

# 전자해도용 XML 스키마의 정의 및 변환

이성대\* · 강형석\* · 박휴찬\*\*

## XML Schema Definition and Translation of Electronic Navigational Chart

Seong-dae Lee\* · Hyung-suk Kang\* · Huy-chan Park\*\*

이 논문은 한국해양대학교 BK21 사업단의 지원을 받았음.

### 요 약

전자해도는 해안선, 수심, 항해 표시를 포함하고 있는 정교한 전자 지도이다. 또한 고급의 문서적, 공간적, 지리적인 정보를 포함하고 있으며, 빠르게 전통적인 종이지도를 대체하고 있다. 그러나 전자해도가 항해에 성공적으로 적용되었지만, S-57이라는 특정 데이터 형식은 전용의 시스템을 필요로 한다. 또한 전자해도는 특정 영역이나 제한된 전문가들만 사용하고 있다. 전자해도의 이러한 한계들을 극복하기 위해서는 XML과 같은 범용의 데이터 형식이 필요하다. XML을 사용하여 전자해도를 저장한다면 인터넷을 통하여 쉬운 접근이 가능하며, 더욱 다양한 사용자와 응용을 위해 사용될 수 있을 것이다.

본 논문에서는 전자해도의 데이터 형식인 S-57을 저장할 수 있는 XML스키마를 제안하고, S-57 전자해도를 제안된 XML 스키마를 따르는 XML 문서로 변환하는 프로그램을 개발한다.

### ABSTRACT

Electronic Navigational Chart is a sophisticated digital chart which contains navigational information such as coastline, depth of water, and nautical mark. It contains high levels of textual, spatial and graphical data, and rapidly replaces traditional paper charts. Although Electronic Navigational Chart has been successfully applied to the navigation of ships, the specific data format, S-57, requires some specialized systems. Furthermore its usage may be limited to specific domains and experts with restrictions. To overcome these limitations resulting from the specific data format, S-57, of Electronic Navigational Chart, more general data format such as XML will be preferred. If Electronic Navigational Chart is transformed into the form of XML, it can be easily accessed and exchanged on the internet. Therefore it may be used for more users and applications. In this paper, we propose an XML Schema to equivalently represent XML for the S-57 format of Electronic Navigational Chart and develop a program translating the S-57 Electronic Navigational Chart to XML document according to the proposed XML Schema.

### 키워드

전자해도, Electronic Navigational Chart, S-57, XML

### 1. 서론

전자해도(ENC: Electronic Navigational Chart)는 종이 해도에 나타나는 해안선, 수심, 등대, 항

로표지, 항로 등 선박의 항해와 관련된 모든 해도 정보를 국제수로기구(IHO: International Hydrographic Organization)의 표준 규격인 S-57에 따라 제작된 디지털 해도이다[1,2]. 전자 해도의

\* 한국해양대학교 대학원 컴퓨터공학과  
접수일자 : 2003. 12. 15

\*\* 한국해양대학교 기계·정보공학부

NODE, EDGE, FACE의 3가지이며, 실제 데이터에서는 점(Point), 선(Line), 면(Area)으로 표현된다. 점은 위도와 경도 좌표를 가지는 한 지점의 위치 정보이며, 선은 점과 점의 연결, 면은 점들의 연결로 구성된 다각형 공간으로 표현된다[4].

2.1.3 객체 식별자와 어트리뷰트

S-57 전자해도 표준 형식에는 각 객체 클래스들을 위하여 식별자와 여러 가지 어트리뷰트들을 제공한다. 식별자는 해당 객체 클래스가 어떤 객체인지에 관한 정보를 가지고 있으며, 어트리뷰트들은 해당 객체가 어떤 정보를 가질 수 있는지 결정한다. 객체 식별자의 종류와 의미는 표 2와 같다[4]. 표 2의 'LIGHTS' 객체 클래스는 식별자가 어떤 값을 가지는지를 보여준다. 예를 보면 'LIGHTS' 객체 클래스는 75(OBJL)번 객체 클래스이며, 같은 'LIGHTS' 그룹에서 2(GRUP)번째 특징 객체이며, 레코드 버전은 1(RVER), 발행 기관은 550(AGEN), 특징 객체 식별 번호는 234260030(FIND), 특징 객체 식별 상세번호는 1315(FIDS)라는 것을 알 수 있다.

