

III. 논문 발표

SDTS 변환 시스템

2

1998. 7. 2

한 기 준
건국대학교

SDTS 변환 시스템[†]

(SDTS Conversion System)

이강준*, 김준종**, 설영민**, 한기준***

(Kang-Jun Lee, Jun-Jong Kim, Young-Min Sul, Ki-Joon Han)

초록

지리 정보 시스템(GIS)은 그 특성상 대용량의 GIS 데이터를 사용하며, 다양한 운영체제와 하드웨어 상에서 구현된다. 이렇게 상이한 운영체제와 하드웨어상의 공간 데이터들은 일반적으로 서로 다른 GIS 데이터 포맷을 갖고 있기 때문에 효율적인 자료 교환이 불가능하다면 데이터 공유가 매우 어려울 뿐만 아니라 공통 데이터의 중복 보관 및 관리로 인해 막대한 경제적 손실을 가져온다. 이와 같은 문제점을 해결하기 위해 국외에서는 이미 GIS 데이터 교환을 위한 방안으로 공통 교환 표준 포맷 작업이 10여년 전부터 진행되어왔으며, 국내의 경우도 국가 차원에서 지리 정보 시스템의 국가 표준을 설정하고, 기본 공간 데이터베이스를 구축하고 있다. 그리하여, 국가 기본 포맷과 공통 데이터 교환 포맷의 표준으로 SDTS(Spatial Data Transfer Standard)를 채택하였다. 본 논문에서는 범용의 SDTS 변환 시스템의 구현에 필요한 공간 데이터 분석, 논리적 설계, 물리적 설계 등의 전반적인 SDTS 변환 과정, 테스트 검증 사항, 그리고 반드시 지켜야 할 규칙들을 제시한다. 마지막으로, 실제 공간 데이터인 GOTHIC과 SDTS 간의 변환 시스템의 설계와 구현에 대해서 언급한다.

Keywords: 지리 정보 시스템, GOTHIC, SDTS, SDTS 변환 시스템, 공간 데이터

1. 서론

지리 정보 시스템에서 사용되는 지리 데이터는 기존의 데이터 저장 방식과는 달리 지리 정보를 위한 공간, 비공간 데이터들로 구성되며, 그 크기도 일반적으로 대용량인 특성을 갖고 있다 [4,10]. 또한, 이러한 지리 정보 시스템은 다양한

분야에서 개발되고 있는데, 예를 들어 위성 자료를 관리하는 위성 전문가 시스템, 항해/항법 지도 정보를 관리하는 항해/항법 시스템, 강수량과 토양 등의 데이터를 가지고 홍수를 예방하는 홍수 재해 방지 시스템 등과 같이 각기 다른 운영체제하에 다른 응용 소프트웨어와 하드웨어 상에서 각기 자신의 목적에 맞는 독특한 지리 정보 시스템들이 구축되고 있다.

[†] 본 논문은 과학기술부의 국가지리정보시스템(NGIS) 기술개발 프로젝트의 연구비 지원에 의해 이루어졌음.

* 건국대학교 컴퓨터공학과 박사과정

** 건국대학교 컴퓨터공학과 석사과정

*** 건국대학교 컴퓨터공학과 교수

서로 다른 지리 정보 시스템들은 서로 간의 데이터들을 긴밀하게 공유할 필요가 발생하지만 이러한 시스템들이 자기 자신들의 데이터 포맷을 가지고 있기 때문에 상호 데이터 교환에 큰 어려움을 가지게 된다. 일반적인 경우에 지리 정보 시스템들은 상호 데이터 교환의 어려움으로 인하여 공통 지리 데이터들의 중복 보관과 관리로 많은 기억 공간의 낭비와 경제적인 손실을 안고 있다. 이러한 문제점을 해결하기 위해 지리 정보 시스템들은 서로간의 공간 데이터들을 다른 시스템과 교환을 위해 데이터 저장 구조의 표준을 채택하게 되었고, 또한 이러한 표준 데이터 포맷으로의 변환 시스템의 개발 작업 또한 활발히 이루어지고 있다 [3,9,12,14].

최근 국내에서는 지리 정보 시스템이 국가 경쟁력 강화 및 행정 생산성 제고 등에 기반이 되는 사회 간접 자본이라는 전제하에 국가 차원에서 지리 정보 시스템의 국가 표준을 설정하고 기본 공간 데이터베이스를 구축하며, 또한 지리 정보 시스템 관련 기술개발을 지원하여 지리 정보 시스템 활용 기반과 여건을 성숙시켜 가고 있다. 또한, 국가 지리 정보 시스템 사업의 일환으로 정보통신부(한국전산원)에서는 표준화분과 위원회를 구성하여 한국 표준을 제정하고 있다. 특히, 국가 기본도 포맷과 공통 데이터 교환 포맷의 표준 제정을 위한 소위원회를 구성하여 표준화 작업을 진행한 결과 공통 데이터 교환 포맷으로 SDTS (Spatial Data Transfer Standard)를 채택하였다 [1,8,11].

본 논문에서는 다양한 GIS 데이터들의 공통 교환 표준 포맷인 SDTS로의 상

호 변환 작업을 수행하는데 필요한 공간 데이터 분석, 논리적 설계, 물리적 설계를 위한 단계 및 방법을 제시하며, SDTS 변환 과정에 따르는 문제점, 지켜야 할 규칙 등을 제시한다. 그리고, GOTHIC과 SDTS 간의 상호 변환 시스템의 설계 및 구현에 대해서 기술한다.

