

객체지향 데이터베이스를 이용한 주식데이터 관리에 관한 연구

허순영* · 김형민*

A Study on the Management of Stock Data with an Object-Oriented Database Management System

Soon-Young Huh* · Hyung-Min Kim*

Abstract

Financial analysis of stock data usually involves extensive computation of large amount of time series data sets. To handle the large size of the data sets and complexity of the analyses, database management systems have been increasingly adopted for efficient management of stock data. Specifically, relational database management system is employed more widely due to its simplistic data management approach. However, the normalized two dimensional tables and the structured query language of the relational system turn out to be less effective than expected in accommodating time series stock data as well as the various computational operations.

This paper explores a new data management approach to stock data management on the basis of an object-oriented database management system (ODBMS), and proposes a data model supporting time series data storage and incorporating a set of financial analysis functions. In terms of financial stock data analysis, it specially focuses on a primitive set of operations such as variance of stock data. In accomplishing this, we first point out the problems of a relational approach to the management of stock data and show the strength of the ODBMS. We secondly propose an object model delineating the structural relationships among objects used in the stock data management and behavioral operations involved in the financial analysis. A prototype system is developed using a commercial ODBMS.

* 한국과학기술원 테크노 경영대학원

1. 서 론

주식과 관련된 데이터중 주식이가격에 대한 데이터는 그 양이 방대하며 이에 대한 분석을 위한 연산은 많은 계산량을 필요로 한다. 따라서 주식데이터에 대한 관리와 이의 분석에는 시스템의 이용이 필요하다. 특히, 주가데이터의 관리를 위해서는 데이터베이스 시스템의 이용이 필요하다. 주가데이터는 주식종목별로 주식이가격을 모아놓은 것으로 이는 데이터가 시간순으로 저장, 관리된다. 따라서 주가데이터는 시계열데이터(time-series data)의 성질을 갖는다.

현재 정보시스템을 구현함에 있어서 가장 널리 쓰이고 있는 데이터베이스 시스템은 관계형 데이터베이스 시스템이다. 그러나 관계형 데이터베이스 시스템은 주식데이터와 같은 시계열 데이터의 성질을 가지는 데이터를 관리하는데 있어서 약점을 가지고 있다고 언급된다. 이러한 가장 큰 이유로는 관계형 데이터베이스의 기본적 구조 때문인데, 관계형 데이터베이스는 그 구조가 2차원적인 테이블의 형태로 한 시점의 데이터들을 저장, 관리하는데 적합하도록 되어있기 때문이다[1, 4, 5, 10, 13]. Rozen은 채권에 대한 데이터베이스를 구축함에 있어서 관계형 데이터베이스를 이용할 때의 장단점에 대하여 설명하고 이러한 단점을 극복하기 위하여 객체지향 데이터베이스가 필요함을 설명하고 있으며[9], Dreyer는 시계열데이터 관리시스템이 가져야할 기능에 대하여 설명하고 이러한 기능을 시스템이 가지기 위해서 객체지향 개념의 이용이 필요함을 설명하고 있다[3]. 그리고 Segev는 시계열데이터의 성질을 가지는 재무데이터 모델의 구조와 성질에 대하여 설명하고 있다[12]. 이러한 기존의 연구들의 특징

은 주식 데이터베이스가 시계열데이터 관리기법을 도입하여야 하고 새로운 데이터베이스 시스템을 기반으로 사용하는 것이 더 바람직함을 설명한다. 그러나 기존의 연구들은 이러한 방향만을 제시해줄 뿐 구체적인 방법에 대한 제시는 아직까지 없는 실정이다.

본 논문에서는 주식데이터를 관리하는 시스템에 대한 요구사항을 정리하고 이러한 요구사항을 만족시키는데 있어서 관계형 데이터베이스가 가지는 약점을 설명하고 이에 대한 보다 효과적인 해결방안으로 객체지향 데이터베이스의 이용이 유리한 점을 가지고 있음을 제시하고자 한다. 그리고 객체지향 데이터베이스를 이용하여 주식데이터 관리시스템에 대한 요구사항을 만족시키기 위한 데이터 모델의 기본구조와 주식데이터의 조사를 위한 연산의 기본구조를 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 시계열 주식데이터의 특성에 대하여 살펴보고, 3장에서는 주식데이터 관리시스템에 대한 요구사항을 파악하고자 한다. 4장에서는 이러한 요구사항을 만족시키는데 있어서 관계형 데이터베이스가 가지는 약점과 객체지향 데이터베이스의 적용가능성에 대하여 살펴보고, 5장에서는 본 논문에서 제시하는 객체지향 데이터베이스를 이용한 주식데이터 관리시스템의 데이터 구조와 연산의 구조에 대하여 설명하고자 한다. 6장에서는 앞에서 논의한 바를 객체지향 데이터베이스인 ObjectStore를 이용하여 구현한 결과를 소개하며, 7장에서는 본 논문의 결론을 정리하고자 한다.

2. 시계열데이터의 특성

이 장에서는 일반적인 시계열데이터의 구조와 연산에 대하여 알아보고, 이러한 시계열데이터의 특성을 가지는 주식데이터의 특성에 대하여 설명하고자 한다.

2.1 시계열데이터의 구조와 연산

주식데이터들은 기본적으로 시간의 변화에 따른 각 종목의 주가의 변화를 나타내는 시계열데이터로 볼 수 있다. 그러므로 주식데이터를 다루기에 앞서 시계열데이터의 구조적 특성과 연산적 특성에 대해 먼저 알아보도록 한다.

시계열데이터의 구조를 정리하면, 시계열데이터들은 기본적으로 일정한 객체가 일정시점에 가지는 속성을 나타내며 따라서 시계열데이터는 객체의 이름, 시간, 속성 3가지 구성요소로 이루어진다고 할 수 있고, 이는 $\langle s, t, a \rangle$ 로 나타낼 수 있다. 여기서 s 는 객체의 이름, t 는 시간, a 는 객체의 속성을 가리킨다. 이 3가지 구성요소는 다시, $\langle s, (t, a)^* \rangle$ 로 나타낼 수 있다 [12]. (*는 반복을 의미함) 즉, 시계열데이터는 하나의 객체에 대하여 데이터가 시간순으로 배열됨으로써 하나의 객체 이름에 대하여 (t, a) 쌍이 반복적으로 그리고 시간순으로 나타나게 됨을 의미한다. 본 논문에서는 이를 시계열데이터(time sequence)라고 부르기로 한다. 주식에 대한 시계열데이터 즉, 주가데이터는 종목별로 시간의 흐름에 따라 여러종류의 가격을 가지게 되므로 $\langle - \rangle$ 가 둘이상인 경우를 의미한다고 할때, $\langle s, (t, \bar{a})^* \rangle$ 로 나타낼 수 있으며 이는 하나의 객체이름에 대하여 시간과 속성이 반복적으로 나타나며 각 시간과 속성의 쌍안에

는 한 시점에서 여러가지 속성들이 존재함을 나타낸다.

