도록 하며, 복잡한 시스템의 초기 개발을 용이하게 할 뿐만 아니라 이렇게 개발한 소프트웨어를 차후에 쉽게 재사용할 수 있도록 하여 전체 시스템 개발 비용을 크게 절감할 수 있다. 객체지향기법을 통해 수문분석 시스템을 구축할 경우 데이터베이스의 변경이나 수문분석 방법의 추가나 변경에 따른 시스템의 수정이 용이한 장점을 지니게 된다.

따라서, 본 연구에서는 객체지향기법을 이용해 유출해석을 위한 객체를 분석하여 정의하고, 객체에 포함된 정보를 데이터베이스를 이용하여 제어하는 유출해석 방법을 제시하는 데 있다.

II. 연구방법

2.1 객체지향기법

객체지향기법은 소프트웨어의 문제영역 또는 대상을 여러 개의 객체로 구분하여 각 객체를 자료(data)와 메소드(method)로 정의한 후, 객체별로 프로그램을 개발하여 연결함으로써 각 객체에 의해 문제의 해답을 얻을 수 있도록 소프트웨어를 개발하는 기법이다.

객체지향기법에서는 실세계를 모사함에 있어 실존하는 대상들을 독립적인 객체로 보고 존재하

는 객체들의 공통된 속성을 분석하여 일반화 또는 추상화된 클래스를 만들어내며 이를 추상화(abstraction)라 한다. 모든 객체는 자료와 메소드를 갖게 되며 이러한 객체들은 그 유사성이나 상호 관계에 따라 보다 일반적인 슈퍼클래스(superclass)로 그룹화되고 반대로 서브클래스(subclass)로 확장되기도 한다. 이때 기존의 클래스에서 가지고 있던 자료와 메소드는 서브클래스에서 정의할 필요 없이 재사용이 가능하게 되며, 이것을 상속(inheritance)라고 한다. 이러한 상속의 개념은 소프트웨어의 재사용이 가능하며, 중복개발의 수고를 덜어준다. 각 객체는 입력의 형태에 따라 상이한 결과를 도출하게 되는데, 이를 다형성(polymorphism)이라 한다.

객체지향기법은 이러한 추상화, 상속, 다형성 등의 개념을 이용해 기존의 절차지향기법에서 발생하는 자료와 절차의 분리, 중복개발에서 오는 문제, 묵시적 이용순서에서 발생하는 문제, 기능 중심에서 발생하는 문제 등을 해결한다.

객체지향 분석/설계 단계는 객체의 인식과 정의, 클래스의 구성, 클래스들간의 관계 파악 및 클래스 계층 구조 구성, 재사용 가능한 클래스 라이브러리와 응용 프레임워크 제작 등의 단계를 거치게 된다.

농업유역에서의 유출해석을 위한 객체는 <표 1>에서와 같이 강우, 기상, 유역, 하천, 저수지, DB관리, GUI 객체 등으로 구성할 수 있으며 각각의 객체에 대한 자료와 메소드는 <표 1>에서 정의한 바와 같이 나타내었다.

<표 1> 유출해석을 위한 객체의 구성

객 체	구 분	내 용
강우객체	Data	기상대 명칭, 기상대번호, 일별 강우량, 폭우사상에 대한 시간별 강우량
	Method	과거 강우량 기록치 선택, 폭우사상에 대한 강우자료 불러오기, 폭우사상별 강우량 계산, 지속시간 계산, AMC 조건 계산
기상객체	Data	기상대 명칭, 기상대번호, 일평균기온,
	Method	과거 기상자료 선택, 기상자료 불러오기
유역객체	Data	유역명, 유역면적, 유역주하천의 유로연장, 유역의 표고차, 유역평균경사, 유역 평균 CN, 토지이용상태, 인접 기상대 명칭
	Method	유역자료 불러오기, SCS 방법을 이용한 홍수유출량 계산, 수정 Tank 방법을 이용한 일별 유출량 계산
저수지객체	Data	저수지명, 저수지 제고, 제당길이, 형태, 만수면적, 관개면적, 수위-내용적 곡선, 일별 관개량
	Method	저수지 자료 불러오기, 저류지시법을 이용한 저수지 홍수추적 관개량 계산
하천객체	Data	하천구간명, 하천길이, 하도유입량, 하도유출량, 양수량, 하천단면
	Method	하천 특성자료 불러오기, 블록추적법을 이용한 하도구간 홍수추적
DB관리객체	Data	DB 명칭, DB 형식, DB 드라이버, 사용자 ID, Password
	Method	DB 연결, Table 생성/수정/추가/삭제, 자료 불러오기
GUI객체	Data	메뉴 명칭/내용/크기, 패널 명칭/내용/크기 등
	Method	메뉴 만들기, 패널 만들기, 이벤트 처리 등

2.2 데이터베이스의 설계

데이터베이스는 관련있는 데이터의 모임으로, 데이터를 보관하고 사용자의 물음에 대답하는 시스템이라 할 수 있다. 데이터베이스에 저장된 데이터는 끊임없이 변화하므로 데이터베이스는 새로운 데이터를 저장하거나 기존의 데이터를 삭제, 변경시키는 작업을 저장된 데이터가 일관성을 유지할 수 있도록 수행해야 한다. 데이터베이스를 생성하고, 관리하며, 데이터로부터 사용자의 물음에 대한 대답을 추출하는 프로그램의 집합을 데이터베이스 관리시스템(Database Management System, DBMS)이라 한다. 데이터베이스 관리시스템의 주된 목표는 데이터를 저장하고 이 데이터로부터 유용한 정보를 얻어내기 위한 효율적이고 편리한 방법을 사용자에게 주는 데 있다.