본 논문의 구성은 다음과 같다. 제 2장의 관련연구에서 SDTS 분석과 SDTS 상호 변환 시스템 개발의 예로 든 GOTHIC을 분석한다. 제 3장에서 SDTS 변환 시스템의 전체적인 구조, 변환 흐름, 인코딩, 디코딩 과정, SDTS 프로파일 개발과 수정에 관해 설명한다. 제 4장에서는 SDTS 변환 시스템 설계에 따른 SDTS 변환 시스템 구현을 언급하며, 여기에 GOTHIC의 변환 시스템 제작에 관한 과정을 예로 든다. 제 5장에서는 SDTS 변환 시스템의 테스트 및 검증 방법을 논의하며, 마지막으로 제 6장에서는 본 논문의 결론에 대해 언급한다.

2. 관련 연구

2.1 SDTS 분석

SDTS는 모든 종류의 공간 데이터(지리 정보, 지도)들을 서로 변환 가능하게 해주는 표준으로 상이한 하드웨어, 소프트웨어, 운영체제 사이에서 데이터 교환을 가능하게 한다. 미연방에 의해 추진된 SDTS는 1992년 미연방 표준(FIPS Publication 173)으로 채택되었다. SDTS는 많은 다른 형태, 모델, 구조들로 서로 연관된 속성 데이터들을 FIPS 173을 사용하여 다른 시스템간에 교환될 수 있도록 정의되어 있으며, 이는 크게 3개의 부

본인 Part 1, Part 2, Part 3으로 구성된다. Part 1, 2, 3에 의해 특정 데이터 유형을 위한 구체적인 변환 형식으로서 Part 4는 벡터 데이터를 위한 TVP (Topological Vector Profile)를 포함한다 [11,14].

2.1.1 Part 1

SDTS Part 1은 공간 데이터 표준의 일반적인 정의 및 규정으로 구성된다. Part 1은 5개의 Section으로 분류된다. Section 1은 개론, Section 2는 공간 데이터의 개념적 모델, Section 3은 공간 데이터 질, 그리고 Section 4, 5는 변환과 관련된 레코드와 필드 수준의 내용을 나타낸다.

Section 2의 공간 데이터 개념적 모델을 살펴보면 속성을 가진 공간 개체로서 실제 세계를 표현하는 공간 현상, 공간 현상과 공간 객체간의 관계를 규정하는 공간 형상(Spatial Feature), 그리고 다양한 공간 실체들을 수치적으로 순서적 및 추상적으로 표현하는 공간 객체 등의 모델로 구분된다. 래스터와 벡터 객체들은 Geometry Only(G)와 Geometry and Topology(GT)같은 단순 개체와 분류 개체의 집합으로 표현된다. Geometry-Only(G) 공간 객체들에는 포인트, 라인 세그먼트, 스트링, 호, G-링, 내부 영역, G-다각형, 점, 그리드 셀 등이 있으며, Geometry and Topology (GT) 공간 객체에는 노드, 링크, 체인, GT-링, GT-다각형 등이 있다.

Section 3은 사용에 적합한 데이터 특성이나 데이터의 질에 대한 정보들로서

Lineage, 위치 정확성 (Positional Accuracy), 속성 정확성 (Attribute Accuracy), 논리 일관성 (Logical Consistency), 그리고 완전무결성 (Completeness) 등을 정의하고 있다. Lineage는 최종 수치 화일 생성과 관련된 모든 변형을 비롯한 원본 데이터에 대한 기술과 획득방법의 내용을 기술한다. 위치 정확성은 위치 정보가 실제 위치와 얼마나 근접한지에 대한 정보를 가진다. 속성 정확성은 비위치성 데이터에 대해 기술하며, 논리적 일관성은 인코딩시 공간 데이터 구조와 관계가 충실하게 조작되어 있는지를 검증한다. 마지막으로 완전무결성은 사용된 정의, 척도 (기준), 다른 관련 규정에 대한 정보를 기술한다.

Section 4, 5는 공간 데이터 변환을 위한 논리적 변환 방식을 정의하고 있는데 이 단계는 변환의 중간 단계로 개념적 수준에서 물리적 수준으로의 연결 역할을 수행한다. SDTS 변환은 모듈들로 구성되며, 각 모듈은 하나 이상의 모듈 레코드들로 구성된다. 또한, 각 모듈 레코드는 하나 이상의 모듈 필드들로 구성되고, 각 필드는 하위 필드들로 구성된다.

SDTS는 34개 유형의 모듈들로 구성되며, 레코드, 필드, 하위 필드가 구성되는 방식을 상세하게 정의하고 있다. 34개의 모듈은 글로벌 모듈, 데이터 질 모듈, 공간 객체 모듈, 속성 모듈, 그리고 그래픽 표현 모듈 등 5개의 범주로 나뉘어 진다.

2.1.2 Part 2

데이터 변환 과정 중에서 공간 형상의

공통적인 정의가 필요하게 되며, 특히 공간 데이터 사회 (Spatial Data Community)에서의 데이터 공유 필요성은 더욱 강조되고 있다. SDTS의 Part 2는 표준화된 개체 리스트의 개발을 위한 표준화된 시도이다. Part 2의 리스트들은 서로 다른 기관에서 저마다 다른 요구에 의해 지리형상 약 2600개의 정의를 모두 고려할 수 있도록 수년간에 걸친 노력으로 이루어졌다. 현재 리스트들은 200개의 개체 형태 정의, 244개의 속성 정의, 1200개의 포함 용어(Included Term)들을 나타낸다.

2.1.3 Part 3

ISO 8211 화일은 Data Description File(DDF)로 불리며 두 가지 형태의 레코드인 DDR(Data Descriptive Record)과 DR(Data Record)을 가지고 있다. DDR은 데이터 구조와 데이터 구조 설명을 가지는 헤더 부분이며, DR은 실질적인 데이터를 가지는 데이터 부분이다. 하나의 화일은 한 개의 DDR과 한 개이상의 DR로 구성된다. DDF의 레코드는 한 개 이상의 필드로 구성되는데, 각 필드는 서브 필드들로 이루어지며 이들은 데이터의 기본적인 요소(element)들을 가진다.