시계열데이터의 연산이란 시계열데이터 값들에 대한 일련의 조작을 하는 것으로 데이터베이스의 관점에서 보면 검색연산, 수정연산, 삽입연산, 삭제연산 등이 있다. 주식데이터의 주된 용도가 주어진 기간 또는 시점에서의 주식데이터 값에 대한 분석이므로 본 논문에서는 검색연산에 주된 관심을 가지며, 따라서 시계열데이터에서의 기본적인 검색연산에 대하여 설명한다.

검색연산이 적용되는 시계열데이터를 원천(source) 시계열데이터라고 하고, 검색연산의 결과로 생성되는 시계열데이터를 목적(target) 시계열데이터라고 하며, 시계열데이터에 적용될 수 있는 기본적인 검색연산들은 다음과 같은 4가지로 분류할 수 있다[16]. 첫째, 선택연산(selection operation)은 시계열데이터의 원하는 부분을 선택해 내는 연산으로 이름 또는 시간 또는 속성에 대한 조건을 만족시키는 원천 시계열데이터의 부분을 목적 시계열데이터로 생성한다. 둘째, 축적연산(accumulation operation)은 시간을 기준으로 이루어지며 원천 시계열데이터의 각 데이터점들에 대하여 그 데이터점 전후의 데이터들을 가지고 그 데이터점에 대응하는 목적 시계열데이터를 생성한다. 셋째, 집단연산(aggregation operation)은 시간 또는 이름을 기준으로 시계열데이터들을 서로 상호 배타적으로 나누어 적용되는 연산을 말한다. 넷째, 복합연산(composition operation)은 두개 이상의 원천 시계열데이터의 데이터를 가지고 목적 시계열데이터를 생성해 내는 연산을 말한다.

2.2 시계열 주식데이터의 구조와 연산

주식데이터중 본 논문에서 다루려고 하는 데이터는 주식의 가격 등과 같은 시계열데이터들로 여기에서는 이를 시계열 주식데이터라고 하기로 한다. 시계열 주식데이터는 기본적으로 시계열데이터로서 앞절에서 서술한 시계열데이터의 구조를 가지며 시계열데이터에서의 연산들을 기본적인 연산으로 하여 보다 복잡한 연산을 수행하게 된다.

앞에서 설명한 바와 같이 시계열 주식데이터의 구조는 $\langle s, (t, \bar{a})^* \rangle$ 로 나타낼 수 있다. 여기

서 s 는 시계열 주식데이터의 이름으로 종목명이 되고 t 는 시간으로 일별 주식데이터의 경우 날짜가 된다. 그리고 a 는 속성으로 각 종목에 대하여 해당하는 날짜의 시가(open price), 종가(closing price), 고가(high price), 저가(low price) 그리고 거래량(trading volume)등이 있다. 우리는 이 5가지를 주식가격의 분석을 위하여 시계열데이터로서 관리되어야 하는 기본적인 값으로 삼는다.

이러한 시계열 주식데이터의 구조를 테이블 형식으로 나타내 보면 그림1과 같다.

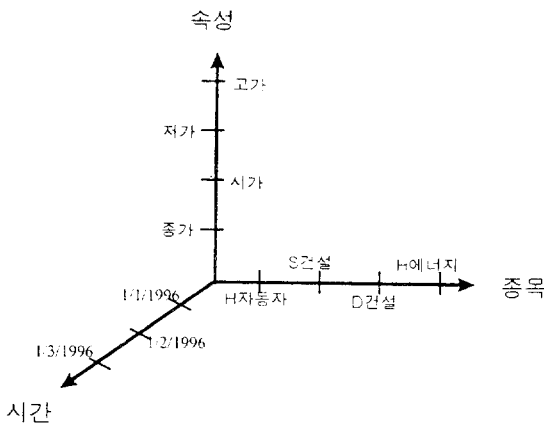
종목명	날짜	시가	종가	고가	저가	거래량
H자동차	1/1/1996	13000	12800	13500	12700	116753
	1/2/1996	12800	12600	13200	12400	127482
	:	:	:	:	:	:

[그림1] 시계열 주식데이터의 구조

사용자가 원하는 주식정보를 검색해 내기 위해서는 필요한 주식데이터가 무엇인가를 나타내는 조건이 필요하다. 시계열 주식데이터를 검색해 내기 위한 조건의 기준으로 사용되는 것은 종목, 시간, 그리고 속성이다.

그림2는 종목, 시간, 속성을 3차원 공간의 개별축으로 표현한 그림이다. 시계열 주식데이터에 대한 검색연산은 이 3차원 공간중 사용자가 원하는 부분에 해당하는 데이터를 추출해 내는 것으로부터 시작된다. 그리고 이는 앞절에서 설명한 선택연산에 해당한다.

앞절에서 설명한 선택연산 이외의 집단연산, 축적연산, 복합연산은 선택연산의 결과로 추출된 데이터에 일련의 계산을 적용한 것이다. 예를 들어, 특정일의 제조업체들의 주식가격의 평균은 특정일의 제조업체의 주식가격들을 검색해 내는 선택연산을 적용한 다음, 이 데이터들에 평균을 구하는 계산을 적용한 집단연산이며, 어느 특정종목의 25일 주가 이동평균은 그 종목의 25일 동안의 주식가격을 검색해 내는 선택연산을 적용한 다음 이 데이터에 평균을 구하는 계산을 적용한 축적연산이다.



[그림2] 시계열 주식데이터의 검색기준

따라서 시계열 주식데이터에서의 연산은 계산의 대상이 되는 시계열 주식데이터를 얻어내는 연산인 선택연산과 선택연산을 통해 선택되어진 데이터에 적용되는 일련의 계산을 행하는 계산연산으로 나누어 진다고 할 수 있다.

3. 주식데이터 관리시스템

이 장에서는 주식데이터 관리시스템의 기본적인 구성요소와 주가데이터에 대한 질의 그리고 주식데이터 관리시스템에 대한 요구사항에 대하여 설명하고자 한다.

3.1 구성요소

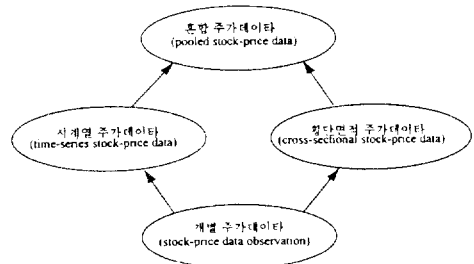
주식데이터 관리시스템은 주식데이터를 보관하고, 사용자가 주식에 관한 일련의 정보를 필요로 할때 그 필요로 하는 정보를 제공해 주는 기능을 가진다. 따라서 주식데이터 관리시스템은 주식데이터에 대한 분석이나 복잡한 연산을 쉽게 구현할 수 있도록 지원해 주어야 한다.