데이터베이스를 설계할 때 어떤 용도로 사용될 지 모르는 상태에서 만들어질 수 있기 때문에 데이터베이스는 되도록 유연하고, 포괄적이며, 적응성이 있어야 한다.

유출해석을 위한 데이터베이스는 <표 1>에서 정의한 객체별로 구축될 수 있으며, 강우, 기상, 유역, 저수지, 하천 객체에 대한 데이터베이스를 개별적으로 구축하여 유출해석에 이용할 수 있다. 강우 데이터베이스의 경우 폭우사상과 일별강우를 각각 기상대별로 구축하고 유역 데이터베이스의 경우 유역별 자료를 하나의 테이블로 구축하였다. <그림 1>은 유역의 홍수유출해석을 위한 폭우사상과 유역 데이터베이스의 구축예를 보여주고 있다.

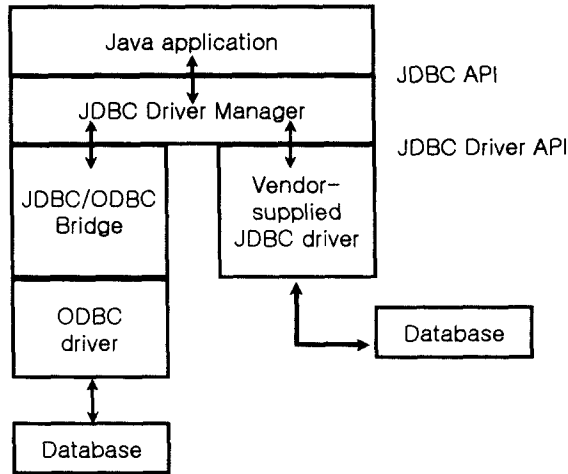
Station	River/Channel	Storm/Event	Rainfall Amount	AMC	Date	Time	Rainfall
미천		95-990715	120.3		99-07-15	오전 1:00:00	20
					99-07-15	오전 2:00:00	15
					99-07-15	오전 3:00:00	5
대주		95-990716	110.2		99-07-16	오전 1:00:00	10
					99-07-16	오전 2:00:00	15
미천		95-990818	75.1		99-08-18	오전 5:00:00	10
					99-08-18	오전 6:00:00	15
					99-08-18	오전 7:00:00	8

<그림 1> 유역 홍수유출해석을 위한 데이터베이스 구축예

Ⅲ. 유출해석 시스템의 개발

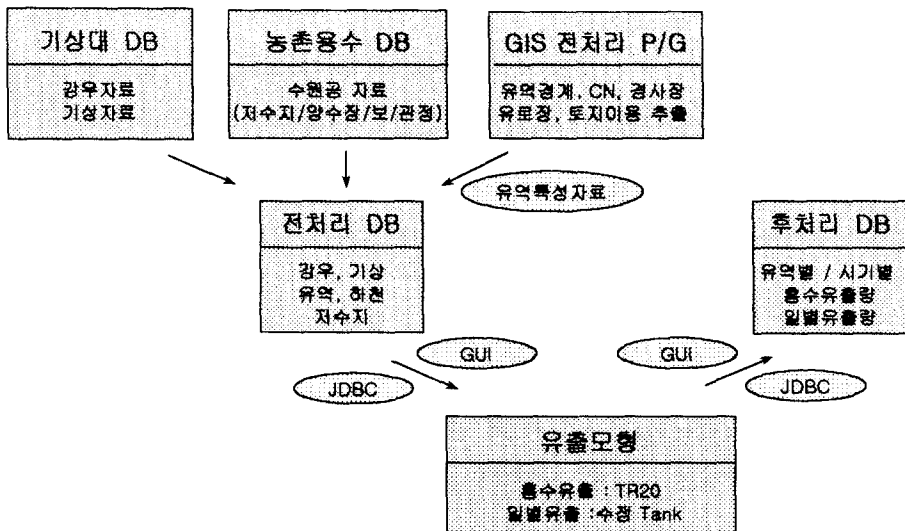
3.1 시스템의 구성

본 연구에서는 데이터베이스를 이용한 객체지향 유출해석 시스템의 개발을 위해 객체지향 언어인 Java™을 사용하였으며, 데이터베이스와의 연결을 위해 JDBC™(Java Database Connectivity)을 이용하였으며, 데이터베이스는 MS-Access를 이용하여 구축하였다. JDBC는 모든 관계형 데이터베이스들의 필수적인 인터페이스인 SQL(Standardized Query Language)에 대한 인터페이스다. <그림 2>는 JDBC에서 데이터베이스로의 통신 경로를 보여주고 있다.