2.1.4 Part 4

SDTS Part 4는 Part 1, 2, 3에서 정의되고 규정된 SDTS의 개념적, 논리적 부분을 물리적 수준에서 구체적으로 변환하는 방법을 기술하며, 특히 벡터 데이터의 규정 및 변환을 위한 내용이다. TVP는 지리 벡터 데이터를 플래너 그래프 위상과 함께 가지며 변환의 적합성, 인코더 및 디코더의 적합성 등의 규정을 갖는다. 또한, 공간 객체로 노트, 플래너, 완전무

결 체인, GT-다각형, 합성들이 필수적으로 필요하다. 공간 객체를 수반하는 모듈에 있어서 모듈명은 표준화된 4문자로 구성되고 마지막의 두 문자는 상이한 모듈/화일을 구분하는데 사용된다. 또한, 모듈은 실제 변환시 모듈의 역할, 포함되는 정보 등에 대한 설명, 모듈의 한계와 요구되는 내용, 그리고 구체적으로 ISO 8211 방식으로 변환될 때 Part 1에서 규정된 내용이 어떻게 변환되는지에 대한 내용 등을 포함한다 [8].

2.2 GOTHIC 분석

2.2.1 GOTHIC 데이터베이스

GOTHIC은 영국 Laser-Scan사에서 개발한 객체 지향 GIS 개발 도구로써 개발환경 내에 자체 객체 지향 데이터베이스를 두고 모든 공간 데이터와 비공간 데이터를 저장한다. GOTHIC에서는 실세계를 구성하는 개체들을 유일하면서 서로간에 구분할 수 있는 공간 객체(spatial object)들로 생각하고, 이들 중 유사성을 가진 공간 객체들을 그룹화하여 클래스로 정의한다. 즉, 하나의 공간 객체는 정의된 클래스에 대한 하나의 인스턴스(instance)이고, 그러기 위해서 하나의 공간 객체는 반드시 하나의 클래스에 소속되어 있어야 한다. 공간 객체들은 속성이라 불리는 데이터와 메소드를 캡슐화하고, 캡슐화된 항목들을 다른 공간 객체에서는 보이지 않게 하여 오직 메소드에 의해서만 호출된다 [5].

GOTHIC 데이터베이스에서는 논리적으로 관련있는 데이터들을 규정된 방법에 따라 여러 데이터 셋(dataset)으로 구성하

여 데이터베이스의 작업단위를 세분화하고 있다. 그리고, 데이터 셋들을 계층화하여 데이터 셋 단위로 데이터베이스 내의 데이터의 생성, 갱신, 삭제와 같은 일련의 트랜잭션을 수행하도록 하고 있다.

2.2.2 GOTHIC 버전 관리

만약 데이터가 갱신될 때는 갱신된 내용들은 해당 데이터 셋에서 바로 갱신되어 저장되지 않고 새로운 데이터 셋을 자동으로 생성하여 저장되며, 기존의 데이터 셋과 새로 생성된 데이터 셋은 부모-자식(parent-child) 관계가 설정된다. 이때 자식 데이터 셋은 부모 데이터 셋의 모든 데이터베이스 스키마를 상속하고, 새로 변경된 부분만을 저장한다. 이러한 GOTHIC의 버전 메커니즘은 자식 데이터 셋의 생성으로 생기는 부하(overload)를 줄여주고, 또한 장기 트랜잭션(long transaction)을 지원해 준다. 장기 트랜잭션이 다수의 사용자에게 의해 수행될 경우 한 데이터 셋을 기준으로 사용자마다 논리적 내용이 같은 여러 개의 자식 데이터 셋을 생성한다. 그리고, 각각의 사용자에게 자신만의 가지(branch) 데이터 셋을 지정하여 일련의 트랜잭션을 수행한다. 다수 사용자가 동시 접근할 수 있는 기능은 각각의 사용자에게 지정된 개인 버전에만 갱신 액세스 권한을 실질적으로 부여함으로써 가능하다.

같은 데이터 셋으로부터 출발한 버전은 논리적으로 같은 내용에서 출발하기 때문에 다수 사용자가 갱신 및 삭제를 수행할 수 있다. 그 이유로 장기 트랜잭션을 마치고 마스터 버전에 결합시 객체간의 충돌이 발생할 수 있는데 이를 방지하기 위해 장기 트랜잭션을 수행하기 전에

사용자마다 데이터 셋의 편집대상 부분을 저장된 클래스별 혹은 지역별로 겹치지 않게 설정해 준다.

2.2.3 GOTHIC 클래스

GOTHIC에서 클래스는 공간 객체의 표현 단위이다. 모든 공간 객체들은 그들이 표현되어야 할 데이터 포맷에 따라 다음의 여러 클래스 중의 하나에 속해야 한다. GOTHIC 기본 클래스의 상속관계를 살펴보면 그림 1과 같이 나타낼 수 있다.