본 논문에서 다루려고 하는 주식데이터 관리시스템은 주식에 관한 데이터의 관리와 이를 검색하고 검색된 데이터에 적용하는 연산을 그 범위로 하며, 여기에서는 주식데이터 관리시스템의 구성요소를 시간-변동 데이터(time-variant data), 시간-불변동 데이터(time-invariant data), 연산(operation), 달력(calendar)으로 나눈다. 좀더 자세히 말하면, 시간-변동 데이터는 시간의 흐름에 따라 그 값의 변화가 기록, 관리되는 데이터로서 주가가격과 거래량 등이 있다. 시간-불변동 데이터는 시간의 흐름에 따른 그 값의 변화가 기록, 관리되지 않는 데이터로

서 각 종목별 상세정보와 종목들의 그룹에 관한 정보가 있다. 각 종목별 상세정보는 종목별로 발행회사, 주식종류(우선주 또는 보통주) 등이 있으며, 종목들의 그룹에 관한 정보는 종목들을 부별, 자본금규모별, 산업별 등으로 나눌 때 각 그룹에 속하는 종목들이 무엇인가에 관한 정보를 말한다. 주식데이터 관리시스템에서의 연산은 위에서 서술한 시간-변동 데이터와 시간-불변동 데이터에 적용되는 일련의 조작(manipulation)들로서 다양한 검색을 위한 선택연산과 검색된 데이터들을 분석하기 위한 연산이 있다. 달력은 주식데이터들이 데이터를 가질 수 있는 시간점들의 집합으로 주식시장의 상장여부 등과 같은 각 시간점에 대한 정보를 관리한다. 일별 주식데이터에서는 일을 단위로 한 달력을 가지게 된다.

3.2 주가데이터에 대한 질의

주식데이터 관리시스템의 사용자가 시스템을 통하여 얻고자 하는 주가데이터에 관한 정보는 그룹3과 같이 3가지 단계로 정리할 수 있다.



[그림 3] 주가데이터에 대한 질의

가장 하위 단계는 개별 주가데이터에 관한 정보로서 이는 특정종목 특정시점의 주가데이터에 관한 정보이며 주가데이터에 대한 질의에서 얻을 수 있는 가장 기본이 되는 정보이다.

개별 주가데이터에 관한 정보가 모이면 다음 단계의 주가데이터에 관한 정보가 생성된다. 다음단계의 정보로는 시계열 주가데이터에 관한 정보와 횡단면적 주가데이터에 관한 정보가 있다. 시계열 주가데이터에 관한 정보는 특정종목 특정기간의 주가데이터에 관한 정보로서 이는 개별 주가데이터를 시간순으로 모은 배열에 해당하며, 이를 통해 시간의 흐름에 따른 특정종목의 주가데이터 값의 변화를 알 수 있다. 그리고 횡단면적 주가데이터에 관한 정보는 특정시점의 여러종목이 갖는 주가데이터 묶음에 대한 정보를 갖는데 대개 관련된 종목을 그룹으로 묶은 정보이다. 따라서 이는 개별 주가데이터중 같은 시점의 데이터를 모은 것으로 이를 통해 한 시점에서의 그룹별 주가데이터 값의 상태를 알 수 있다. 주가 데이터에 대한 질의 중 가장 상위 단계는 혼합 주가데이터에 관한 정보로서 이는 특정기간 여러종목(그룹)의 주가데이터에 관한 정보이다. 이는 시계열 주가데이터와 횡단면적 주가데이터를 혼합해 놓은 것으로 이를 통해 시간의 흐름에 따른 그룹별 주가데이터 값의 변화를 알 수 있다.

3.3 주식데이터 관리시스템에 대한 요구사항

이 절에서는 주식데이터 관리시스템이 가져야할 특징을 구조적인 측면과 연산적인 측면으로 나누어 서술한다. 여기서 말하는 구조적인 측면의 요구사항은 주식데이터 관리시스템이 데이터의 저장소로서 가져야할 특징을 말하며, 연산적인 측면의 요구사항은 주식데이터 관리시스템이 저장되어 있는 주식데이터를 가지고 조작을 하는데 있어서 가져야할 특징을 말한다.

1) 구조적인 측면의 요구사항

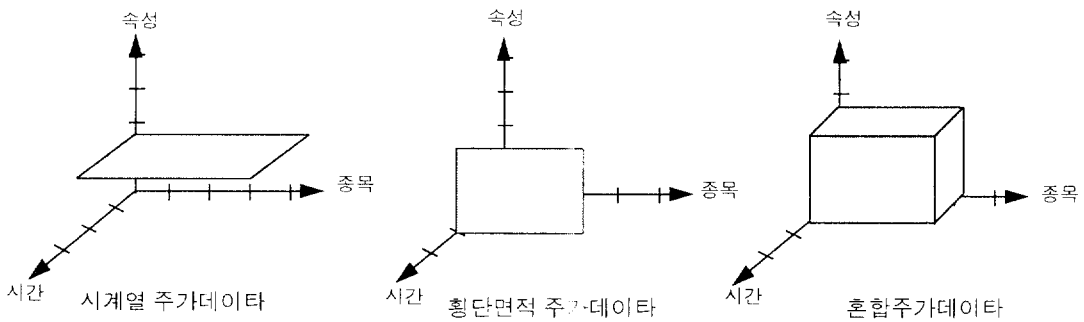
첫째, 주식데이터 관리시스템은 3.2절에서 언급한 바와 같이 개별 주가데이터는 물론 시계열 주가데이터와 횡단면적 주가데이터를 동시에 관리할 수 있어야 한다. 그러므로 주식데이터 관리시스템은 사용자가 주식데이터를 시계열데이터로서 시간축을 중심으로 접근하는 것과 횡단면적으로 종목축을 중심으로 접근하는 것을 모두 허용하여야 하며, 가능하면 이를 사용자가 쉽고 빠르게 처리할 수 있어야 한다. 둘째, 주식데이터 관리시스템안에 데이터를 모델링함에 있어서 재무상품들의 의미(semantics)를 반영할 수 있어야 한다. 재무상품들이 가지고 있는 의미를 충분히 데이터 모델에 표현하기 위해서는 시스템에서 기본적으로 제공해 주는 데이터형을 이용하여 사용자가 나름대로 주식데이터의 의미를 잘 반영할 수 있는 새로운 데이터형을 정의할 수 있어야 한다. 사용자정의 데이터형을 사용함으로써 오는 가장 큰 장점은 사용자가 현실에서 인식하는 재무상품 구조를 그대로 데이터 모델에 옮김으로써 그 데이터 모델을 이용하는 사용자가 논리적으로 실제 현실과 일관성을 가지고 데이터에 접근할 수 있도록 한다는 점이다. 셋째, 주식데이터 관리시스템은 공통적인 성질을 가지는 종목들을 부별, 자본금규모별, 산업별 등의 그룹으로 묶을 수 있는 메카니즘을 제공하여야 한다.

2) 연산적인 측면의 요구사항

주식데이터를 이용한 분석에는 복잡한 계산을 필요로 하는 분석방법이 많이 존재한다. 그러므로 주식데이터 관리시스템은 이러한 복잡한 계산에서 필요로 하는 주식데이터에 관한

자료를 빠르고 효율적으로 제공해줄 수 있어야 하며 이러한 자료를 이용하여 분석자가 행하고자 하는 분석기법을 프로그램상에 용이하게 구현할 수 있는 환경을 제공해 주어야 한다. 이러한 연산적인 측면에서의 요구사항을 만족시키기 위하여 주식데이터 관리시스템이 갖추어야 할 특징들은 다음과 같다. 첫째, 다양한 선택

연산을 지원해 주어야 한다. 앞에서 설명한 바와 같이 기본적으로 시계열 추가데이터, 횡단면적 추가데이터 그리고 혼합 추가데이터를 검색해 낼수 있어야 한다. 이를 선택연산을 위한 3차원 공간에 나타내보면 그림 4와 같다. 그리고 이러한 선택연산에 있어서 특히 시간을 기준으로한 다양한 조건을 줄 수 있어야 한다.