<그림 2> JDBC에서 데이터베이스로의 통신 경로

농업유역의 유출해석을 위해 필요한 자료로 강우, 기상, 유역, 하천, 저수지 등이 있으며, 강우자료와 기상자료는 기상대 자료를 이용하고, 저수지 등의 수원공 자료는 농촌용수 데이터베이스 자료를 활용하며, 유역에 대한 자료는 GIS를 이용하여 사용자가 원하는 유역의 특성인자를 도출하여 데이터베이스를 구축할 수 있다. 유출해석의 편의를 도모하기 위해 사용자 편의 시스템(Graphic User Interface, GUI)을 통해 사용자가 원하는 유역의 필요한 자료를 쉽게 얻을 수 있도록 시스템을 구성하였다. 또한 유출해석을 통해 얻어진 자료는 유역별/시기별로 데이터베이스를 구축하여 활용할 수 있도록 구성하였다.

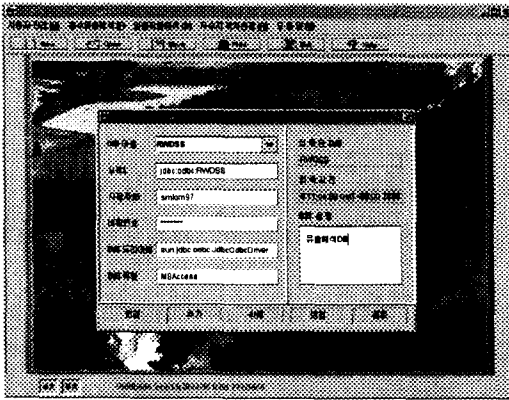


<그림 3> 유출해석 시스템 구조

3.2 시스템의 적용

유출해석을 위해 구성된 객체를 이용해 시스템을 개발하고, 구축된 데이터베이스를 연결하여 사용자 편의 시스템상에서 사용자가 원하는 유출정보를 얻을 수 있도록 하였다.

<그림 4>는 유출해석을 위해 데이터베이스와 연결하는 화면을 보여주고 있다. 데이터베이스가 위치한 주소와 해당 데이터베이스가 제공하는 드라이버를 설정해 데이터베이스를 사용할 수 있도록 하였다. <그림 5>는 홍수 유출해석을 위한 GUI를 보여주고 있다. 홍수 유출해석을 위한 GUI에서는 유역자료와 폭우 사상 자료를 사용자가 선택할 수 있도록 하였으며, 현재 자료가 구축되어 있는 유역과 폭우사상 자료를 화면에 보여지도록 하였다. 사용자가 유역을 선정하면 그 유역에 인접한 기상대의 자료로부터 이용가능한 폭우사상 자료를 선택하여 실행하면 폭우사상별 홍수유출량을 추정하도록 하였다. 추정된 유출량은 후처리 데이터베이스에 저장되고, 사용자 편의 시스템을 통해 그래픽으로 보여질 수 있도록 현재 개발중에 있다.



<그림 4> 데이터베이스 연결 화면



<그림 5> 홍수유출해석을 위한 GUI

IV. 요약 및 결론

본 연구에서는 데이터베이스를 이용한 객체지향 유출해석을 위해 필요한 데이터베이스를 설계하고 객체를 정의하였으며 객체지향 언어인 Java™을 이용하여 시스템을 개발하였으며, 사용자 편의 시스템을 제공해 사용자가 원하는 유출해석 결과를 쉽게 얻을 수 있도록 하였다.

본 연구의 결과를 요약하면 다음과 같다.

1. 객체지향기법에 대해 고찰하고 유출해석을 위한 객체를 정의하고 각 객체별 자료와 메소드를 구성하였다.
2. 유출해석을 위한 데이터베이스의 구축에 대해 고찰하고 MS-Access를 이용한 홍수유출을 위한 데이터베이스의 구축사례를 살펴보았다.
3. 유출해석을 위한 시스템의 개발환경과 JDBC를 이용한 데이터베이스의 통신 경로를 고찰하고 유출해석 시스템의 구조를 설계하였다.
4. 데이터베이스를 이용한 객체지향 유출해석을 위한 시스템을 구축하고 홍수유출에 대한 적용 사례를 보여주었다.

참고문헌

1. Booch, G., 1994. Object-Oriented Analysis and Design, Addison Wesley.
2. Belkhouche, B., I. Demtchouk, and L. J. Steinberg, 1999. Design of Object-Oriented Water Quality Software System, J. of Water Resources and Planning and Management, ASCE, 125(5). pp. 289-296.
3. Hamilton, G., R. Cattell, and M. Fisher, 1998. JDBC™ Database Access with Java™. Addison Wesley.
4. Ponce, V. M., 1989. Engineering Hydrology, Prentice Hall.