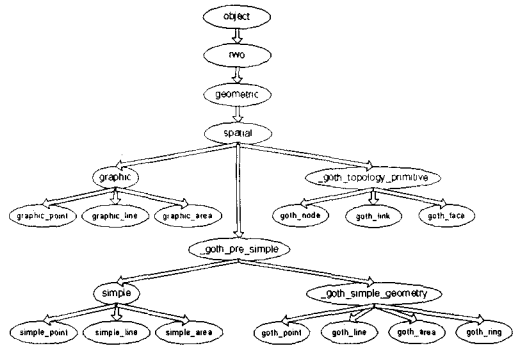


그림 1 GOTHIC 기본 클래스 계층구조

그림 1에서 최상위의 4개의 클래스인 object, rwo, geometry, spatial 클래스는 하위의 모든 클래스에 상속됨을 알 수 있고, 그 4개의 클래스들로서 모든 공간 객체들이 모두 포함되는 일반적인 값을 정의한다. 4개의 상위 클래스들은 각 객체에 대한 설명, 장기 트랜잭션이 끝났을 때 통합(merging)을 위한 메소드, 클래스들의 기하(geometry)정보, 기하정보를 얻고 저장하는 메소드 등을 정의한다. 하위의 클래스들은 점, 선, 영역들에 관한 실제 공간 객체가 인스턴스로 설정되는 클래스들이다.

graphic 클래스는 위상 구조를 고려하

지 않는 점, 선 면의 표현에 대한 값들의 타입을 정의한다. 점, 선, 면의 표현에 각각 `graphic_point` 클래스, `graphic_line` 클래스, `graphic_area` 클래스가 상속되어 있고, 이 클래스들에 해당하는 공간 객체들은 메모리와 저장공간을 적게 요구한다.

`simple` 클래스는 위상 구조의 기하정보를 갖는 공간 객체를 구성하기 위해 사용되는 클래스이다. 이 클래스는 `_goth_pre_simple` 클래스의 값을 상속받으며, 그 하위에 점, 선, 면에 대한 공간 객체를 표현하는 `simple_point` 클래스, `simple_line` 클래스, `simple_area` 클래스가 상속된다.

`_goth_topology_primitive` 클래스는 공간 객체를 표현함에 있어 각각의 위상 관계를 체계적으로 표현하는데 이용되는 클래스이다. `goth_node` 클래스는 위상 구조를 갖는 점을 표현할 수 있는데, 이 위상 규칙이 적용되어 선들이 교차하는 부분 등에 점이 새로 생성되기도 한다. `goth_link` 클래스는 두개의 노드를 연결하여 선을 표현하기 위한 클래스이며, `goth_face` 클래스는 선이나 영역으로 구분되는 면을 표현하기 위한 클래스이다.

`_goth_simple_geometry` 클래스는 `_goth_pre_simple` 클래스에서 상속되며 위상 구조의 공간 객체에 대한 기하정보 값을 가지고 있다. 그 하위에 다음의 네 가지 클래스가 상속되어 있는데 각각 점, 선, 영역, 링을 표현하기 위한 `goth_point` 클래스, `goth_line` 클래스, `goth_area` 클래스, `goth_ring` 클래스이다. 여기서 점은 하나의 노드 객체로 표현되고, 선은 두 개 이상의 노드와 노드의 수보다 하나 적

은 링크로 이루어진다. 영역은 링크와 노드로 구성된 닫힌 루프로 표현되는 내부를 나타낸다. 링크와 노드로 구성된 경계 선을 링이라 한다 [6,7].

3. SDTS 변환 시스템 설계

3.1 시스템 구성

본 논문에서 제시하는 SDTS 변환 시스템의 전체적인 구성은 그림 2와 같다. SDTS 변환 시스템은 변환 데이터 인코딩, 디코딩 처리 모듈과 SDTS 인코딩, 디코딩 처리 모듈, 상호 변환 처리 모듈, 로그 파일 처리 모듈, 에러 처리 모듈, 변환 데이터 물리적 매핑 모듈, 그리고 SDTS 물리적 매핑 모듈로 구성된다.

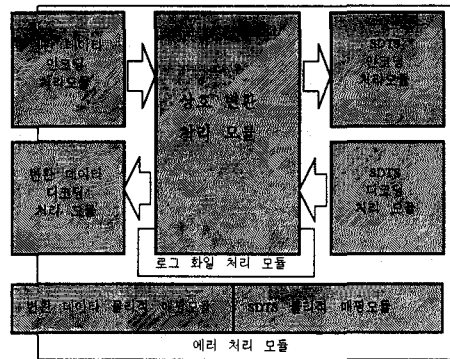


그림 2 SDTS 변환시스템 구성

그림 2는 변환할 데이터 포맷과 SDTS로의 상호 변환을 수행하는 SDTS 변환 시스템의 설계를 보여 준다. 변환 데이터 인코딩 처리 모듈에서 상호 변환 처리 모듈을 통해 SDTS 인코딩 처리 모듈과 SDTS 매핑 모듈을 수행하는 과정을 인코딩 과정이라 하며, 그 반대의 화살표 과정은 디코딩 과정이라 한다. 인코딩과 디코딩시 상호 변환 처리 모듈은 변

환에 필요한 규칙과 조건에 맞는 적합한 변환을 수행하며, 상호 변환에 필요한 게이트웨이 함수들의 인터페이스로 이루어진다 [15].

각각의 구성모듈에 대한 상세한 설명은 다음과 같다.

- (1) 상호 변환 처리 모듈: 게이트웨이 함수들로 구성되며 변환 방향에 따라 상호 변환 처리 모듈을 호출한다.
- (2) 변환 데이터 인코딩, 디코딩 모듈: 변환 데이터 물리적 매핑 모듈을 통해 얻은 논리적 데이터를 가지고 인코딩 및 디코딩 작업을 수행한다.
- (3) SDTS 인코딩, 디코딩 처리 모듈: SDTS 물리적 매핑 모듈을 통해 얻은 논리적 데이터인 각 모듈들에 대한 인코딩, 디코딩 작업을 수행한다.
- (4) 변환 데이터 물리적 매핑 모듈: 변환 데이터 물리적 매핑 모듈은 화일이나 타입 등의 미디어로 저장 및 읽기를 수행하는 물리적인 매핑을 수행한다.
- (5) SDTS 물리적 매핑 모듈: SDTS 물리적 매핑 모듈은 SDTS TOOLKIT을 이용하여 구현되며, SDTS TOOLKIT은 FIPS123 라이브러리, SDTS++ 등이 있다. 이러한 SDTS TOOLKIT은 SDTS PART3의 물리적 계층을 구현하는 규칙에 대한 함수들을 가지고 있다.