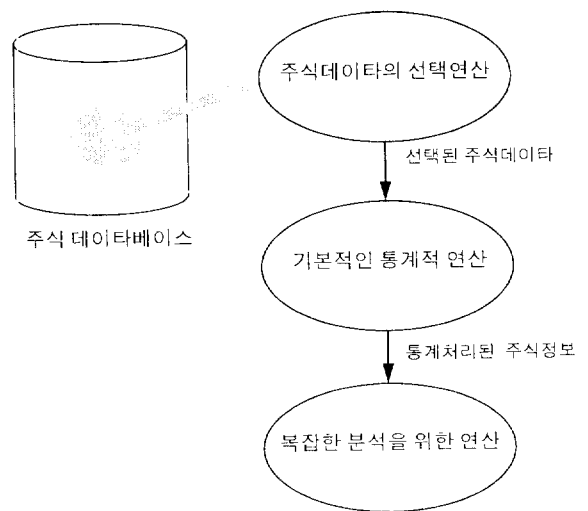


[그림 4] 주식데이터에 대한 선택연산

둘째, 주식데이터에 대한 선택연산을 통하여 얻어진 주식데이터 값들에 적용될 수 있는 기본적인 통계적 연산들을 제공해 주어야 한다. 셋째, 기본적인 통계적 연산을 통하여 얻어진 통계치들을 이용하는 각종 분석방법을 정의할 수 있는 확장성을 제공하여야 한다.

주식데이터 관리시스템에서는 분석자가 주식데이터를 원하는 기준에 따라 검색하는 방법과 검색한 주식데이터에 적용될 수 있는 기본적인 연산을 제공해주고 이로부터 각종 분석방법으로 확장해 나가는데 있어서 그 기본적인 연산을 일관성있게 적용할 수 있어야 한다. 이러한 3가지 연산적인 측면에서의 요구사항을 정리하면 그림5와 같다.

그림5에서 나타낸 바와 같이 주식데이터의 선택연산은 주식 데이터베이스에서 사용자가 원하는 주식데이터를 뽑아내며, 기본적인 통계적 연산은 선택된 주식데이터를 가지고 기본적



[그림 5] 주식데이터에 적용되는 연산의 구조

인 통계처리를 하게된다. 그리고 복잡한 분석을 위한 연산은 통계처리된 주식정보를 가지고 각종 분석연산을 하게 된다. 이와 같이 주식데이터 관리시스템이 제공해야 하는 연산은 계층

적인 구조를 가지며, 본 논문에서는 이를 연산 계층(operation hierarchy)이라고 한다. 네째, 그룹에 관한 연산을 제공해 주어야 한다. 앞의 구조적인 측면에서 설명한 바와 같이 종목들은 여러가지 기준에 따라 그룹을 형성한다. 그러므로 위에서 설명한 주식데이터에 적용되는 연산이 이러한 그룹단위로도 적용될 수 있어야 한다.

점을 가진다. 그러나 주식데이터와 같이 시간 축으로 무한대로 증가하는 시계열데이터를 다루거나 모델링하려는 대상이 계층적인(hierarchical)구조를 갖는 경우에는 이를 표현하고 자료를 다루는데 있어서 약점을 가지고 있다. 이 절에서는 관계형 데이터베이스를 이용하여 시계열 주식데이터를 다루는데 있어서의 약점들에 관하여 좀 더 구체적으로 설명하고자 한다.

4. 주식 데이터베이스

이 장에서는 앞에서 설명한 주식데이터 관리 시스템에 대한 요구사항을 만족시키는데 있어서 관계형 데이터베이스가 가지는 약점과 객체 지향 데이터베이스의 적용 가능성에 대하여 설명하고자 한다.

1) 구조적인 측면의 문제점

첫째, 시계열데이터와 횡단면적 데이터를 동시에 관리하는데 어려움을 갖는다. 주식데이터를 모든 종목에 대해 매일 한번씩 관리하기 위해 날짜 필드를 추가하여 그림6-a와 같은 테이블로 관리한다고 하면, 횡단면적 데이터는 쉽게 검색할 수 있으나 일정기간동안 한 종목에 대해 시계열데이터를 얻고자 하는 경우, 매번 종목에 대해 검색한 후 날짜에 대해 정렬하여야 한다. 종목별 시계열데이터를 쉽게 검색할 수 있도록 그림6-b와 같이 종목별로 다른 테이블로 관리한다고 하면, 종목별 시계열데이터의 검색은 용이하나 특정일의 전종목의 데이터를 필요로하는 경우에 종목수 만큼의 테이블을 검색하여야 하는 불편함이 생긴다.

4.1 관계형 데이터베이스의 문제점

관계형 데이터베이스는 모델링하려는 대상을 2차원적인 평면테이블로 모두 표현한다. 이 방법은 데이터베이스를 이해하기가 쉽고, SQL이나 QUEL과 같은 강력한 질의어를 가지고 있어서 횡단면적 데이터를 다루는데 있어서는 장

날짜	종목	종가
1/1/1996	stock 1	100
1/1/1996	stock 2	150
:	:	:
1/1/1996	stock 1000	130
1/2/1996	stock 1	110
:	:	:

(a) 횡단면적 데이터 위주의 관리

날짜	종가
1/1/1996	100
1/2/1996	110
:	:

stock 1

날짜	종가
1/1/1996	150
1/2/1996	145
:	:

stock 2

날짜	종가
1/1/1996	120
1/2/1996	125
:	:

stock 3

(b) 시계열데이터 위주의 관리

[그림 6] 혼합데이터관리의 문제점

둘째, 관계형 데이터베이스는 복잡한 객체를 모두 2차원적인 테이블형식으로 표현하여야 하므로 현실에서 가지는 객체의 의미(semantic)가 여러 테이블에 나뉘어 저장되게 된다. 이는 관계형 데이터베이스에서의 정규화(normalize)의 필요성 때문에 발생하게 되며, 현실에서의 객체의 의미를 유지하기 위하여 정규화를 하지 않으면 삽입, 삭제, 수정시 문제(anomaly)가 발생하게 된다. 데이터 모델이 간단한 경우에는 이러한 점이 큰 문제가 되지 않으나 주식에 관한 데이터뿐 아니라 파생증권과 같은 여러가지 재무상품에 관한 데이터들을 주식데이터와 연관지어 관리하게 되면 데이터 모델이 보다 복잡하게 되며 이러한 경우, 객체가 가지는 현실에서의 의미를 데이터 모델안이 표현해 주는 것이 데이터 모델을 이용하는 사용자에게 편리함을 제공해 준다. 셋째, 개별종목들이 모여 그룹을 형성하는 것과 같은 계층형구조를 구현하는데 있어서 어려움을 갖는다. 두 테이블간의 관계는 데이터가 저장되어 있는 곳의 주소인 포인터를 이용하면 효율적으로 구현될 수가 있다. 그러나 관계형 데이터베이스에서는 두 테이블간의 관계를 한 테이블의 기본키(primary key)와 다른 테이블의 외래키(foreign key)로 이를 표현하여야 하기 때문에 계층형구조를 갖는 데이터 모델을 관계형 데이터베이스에서 구현하기 위해서는 테이블마다 많은 키값을 저장

해야만 하기 때문에 많은 비효율성을 갖는다.

2) 연산적인 측면의 문제점

첫째, 관계형 데이터베이스에서는 앞에서 언급한 바와 같이 데이터들이 물리적으로 여러 정규화된 2차원적 테이블에 나누어 저장되기 때문에 사용자가 원하는 정보를 질의를 통해 얻기 위해서는 먼저 여러 테이블들을 조인하여야 한다. 데이터 모델이 점차 복잡하여짐에 따라 데이터들을 점점 더 여러 테이블에 나누어 관리되게 되고 여러 테이블에 흩어져 있는 데이터를 조합하여 정보를 얻기 위해서는 질의과정이 점점 더 복잡하여지고 어려워지게 된다. 둘째, 날짜에 대한 기준을 쉽고 다양하게 줄 수 있는 연산을 제공해 주지 못한다. 주식데이터는 기본적으로 시계열데이터로서 데이터의 발생시점을 나타내는 날짜를 가지게 된다. 주식데이터 분석에 필요한 데이터 질의시에 날짜를 기준으로 데이터를 질의 하는 경우가 많으며 이때 날짜에 관한 다양한 연산(예를 들어, 특정일을 기준으로 5일 간격으로 데이터를 검색하고 싶은 경우 이를 지원해 줄 수 있는 연산)을 제공해주면 사용자가 원하는 데이터를 보다 쉽게 얻어낼 수 있으나 관계형 데이터베이스의 질의어에서는 이러한 날짜에 관한 연산을 다양하게 제공해 주지 못한다. 관계형 데이

타베이스의 이러한 어려움을 극복하기 위하여 Sarda는 HSQL을 만들어 SQL에 시계열데이터를 다루기 위한 기능을 추가하였고[11], Snodgrass는 TQuel을 개발하여 QUEL을 보완하고자 하였다[15].

4.2 객체지향 데이터베이스의 적용가능성

이 절에서는 4.1절에서 언급한 관계형 데이터베이스의 문제점을 극복하기 위한 방안으로 객체지향 데이터베이스의 적용가능성을 구조적인 측면과 연산적인 측면으로 나누어 설명하고자 한다.

1) 구조적인 측면의 적용가능성

객체지향 데이터베이스에서는 정수형, 실수형, 문자형과 같은 기본적인 데이터형(primitive data type)을 이용하여 보다 복잡한 데이터형을 사용자가 정의하여 이를 데이터형으로 사용할 수 있다. 이는 복잡한 특성들을 가지는 주식데이터들을 모델링할 때 현실에서의 의미를 보다 많이 포함할 수 있도록 해주며 이를 통해 주식데이터를 현실에서 사용자가 인식하는 대로 다룰 수 있도록 하여 데이터를 다룸에 있어서 일관성과 편리함을 제공해 준다. 그리고 이는 주식데이터뿐 아니라 더 나아가 서로 관계를 가지는 재무상품들에 대한 데이터 모델링시 그 의미를 데이터베이스 안에 풍부하게 담을 수 있게 해 준다.

앞에서 살펴본 바와 같이 주식데이터는 시계열데이터의 성질과 횡단면적 데이터의 성질을 동시에 갖는다. 그런데 이를 관계형 데이터베이스에서 관리하는 데는 앞에서 밝힌 바와 같은 문제점이 있게 된다. 객체지향 데이터베이스에

서는 객체안에 객체를 포함하는 기능과 inverse표현기능을 이용하여 이를 해결할 수 있다. inverse표현기능이란 두 객체간의 관계가 양방향으로 연결되는 경우 이를 inverse라는 키워드를 이용하여 시스템에 알려주면 시스템이 두 객체간의 관계를 나타내는 속성간의 참조무결성(referencial integrity)을 유지 시켜주는 기능을 말한다. 이러한 기능을 이용하여 특정 종목에 대한 시계열 데이터로의 접근뿐 아니라 특정시점에서의 횡단면적데이터로의 접근을 용이하게 할 수 있다.

주식데이터들은 그룹으로 묶어서 관리할 필요가 있다. 객체지향 데이터베이스에서는 속성의 값으로 객체식별자를 넣을 수가 있고 또한 한 속성이 여러개의 값을 가질 수도 있으므로 이를 이용하여 주식데이터를 그룹으로 쉽게 관리해 줄 수 있다.

2) 연산적인 측면의 적용가능성

주식데이터에 대한 선택을 위한 기준은 시간, 종목 그리고 속성이다. 주식데이터는 기본적으로 시계열데이터이기 때문에 이 3가지 기준중 특히 시간을 기준으로한 선택연산을 필요로 한다. 그런데 관계형 데이터베이스는 날짜를 기준으로 한 검색방법을 다양하게 제공하고 있지 못하다. 객체지향 데이터베이스에서는 객체타입 안에 그 객체의 속성들에 적용될 수 있는 연산을 정의하여 캡슐화(encapsulation)시킬 수가 있다. 객체지향 데이터베이스의 이러한 기능을 이용하여 날짜를 이용한 다양한 선택연산을 가능하도록 할 수 있다.

주식데이터를 가지고 여러가지 분석을 하기 위해서는 주식데이터에 대한 기본적인 통계적 연산을 쉽게 수행할 수 있어야 한다. 그리고

이러한 통계적 연산을 다양하게 선택된 주식데이터에 적용할 수 있어야 한다. 이러한 요구사항을 만족시키기 위해서 객체지향 데이터베이스에서 제공하는 계승(inheritance)메카니즘을 이용할 수 있다. 연산에 있어서의 계승 메카니즘의 이용은 주식데이터에 대한 복잡한 연산의 구현을 위한 확장성을 제공해 준다.

5. 객체지향 데이터베이스를 이용한 주식데이터 관리시스템

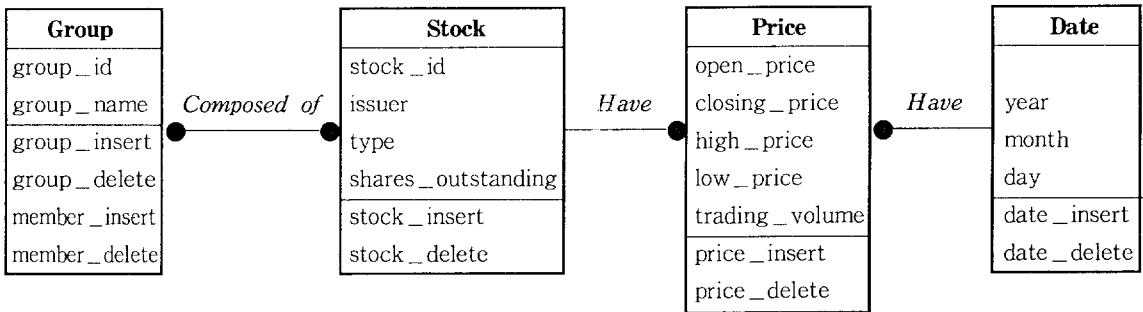
이 장에서는 본 논문에서 제시하는 객체지향

데이터베이스를 이용한 주식데이터 관리시스템의 기본적인 데이터 구조와 연산의 구조에 대하여 설명하고자 한다.