3.2 SDTS 변환 시스템 구현 과정

어떤 변환 시스템이든지 사용자가 가능한 연산에 관한 중재의 범위를 줄여서 전 과정을 자동화하는데 그 가장 큰 목적이 있다. 변환 시스템에 대한 중재의 범위는 사용자의 변환 목적 시스템에서 사

용되는 데이터의 목록과 변환 시스템에서 실행시킬 특수한 옵션의 양에 따라 결정된다. SDTS 변환 시스템의 일반적인 개발 과정은 그림 3과 같다.

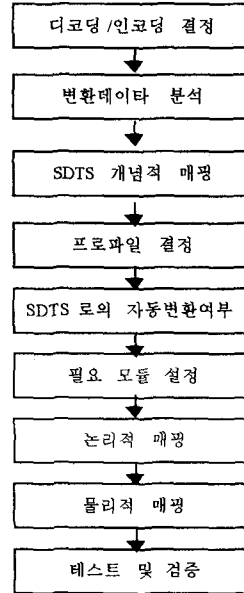


그림 3 매핑 과정

- (1) 개발할 변환 시스템이 SDTS에 대한 인코딩만 할 것인지, 디코딩만 할 것인지 또는 양방향 변환을 모두 할 것인지 결정한다.
- (2) 인코딩과 디코딩을 위한 사용자 데이터 포맷 및 특성을 파악하며 그에 따른 개념적 모델, 논리적 화일 포맷, 물리적 화일 포맷에 대해 상세히 분석한다.
- (3) SDTS와 사용자 데이터 포맷간의 변환의 기본 구조를 파악하는 개념적인 매핑을 수행한다. 이 때 상호 변환 데이터 포맷간의 일치성 등을 파악한다.
- (4) 변환 시스템에 꼭 적용시켜야 하는 적절한 SDTS 프로파일을 결정한다. 변환 시스템은 프로파일에서의 요구사항을 따라 구현되어야 한다.
- (5) SDTS 모듈에 의해서 요구되는 목적 시스템에서의 데이터 요소를 인식하고 규정짓는다. 이 때 상호 데이터 포맷의 변환시 자동적으로 변환될 것인지 사용자의 중재가 필요한지가 주로 결정된다. 만약 포맷의 일치도가 높다면 좀더 자동화가 가능하다.
- (6) 구현할 옵션을 결정한다. 옵션은 프로

파일과 SDTS 기본 명세서에서 언급된 꼭 필요한 옵션과 선택적으로 필요한 옵션을 말하며 그 중에서 실제 구현할 내용을 결정하는 것이다.

- (7) SDTS 모듈에서 또는 SDTS 모듈로의 논리적 매핑을 디자인한다. 논리적 매핑은 상호 변환할 데이터의 포맷 등이 서로 어떻게 매핑되어야 할 것인가에 대한 선택을 의미한다.
- (8) ISO 8211 화일에서 또는 ISO 8211 화일로의 물리적인 매핑을 디자인한다. ISO 8211 화일은 물리적 변환 포맷의 기본 화일이다. SDTS 변환 작업에 대해 ISO 8211 인코딩 프로세스내에서도 도움을 주기 위한 소프트웨어를 제공한다.
- (9) 실제 SDTS 변환 시스템을 완성하고 데이터를 이용하여 테스트한다.
- (10) 데이터 변환 후 변환이 정상적으로 완료되었는지 검증한다.

이상이 SDTS 변환 시스템의 개발에 필요한 일반적인 작업 과정이다. 위의 과정의 순서에 맞게 작업하는 것이 바람직하지만 경우에 따라 다시 위의 과정을 반복 수행하는 것도 필요하다.

3.3 SDTS 변환 규칙

SDTS 변환 작업에는 공통으로 적용될 모듈과 사용에 대한 규칙이 있다. 변환할 데이터의 특성에 따라 SDTS 모듈들은 포함되거나 제거될 수 있는데, 다음의 규칙에 적용되는 모듈은 반드시 변환되어야 한다. 그러나, 특정 프로파일 사용시에는 SDTS 모듈들의 사용에 대한 정책 혹은 추가적인 제한을 정의할 수 있다.

- (1) 물리적인 매체에 따라 순서가 중요한 경우(e.g., 테이프 장치)에는 Catalog 모듈과 Identification 모듈은 맨 처음에 위치해야 한다.
- (2) 글로벌 모듈 중의 3개(Identification, Internal Specification, External Specification) 모듈들은 항상 존재하여야 하며 나머지 모듈들은 필요에 의해 추가될 수 있다.
- (3) 변환에서 속성 모듈이 존재한다면, Data Dictionary, Data Domain, Data Schema 모듈은 반드시 존재해야 된다. 이 경우 SDTS Part 2에 정의된 속성이라면 Data Dictionary Domain, Data Define 모듈은 생략해도 된다.
- (4) 5개의 질 모듈(Lineage, Positional Accuracy, Attribute Accuracy, Logical Consistency, Completeness)은 반드시 존재해야 된다.
- (5) 속성 모듈(Attribute Primary, Attribute Second Module)은 Schema Module과 연관되어야 한다.

3.4 프로파일 개발과 갱신

실제 공간 데이터들의 변환을 위한 SDTS의 변환은 프로파일을 통해 수행된다. 프로파일은 SDTS 변환 작업을 위한 특정 변환 규칙을 제공하며, SDTS 명세서에서 특정한 공간 객체형의 사용을 위한 제한과 요구사항, 기본 명세서에서 표현된 옵션들 간의 선택을 위한 규칙을 포함하거나, SDTS 모듈의 사용을 위한 제한점과 요구 사항을 가지며 모듈의 명칭과 화일 이름의 명칭변환, 그리고 ISO 8211인코딩 명세의 사용 등을 정의한다. 프로파일은 SDTS 명세서의 부분 집합이다. 프로파일의 한 예로 TVP를 들 수 있

는데, TVP는 FIPS173에서 SDTS Part 4로 NIST의 형식 승인되었다. 다른 한 예로 Raster Profile(RP)가 있다. RP는 래스터 모듈에 관한 프로파일이지만 아직 형식 승인되지는 못했다.

SDTS Task Force에서는 사용에 필요한 새 프로파일의 개발과 현존하는 프로파일의 갱신을 허용한다. 프로파일 개발 또는 갱신을 위한 일반적인 과정을 보면 다음과 같다.

먼저 프로파일의 적절한 조사와 계획을 세워 특성을 정의한 후 프로파일에 의해 다루어지는 데이터의 명확한 개념적 모델을 개발하고, 특정 GIS 포맷과 SDTS내의 객체 타입간에 개념적 매핑을 설계한다. 다음으로 프로파일에 있는 객체 타입 또는 속성 이름이 강제적인지, 선택적인지를 정의하고 데이터 질 모듈에 따르기 위한 프로파일의 특정 규칙이나 요구사항 등을 정의한다. 그리고, 프로파일을 위한 일반적인 사항, 즉 SDTS 모듈의 조직 및 내용에 대한 규칙, 모듈 필드와 서브 필드, 모듈간의 관계성, 공간 주소 포맷에 대한 정보 등을 정의한다. 그 후 논리적 매핑을 설계하고 모듈의 요구사항 및 제한 등을 정의함으로써 SDTS 모듈의 포맷과 내용에 대한 완전한 정의를 내리고, 각 모듈의 핵심과 부속 옵션, 특정 요구와 제한 등의 특성을 기술한다. 마지막으로 ISO 8211 물리적 매핑에 대한 요구사항을 정의함으로써 프로파일에 대한 개발 및 갱신을 마치게 된다.

4. SDTS 변환 시스템 구현

본 장에서는 3 장의 범용 SDTS 변환

시스템의 설계 과정을 실제 변환 데이터인 GOTHIC 데이터 포맷을 사용하여 구현한 사항에 대해 언급한다. SDTS와 GOTHIC간의 상호 변환 시스템은 GOTHIC 데이터 셋 중에서 지리 데이터 유형을 변환한다.

4.1 GOTHIC 변환 모듈의 구성

SDTS와 GOTHIC 간의 상호 변환에서 GOTHIC 공간 데이터 분석시 가장 먼저 관심을 가져야 하는 부분은 교환할 GOTHIC 공간 데이터가 위상구조인가 비위상구조인가 하는 부분이다. SDTS와 GOTHIC 시스템 모두 위상/비위상 구조를 제공하고 있으므로 각각 클래스와 모듈의 설정을 위상구조 데이터일 때와 비위상구조 데이터일 때로 분리한다. 그림 4는 GOTHIC 클래스와 이에 대응되는 SDTS 모듈을 보여 준다.

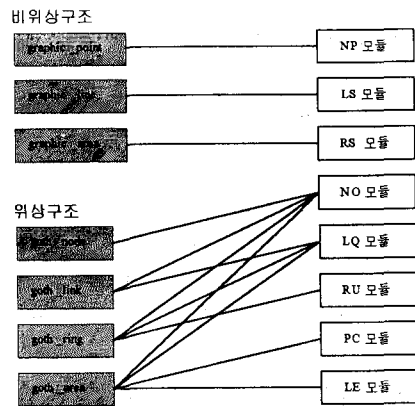


그림 4 상호 변환 모듈

GOTHIC에서는 비위상구조를 가지는 클래스가 명확히 구분되는데, 그림 1에서 보인 GOTHIC 기본 클래스 중 Graphic

클래스에서 상속되는 `graphic_point`, `graphic_line`, `graphic_area` 클래스이다. 점, 선, 영역을 나타내는 이들 하위 클래스들은 각각 SDTS 모듈 NP(노드-점), LS(라인 스트링), RS(링 스트링) 모듈로 변환된다.

위상구조를 가지는 공간 객체를 표현하기 위한 GOTHIC의 클래스는 `goth_node`, `goth_link`, `goth_ring`, `goth_area` 클래스이다. 이때 `goth_node` 클래스는 SDTS의 NO 모듈로 일대일 변환되며, `goth_link` 클래스는 변환 시에 NO(노드), LQ(링크) 두 모듈로 변환된다. `goth_ring` 클래스는 `node`와 `link`로 만들어진 SDTS의 RU(GT-ring) 모듈과 상호 변환될 수 있으며, 선택된 내부 영역을 나타내는 `goth_area` 클래스는 SDTS 모듈 중에 점, 선, 면을 나타내는 NO, LQ, PC(폴리곤-체인), LE(체인) 모듈과 상호 변환된다.

4.2 GOTHIC 변환 모듈의 구현

SDTS 변환 시스템의 개발 단계를 살펴보면, GOTHIC과의 인코딩 과정만을 구현하는 변환 시스템을 제작하며 (단계 1), 이미 GOTHIC의 분석을 관련연구에서 보였다 (단계 2). GOTHIC은 데이터셋이나 버전 내에서 실세계 공간 객체가 기록되는데, 공간 데이터 보관을 위한 구조는 공간 객체, 값, 공간 객체 클래스의 구조를 가진다. GOTHIC의 데이터와 SDTS간의 개념적 매핑은 표 1과 같다 (단계 3).