5.1 데이터 구조

주식데이터를 객체지향 데이터베이스에서 관리하기 위해서는 데이터베이스를 설계하기에 앞서 현실에서 주식의 의미를 파악하여야 한다. 이를 위해 OMT(object modeling technique)[10]를 이용하여 객체 모델링(object modeling)을 해보면 그림7과 같다.

그림 7과 같은 현실의 의미를 데이터 모델안에 표현하기 위하여 본 논문에서 제시하는 주

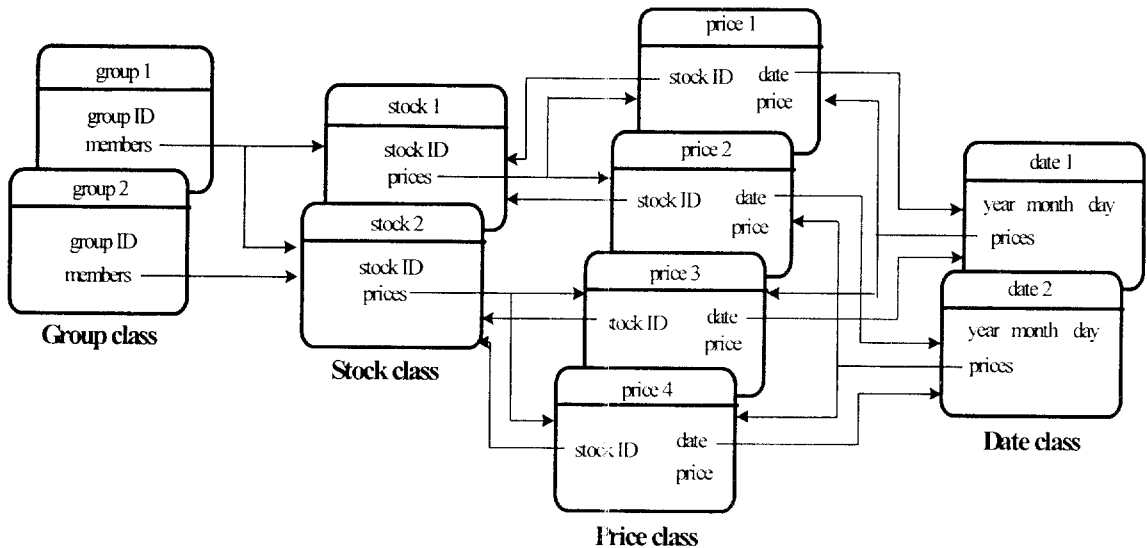


[그림 7] 주식에 대한 객체 모델링

식데이터 관리시스템에서는 주식(Stock)클래스, 가격(Price)클래스, 날짜(Date)클래스, 그룹(Group)클래스를 데이터를 관리하기 위한 기본 클래스로 제시한다. 주식클래스는 주식데이터중 시간-불변동 데이터(종목코드, 발행회사명, 발행수 등)를 관리하기 위한 클래스이고, 가격클래스는 주식데이터중 시간-변동 데이터(시가, 종가, 고가, 저가 등)를 관리하기 위한 클래스이다. 그리고 날짜클래스는 날짜에 관한 정보를 관리하면서 날짜에 관한 연산을 지원하기 위한 클래스이며, 그룹클래스는 공통

적인 성질을 가지는 주식종목들을 그룹으로 묶어주기 위한 클래스이다.

그림7에서 클래스 간의 관계는 실선으로 표시되어 있으며, 실선 끝의 검은 점은 두 클래스간의 관계에 있어서 검은 점이 있는 쪽이 복수개가 될 수 있음을 의미한다. 즉, 하나의 그룹은 여러개의 주식으로 이루어지며, 하나의 주식은 여러개의 그룹에 속할 수 있음을 나타낸다. 또한 각 주식은 시간의 흐름에 따라 여러개의 가격을 가지게 되며, 반대로 하나의 날짜에 대해서는 여러 주식의 가격이 형성됨을



[그림 8] 객체지향 데이터베이스로 구현한 주식데이터의 구조

나타낸다. 이러한 현실의 의미를 객체지향 데이터베이스에 반영하여 구현한 주식데이터의 구조는 그림 8과 같다.

그림8에서는 각 클래스들의 중요한 속성들을 나타내었다. 그림8에서 보는 바와 같이 각 그룹은 그 그룹의 멤버에 해당하는 주식들의 집합을 나타내는 속성을 가지며(Group class에서 members), 각 주식은 그 주식의 일별 주가 데이터들의 집합을 나타내는 속성을 가진다(Stock class에서 prices). 이는 현실에서의 주식데이터들간의 관계를 그대로 데이터 모델로 표현한 것으로 사용자 정의 타입의 사용과 복수개의 값을 하나의 속성에 저장할 수 있는 다중값 속성(multi-valued attribute) 등의 객체지향 데이터베이스 특성을 이용하여 재무상품의 의미를 데이터 모델로 직접적으로 표현한 것을 의미한다. 본 논문에서는 주로 주식데이터에 대한 모델만을 고려하였으나 주식이외의

다른 재무상품이나 파생증권등의 데이터까지 포함한 데이터 모델을 설계할 때는 이러한 기능이 보다 많이 필요하게 된다. 각 일별 주가데이터들은 그 데이터가 생성된 날짜를 나타내는 속성을 가진다(Price class에서 date). 또한 각 날짜들은 그 날짜에 생성된 주가데이터들을 나타내는 속성을 가진다(Date class에서 prices). 따라서 가격과 날짜와의 관계는 양방향으로 연결되게 된다. 이 연결관계는 객체지향 데이터베이스의 특성중 객체간 상호연결성(inverse-relationship)을 이용한 것이다. 이렇게 함으로써 가격에서 날짜로의 접근과 날짜에서 그 날에 형성된 가격으로의 접근이 모두 가능하여 시계열데이터와 횡단면적 데이터의 동시관리가 이루어지게 된다. 이에 대하여 보다 자세히 알아보면 다음과 같다.

```

define type PriceType          /* 가격을 나타내는 객체타입 */
( structure
  DateType          date inverse prices of DateType,
  .....          )
define type DateType          /* 날짜를 나타내는 객체타입 */
( structure
  set of PriceType prices inverse date of PriceType,
  .....          )
    
```

[그림 9] 가격, 날짜를 나타내는 객체타입

그림9를 보면 주식의 가격을 나타내는 객체 타입인 PriceType과 날짜를 나타내는 객체 타입인 DateType이 정의되어 있다. PriceType은 그 가격이 형성된 날짜를 가지며, Date Type은 그 날짜에 형성된 주식의 가격들을 가진다. 이러한 관계를 객체지향 데이터베이스에서는 PriceType에서 date속성과 DateType에서 prices속성으로 표현하고 이들간에 서로 inverse관계가 있음을 선언할 수 있다. 이렇게 객체타입을 정의하면 시스템에 의해 두 속성간