GOTHIC은 위상 구조인 `_goth_topology_primitive` 클래스를 가지며 `Node`,

<표 1> GOTHIC과 SDTS간 개념적 매핑

	GOTHIC	TOPAS	SDTS모듈
비위상 구조	point	point	NP
	line	string	LS
	area	G-ring	RS
위상 구조	node	node	NO
	link	node, link	NO, LQ
	area	node, link, area-point, complete-chain	NO, LQ, PC, LE
	ring	node, link, GT-ring	NO, LQ, RU

Link, Face의 공간 객체를 다루므로 SDTS의 TVP에 따라 변환한다 (단계 4). SDTS와 GOTHIC 간의 개념적 매핑을 통해 Point, Node, Line, Link로의 공간 객체 변환은 자동화가 가능하나, 클래스의 변환은 객체 지향 기반의 GOTHIC과 SDTS 간의 관계에서 별도의 매핑을 필요로 한다 (단계 5). GOTHIC과 SDTS간의 필요 모듈을 살펴보면 GOTHIC은 TVP의 프로파일에 준하게 변환되어야 한다. 표 2는 GOTHIC에서 SDTS로의 변환 모듈을 보여준다 (단계 6).

GOTHIC에서 SDTS 논리적 매핑을 위해 단계 6에서의 SDTS 논리적 계층인 모듈들이 제시되었다. GOTHIC의 객체 클래스에서 공간 객체는 속성, 메소드, 객체 참조 포인터로 구성되어 있다. 각 객체에 해당하는 속성은 AP(Attribute Primary)모듈로 매핑되며, 공간 객체는 SDTS 공간 객체 모듈로 매핑된다.

위상 클래스는 각각의 SDTS 모듈로 매핑이 이루어진다 (단계 7). GOTHIC은 다양한 지리 정보 시스템 개발을 위한 객체 데이터베이스와 다양한 툴킷을 제공한다. 이러한 툴킷은 기본 공간 객체에 대

<표 2> 데이터간 변환 모듈

Global 모듈	Identification, Catalog/Directory, Catalog/Spatial Domain, Spatial Reference Internal, Spatial Reference External, Data Dictionary/Definition, Data Directory Domain, Data Dictionary Schema
Data Quality 모듈	Lineage, Positional Accuracy, Attribute Accuracy, Logical Consistency, Completeness
속성 모듈	Attribute Primary, Attribute Secondary
공간 객체 모듈	Point, Node, Line-String, Complete-Chain, GT-Polygon, Link, Ring, Raster Definition, Raster Cell

한 작업을 위해 사용되며, SDTS Part3의 물리적 계층과의 매핑을 위하여 FIPS173 Library를 사용하여 ISO 8211 포맷으로 저장된다 (단계 8). 완성된 변환 시스템의 테스트 및 검증은 5장에서 설명한다.

5. 테스트 및 검증

변환이 끝나면 변환된 데이터가 원본 데이터에 대해 정보의 손실이 있는지 및 에러의 유무 등을 확인하는 테스트 과정을 거친다 (단계 9). 이 단계에서는 많은 데이터에 대한 테스트를 수행하여 변환 시스템의 문제점을 찾아내고 만약 에러가 발견된 경우 어떤 단계에서 잘못된 매핑이 수행되었는지를 찾아 수정을 하게 된다.

테스트가 끝난 후 SDTS의 실제적 사용은 SDTS 변환 시스템이 SDTS 요구 사항에 따라 만들어졌는지를 검증하는 일치성 검사(Conformance Test) 과정을 필요로 한다. 일치성 검사는 NIST (National Institute of Standards and Technology)에서 수행하며, SDTS 트랜

스퍼, SDTS 인코더, SDTS 디코더의 세 가지 검사가 이루어진다 (단계 10) [9,12].

SDTS 트랜스퍼는 서로 다른 공간 컴퓨터 시스템간의 데이터를 교환하기 위해 사용되어지는 데이터 셋이며 소프트웨어에 의한 자동적 검사가 수행될 수 있다. 그러나, SDTS 인코더나 디코더는 원본과 목적 데이터 모델을 비교하고 검사해야 하는 것을 포함하므로 더 복잡하다. 그러므로, 인코더나 디코더에 대한 일치성 검사는 수동적 검사를 수행한다. 인코더나 디코더에 대한 일치성 검사를 하려면 개발자는 검사 항목을 평가하는데 필요한 정보를 제공하기 위해서 원본과 목적 데이터 모델에 대한 질의사항들을 채워 놓아야 한다.

인코더에 대한 정보를 제공하는 질의 사항에는 입력 데이터 포맷, 강제적 (mandatory) 혹은 선택적(optional) 공간 객체와 그에 대응되는 객체 타입, 두 포맷에서 공간 객체에 대응되는 리스트, 각 SDTS 데이터 성분에 대한 정보를 가지고 있는 원본 리스트, 프로파일로 바뀔 수 없는 입력 포맷에 대한 데이터, 속성 데이터들이 어떻게 저장되는지와 공간 객체에 어떻게 연결되는지에 대한 정보 등이 들어 있다.

디코더에 대한 정보를 제공하는 질의 사항에는 출력 데이터 포맷, 강제적 혹은 선택적 공간 객체와 그에 대응되는 객체 타입, 어떤 프로파일 옵션이 구현되는지 나타내기 위한 옵션의 체크 리스트, 변환에 대한 일반적인 기술을 포함하여 SDTS 공간 주소가 출력 포맷으로 바뀌는 방법, 속성 데이터들이 어떻게 저장되

는지와 공간 객체에 어떻게 연결되는지에 대한 기술 등이 들어 있다.

참고문헌

NIST에서의 일치성 검사는 일반적으로 TMCEC(Test Method Control Executive Committee)에서 감독한다.

6. 결론

본 논문에서는 GIS 데이터의 국가 공통 데이터 교환 포맷으로 채택된 SDTS를 분석하여 임의의 공간 데이터에서 SDTS로의 상호 변환 시스템을 위한 설계, 구현, 테스트 및 검증하는 방법을 제시하였다. 또한, 본 논문에서 제시한 범용의 SDTS 변환 시스템의 설계 및 방법을 가지고 실제 공간 데이터인 GOTHIC과 SDTS 변환 시스템의 구현을 예로 들었다. 그리고, 각 지리 데이터들의 논리적 설계를 설명하였다. SDTS 변환 시스템의 물리적 설계에서는 SDTS 라이브러리 TOOL KIT인 FIPS 123 라이브러리를 분석 및 사용하였다.

현재 지리 정보 시스템은 국가적인 중요성을 인정받아 활발히 연구 개발이 진행되는 분야이며, 매우 다양한 데이터 포맷과 적용 분야를 가지고 있다. 이들간의 효율적인 데이터의 상호 변환은 지리 정보 시스템의 분야의 큰 핵심 기술이면서 반드시 실현되어야 할 부분이다. 본 SDTS 변환 시스템 개발은 상호 변환 시스템 제작을 위한 설계 및 구현과 테스트를 실제 적용되는 근간을 바탕으로 연구되었다. 이 후 연구 과제로는 SDTS 변환에 관한 상세한 모듈화에 중점을 두어 보다 완벽하게 모듈별 상호 변환하는 방법에 대한 연구가 필요하겠다.

- [1] Altheide, P., "An Implementation Strategy for SDTS Encoding", American Congress on Surveying and Mapping, Vol.19, No.5, Dec. 1992, pp.306-310.
- [2] Altheide, P., "Design of a Spatial Data Transfer Processor", Cartography and Geographic Information System, Vol.19, No.5, Dec. 1992, pp.311-314.
- [3] Davis, B.A., George, J.R., and Marx, R.W., "TIGER/SDTS: Standardizing an Innovation", American Congress on Surveying and Mapping, Vol.19, No.5, Dec. 1992.
- [4] Gunther, G., and Buchemann, A., "Research Issues in Spatial Databases", ACM SIGMOD Record, Vol.19, No.4, Dec. 1988, pp.61-68.
- [5] Jon Billing, *Gothic KGIS OODB Consultation*, Aug. 1996.
- [6] Laser-Scan, *Writing and Developing Applications using GOTHIC ADE*, Ver2.0, 1995.
- [7] Laser-Scan, *GOTHIC Module Reference Manual*, Ver2.0, 1994.
- [8] Lazar, R., "The SDTS Topological Vector Profile", Cartography and Geographic Information System, Vol.19, No.5, Dec. 1992, pp.296-299.
- [9] Lazar, R., "Conformance Testing for the Spatial Data Transfer Standard", Cartography and Geographic Information System, Vol.21, No.3, Jul. 1994, pp.159-161.

[10] Medeiros, C.B., and Pires, F., "Databases for GIS", Proc. of ACM SIGMOD Int. Conf., Mar. 1994, pp. 107-115.

[11] National Institute of Standards and Technology, *The Spatial Data Transfer Standard, Federal Information Processing Standard Publication 173*, U.S. Department of Commerce, 1992.

[12] Williams, M.G., "Conversion of a U.S. Geological Survey DLG-3 Data Set to the SDTS Topology Vector Profile", *Cartography and Geographic Information System*, Vol.19, No.5, Dec. 1992, pp.315-320.

[13] 오병우, 한기준, "지리 정보 시스템을 위한 표준화", 한국정보과학회 정보과학회지, 제13권 10호, 1995, pp.46-55.

[14] 오병우, 이강준, 한기준, "SDTS와 Informap간의 데이터 변환 시스템의 설계 및 구현", 한국지형공간정보학회 논문지, 제4권2호, Dec. 1996, pp.109-121.

[15] 정선영, 최신영, 서재화, 이기준, "공간 객체 저장 시스템 GeoStore/Shore와 SDTS 간의 공간 데이터 변환방법에 관한 연구", 한국정보과학회 학술발표논문집 제24권2호, Oct. 1997, pp.245-248.

[16] 한국통신 선로기술연구소, *GIS DB 용 데이터 포맷 변환 도구 개발*, 최종보고서, Dec. 1995.

필자 약력

이 강 준

1995년 건국대학교 전자계산학과 졸업 (공학사)
 1997년 건국대학교 전자계산학과 졸업 (공학석사)
 1997년 ~현재 건국대학교 대학원 컴퓨터공학과 박사 과정
 관심 분야 : 지리 정보 시스템, 객체 관계 데이터베이스, 공간 데이터 마이닝, 실시간 데이터베이스

김 준 종

1998년 건국대학교 전자계산학과 졸업 (공학사)
 1998년 ~현재 건국대학교 대학원 컴퓨터공학과 석사 과정
 관심 분야 : 지리 정보 시스템, 객체 지향 데이터베이스, 공간 데이터 마이닝

설 영 민

1998년 건국대학교 전자계산학과 졸업 (공학사)
 1998년 ~현재 건국대학교 대학원 컴퓨터공학과 석사 과정
 관심 분야 : 지리 정보 시스템, 실시간 데이터베이스, 웹 데이터베이스

한 기 준

1979년 서울대학교 수학교육학과 졸업 (이학사)
 1981년 한국과학기술원 전산학과 졸업 (공학석사)
 1985년 한국과학기술원 전산학과 졸업 (공학박사)
 1990년 Stanford 대학 전산학과 visiting scholar
 1985년 ~현재 건국대학교 컴퓨터공학과 교수
 관심 분야 : 지리 정보 시스템, 객체 지향 데이터베이스, 공간 데이터 마이닝, 주기억-상주 데이터베이스