의 참조부결성이 유지되며, 날짜를 통해서 그 날짜에 형성된 주식의 가격들에 접근할 수 있으므로 횡단면적 데이터를 효율적으로 제공할 수 있다. 그리고 주식과 그 가격들을 그림8과 같이 연결하면 주식ID를 통해서도 그 주식의 가격에 접근할 수 있으므로 시계열데이터 또한 쉽게 접근할 수 있다. 객체지향 데이터베이스에서는 속성의 값으로 객체식별자를 넣을 수 있다고 하였는데 이를 이용하여 다음과 같이 주식데이터를 그룹으로 관리해 줄 수 있다.

```

define type GroupType          /* 그룹을 나타내는 객체타입 */
( structure
  set of StockType members,
  .....          )
    
```

[그림 10] 그룹을 나타내는 객체타입

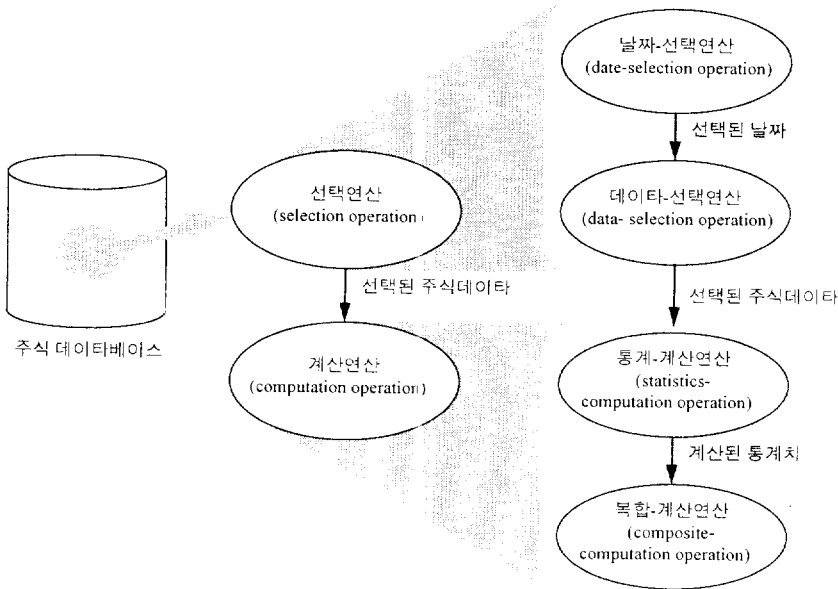
그림10에서 members에는 그 그룹에 속하는 주식종목들의 데이터가 저장되어 있는 객체의 객체식별자들이 저장되게 된다. 따라서 한 그룹에 대한 정보를 알고 싶은 경우에는 그 그룹의 이름을 통해서 그 그룹에 속하는 주식종목들의 데이터에 접근할 수 있다.

5.2 주식데이터에 대한 연산

주식데이터 관리시스템에서의 선택, 분석 및 계산 등의 연산은 그림5의 구조에서 설명된 바와 같이 계층구조를 가진다. 본 논문에서는 이를 연산계층(operation hierarchy)이라고 한다. 연산계층은 크게 2개의 계층으로 나눌 수 있다. 이중 상위계층은 선택연산으로 주식데이터

를 저장, 관리하고 있는 데이터베이스에서 사용자가 필요로 하는 데이터를 선택해 내는 연산을 말한다. 그리고 하위계층은 계산연산으로 선택연산에 의하여 선택된 데이터를 가지고 사용자가 필요로 하는 정보를 얻기 위하여 일련의 계산을 하는 연산을 말한다. 이중 상위계층인 날짜-선택연산은 다시 2개의 계층으로 나눌 수 있는데 이는 날짜-선택연산(date-selection operation)과 데이터-선택연산(data-selection operation)이다. 날짜-선택연산은 사용자가 원하는 데이터를 선택하기 위하여 원하는 데이터를 가지는 날짜를 구해내는 연산으로 이를 통해서 사용자는 다양한 시간기준으로 데이터를 선택

할 수 있게 된다. 그리고 데이터-선택연산은 날짜-선택연산에 의하여 구해진 날짜들에 대하여 검색을 원하는 종목 또는 그룹의 데이터를 구해내는 연산이다. 하위계층인 계산연산 또한 2개의 계층으로 다시 나눌 수 있는데 이는 통계-계산연산(statistics-computation operation)과 복합-계산연산(composite-computation operation)이다. 통계-계산연산은 주식데이터 분석을 하는데 있어서 통계값을 계산하는 연산이다. 그리고 복합-계산연산은 이러한 통계값을 이용하여 보다 복잡한 분석을 위한 계산을 하는 연산이다. 이와 같은 연산계층을 그림으로 나타내 보면 그림11과 같다.



[그림 11] 연산계층의 구조

이러한 연산계층의 개념을 주식데이터 관리 시스템에 넣기 위하여 객체지향 데이터베이스에서 제공하는 계승(inheritance)메카니즘을 이용한다. 가장 상위계층인 날짜-선택연산들은 사용자가 원하는 데이터들의 날짜들을 선택하는

연산으로 5.1절에서 설명한 날짜(Date)클래스의 연산으로 들어가게 된다. 날짜-선택연산들을 가지고 있는 날짜 클래스를 두번째 계층인 데이터-선택연산 클래스가 계승하도록 하고, 이 데이터-선택연산 클래스를 세번째 계층인 통계-

계산연산 클래스가 계승하도록 한다. 이렇게 함으로써 연산계층의 마지막 계층인 복합-계산 연산 부분에서는 통계-계산연산 클래스만 포함 하여도 날짜-선택연산, 데이터-선택연산, 통계-계산연산들을 이용할 수 있게 되며 이를 이용하여 보다 복잡한 계산연산을 수행할 수 있다.

이러한 연산에 있어서의 계승 메카니즘의 이용은 주식데이터에 대한 복잡한 연산의 구현을 위한 확장성을 제공해 준다. 예를 들어, 복합-계산연산을 구현하는 부분에서는 그림 12와 같은 방법으로 상위 클래스들에서 정의해 놓은 연산들을 사용할 수 있다.

```

Statistics_computationType stat;                               /* 통계-계산연산 클래스의 선언 */
....
date_set <- stat. interval(1/1/1996, 7/31/1996);             /* 날짜-선택연산의 적용 */
price_set <- stat. stock_date_set(IBM, date_set);           /* 데이터-선택연산의 적용 */
variance_IBM <- stat. variance(price_set, close_price);     /* 통계-계산연산의 적용 */
....
    
```

[그림 12] 복합-계산연산의 구현 예

그림12에서는 1996년 1월 1일부터 7월 31일 까지의 IBM주식의 종가의 분산을 구하는 부분을 나타내며, 이렇게 구한 분산을 이용하여 복합-계산연산 내에서 보다 복잡한 연산을 쉽게 구현할 수 있다.

주가데이터에 대한 선택은 그림4에서 보인 바와 같이 시계열 주가데이터에 대한 선택, 횡단면적 주가데이터에 대한 선택, 혼합 주가데이터에 대한 선택으로 나눌 수 있다. 시계열 주가데이터에 대한 선택은 날짜-선택연산을 통

하여 두개 이상의 날짜를 선택하고 데이터-선택연산을 통하여 한 종목에 대한 데이터를 선택한 경우이다. 횡단면적 주가데이터에 대한 선택은 날짜-선택연산에서 특정일을 선택하고 데이터-선택연산에서 두개 이상의 종목을 선택한 경우이다. 혼합 주가데이터에 대한 선택은 날짜-선택연산에서 두개 이상의 날짜를 선택하고 데이터-선택연산에서 두개 이상의 종목을 선택한 경우이다. 이를 정리해 보면 표1과 같다.

<표 1> 주가데이터에 대한 질의와 선택연산과의 관계

	시계열 주가데이터	횡단면적 주가데이터	혼합 주가데이터
날짜-선택연산	두개 이상의 날짜	한개의 날짜	두개 이상의 날짜
데이터-선택연산	한개의 종목	두개 이상의 종목	두개 이상의 종목

6. 구 현

본 논문에서 제시한 주식데이터 관리시스템의 원형시스템을 객체지향 데이터베이스 개발틀인 ObjectStore를 이용하여 구현하였다. ObjectStore는 C++를 프로그래밍 언어로 이용하여 객체지향 데이터베이스를 구현할 수 있도록 되어있다.

본 논문의 5장에서 제시한 데이터 구조와 연산계층을 구현하여본 결과 원하는 날짜기준과 종목기준에 따라 다양한 검색이 가능함을 확인할 수 있었고 원하는 통계값을 연산계층의 3가지 단계인 날짜-선택연산, 데이터-선택연산, 통계-계산연산을 통하여 사용자가 간편하게 구할 수 있음을 확인할 수 있었다.

여기에서 구현한 주식데이터 관리시스템에서는 복합-계산연산의 한가지로서 전환사채(convertible bond)에 대한 클래스를 나타내는 객체타입을 정의하고 전환사채의 이론적인 가격을 계산하는 연산을 구현하여 보았으며, 이를 통해 복잡한 분석방법의 구현에 대한 확장성을 보였다.

7. 결 론

본 논문에서는 시계열 데이터로서의 주가데이터를 관리하는 시스템에 대한 요구사항을 구현하는데 있어 관계형 데이터베이스가 부적합함을 설명하고 이러한 한계를 효과적으로 극복하는데 있어 객체지향 데이터베이스의 우월성을 제시하였다. 또한 구체적인 제안으로 객체지향 데이터베이스를 이용하여 주식데이터 관

리시스템을 설계할 때에 가져야 하는 데이터 모델의 기본구조와 주식데이터의 조작을 위한 연산의 기본구조를 제시하였다.

본 논문에서는 주식데이터 관리시스템에 대한 요구사항과 관계형 데이터베이스가 가지는 약점 그리고 객체지향 데이터베이스가 가지는 적용가능성을 구조적인 측면과 연산적인 측면 양쪽에서 파악하였는데 이는 주식데이터 관리시스템을 데이터의 저장소로서의 측면과 저장되어 있는 데이터에 대한 조작에 대한 측면으로 나누어 볼 수 있기 때문이다.

본 논문에서 제시된 데이터 모델의 구조와 연산계층의 구조는 일차적으로 주식데이터에 적용하였으나 그 적용대상이 주식데이터에 국한되지 않고, 시간을 기준으로한 다양한 검색 연산과 복잡한 계산연산을 필요로 하는 다른 시계열데이터를 데이터베이스에 저장관리하는 데에도 적용될 수 있다고 생각된다.

본 논문에서 다루지 못한 연구 과제로는 첫째, 주식뿐 아니라 채권이나 그 밖의 파생증권들에 관한 데이터를 객체지향 데이터베이스로 관리하는데 있어서 이들간의 관계를 포함하여 데이터 모델링을 하는 문제, 둘째, 일별 주가데이터뿐 아니라 거래시점별 주가데이터(lick data)의 관리문제, 셋째, 재부분야의 지식을 시스템안에 포함시키는데 있어서 전문가시스템을 이용하는 문제 등이 있으며 이는 향후 연구과제에서 다루기로 한다.

참 고 문 헌

- [1] Chandra, R. and Segev, A., "Managing Temporal Financial Data in an Exten-

- sible Database" *Proceedings of the 19th VLDB Conference* Dublin, Ireland 1993
- [2] Cochinala, M., and Bradley, J., "A Multidatabase System for Tracking and Retrieval of Financial Data" *Proceedings of the 20th VLDB Conference* Santiago, Chile, 1994
- [3] Dreyer, W., Dittrich, A. K., and Schmidt, D., "Research Perspectives for Time Series Management Systems" *ACM SIGMOD RECORD*, Vol. 23, No. 1, March 1994
- [4] Huh, S. Y., "An Object-Oriented Model Management Framework for Decision Support Systems", Anderson Graduate School of Management, UCLA, 1992
- [5] Kim, W., "Object-Oriented Databases: Definition and Research Directions" *IEEE Transactions on Knowledge and Data Engineering*, Vol. 2, No. 3, September 1990
- [6] Kim, W., *Introduction to Object-Oriented Databases*, The MIT Press 1990
- [7] Nerson, J. M., "O-O Development of a Date and Time management cluster" *Journal of Object Oriented Programming*, March / April 1992
- [8] Peinl, P., Reuter, A. and Sammer, H., "High Contention in a Stock Trading Database: A Case Study" *Proceedings of the ACM SIGMOD Annual Conference*, 1988
- [9] Rozen, S. and Shasha, D., "Using a Relational System on Wall Street: The Good, The Bad, The Ugly, and The Ideal" *Communications of the ACM*, Vol. 32, No. 8, August 1989
- [10] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. and Lorenzen, W., *Object-Oriented Modeling and Design*, Prentice Hall, 1991
- [11] Sarda, N. L., "Extensions to SQL for Historical Databases" *IEEE Transactions on Knowledge and Data Engineering*, Vol. 2, No. 2 June 1990
- [12] Segev, A. and Shoshani, A., "Logical Modeling of Temporal Data" *Proceedings of the ACM SIGMOD Annual Conference*, May 1987
- [13] Silberschatz, A., Stonebraker, M. and Ullman, J., "Database Systems: Achievements and Opportunities" *Communications of the ACM*, Vol. 34, No. 10, October 1991
- [14] Snodgrass, R. and Ahn, I., "A Taxonomy of Time in Databases" *Proceedings of the ACM SIGMOD Annual Conference*, 1985
- [15] Snodgrass, R., "The Temporal Query Language TQuel" *ACM Transactions on Database Systems*, Vol. 12, No. 2, June 1987
- [16] Tansel, A.U., Clifford, J., Gadia, S., Jajodia, S., Segev, A. and Snodgrass, R., *Temporal Databases*, The Benjamin / Cummings publishing company, Inc. 1993

-
- [17] Vatne, B.H., "Avoding Data Chaos",
*Proceedings of International FAME
Users Meeting*, Luxembourg, June
1993
- [18] Wuu, G.T.J. and Dayal, U., "A Uni-
form Model for Temporal Object-
Oriented Database" *IEEE Procee-
dings of the International Conference
on Data Engineering*, 1992