표 2. 식별자의 종류와 의미  
Table. 2 Types and Meaning of Identifiers

식별자 종류	식별자 의미	'LIGHTS' 객체 클래스의 예
GRUP	그룹	GRUP="2"
OBJL	객체 클래스의 산술 코드/라벨	OBJL="75"
RVER	기록 버전	RVER="1"
AGEN	발행 기관	AGEN="550"
FIDN	특징 객체 식별번호	FIDN="234260030"
FIDS	특징 객체 식별 상세번호	FIDS="1315"

S-57 전자해도 표준 형식의 어트리뷰트들은 종류에 따라서 3가지로 구분되며, 사용빈도에 따라서 역시 3가지로 구분된다. 먼저 의미에 따라서 구분하면 표 3과 같다. '특징 객체 어트리뷰트'들은 특징 객체에 관련된 어트리뷰트이며 '국가 언어 어트리뷰트'들은 국가 언어와 관련된 어트리뷰트, 그리고 '공간과 메타 객체 어트리뷰트'들은 공간과 메타 객체에 관련된 어트리뷰트이다.

표 3. 종류에 따른 어트리뷰트 분류  
Table. 3 Attributes Classification According to Type

어트리뷰트 집합의 종류	어트리뷰트 집합의 의미	예
특징 객체 어트리뷰트	특징 객체에 관련된 어트리뷰트	색깔 : COLOUR
국가 언어 어트리뷰트	국가 언어와 관련된 어트리뷰트	국가언어정보 : NINFOM
공간과 메타 객체 어트리뷰트	공간과 메타 객체에 관련된 어트리뷰트	위치정확도 : POSACC

어트리뷰트를 사용빈도에 따라 구분하면 표 4와 같이 3가지 부분 집합으로 구분된다[4]. SubSet Attribute\_A는 객체 개별적 특징을 정의하는 어트리뷰트 집합이며, SubSet Attribute\_B는 최대, 최소 해상도나 국가 언어 정보와 같이 전자해도를 표현하거나 사용하는데 관련된 어트리뷰트 집합이다. SubSet Attribute\_C는 객체와 그 객체를 기술하는 데이터에 대한 관리 정보를 제공하는 어트리뷰트 집합이다[4].

표 4. 사용빈도에 따른 어트리뷰트 분류  
Table. 4 Attributes Classification According to Usage Frequency

어트리뷰트 집합의 종류	어트리뷰트 집합의 의미	예
SubSet Attribute_A	객체의 개별적 특징 정의	색깔 : COLOUR
SubSet Attribute_B	데이터의 사용에 관련된 정보	최대해상도 : SCAMAX
SubSet Attribute_C	객체와 데이터에 대한 관리 정보	기록 날짜 : RECDAT

2.2 XML

차세대 웹 문서 형식으로 부각하는 XML은 W3C에서 제안된 국제 표준의 전자문서 메타 언어이다. XML은 웹에서 구조화된 문서의 전송이 가능하도록 설계된 표준화된 텍스트 형식으로, 문서를 구성하는 각 엘리먼트들의 독립성을 보장하게 함으로써 문서의 호환성, 내용의 독립성, 엘리먼트 변경의 용이성 등의 특성을 제공한다[6,7].

2.2.1 XML 문서

XML은 HTML같은 고정된 태그를 가진 형식이 아니라 사용자가 직접 태그를 정의할 수 있는

확장 가능한 마크업 언어이다. 또한, SGML-(Standard Generalized Markup Language)에 기반을 두고 있으며, 단순하고 매우 유연성있는 텍스트 형식을 지원할 수 있다. XML은 표 5와 같이 엘리먼트, 어트리뷰트, 엔티티, 노테이션, 주석, 처리지시문, CDATA 섹션 등과 같은 구성 요소들로 이루어진다.

표 5. XML 문서의 요소와 의미  
Table. 5 Elements and Meaning of XML Document

XML 문서의 구성요소	각 구성요소의 의미
엘리먼트	문서 기술에 필요한 태그
어트리뷰트	엘리먼트 속성
엔티티	특별한 문자 삽입
노테이션	외부 이진 데이터를 가르키는 선언
주석	사용자가 읽기위해 사용되며 파싱하지 않음
처리지시문	응용에 전달되는 지시문
CDATA 섹션	파서가 마크업에 대해 해석하지 않는 부분

그림 5는 우편에 관련된 정보를 다루기 위해서 <Post> 엘리먼트와 그 부엘리먼트로 <Name> 엘리먼트와 <Address> 엘리먼트로 구조화된 XML 문서의 예이다[6,8,12].

```
<?xml version="1.0"?>
<Post>
  <Name>
    <FirstName>Raymond</FirstName>
    <MiddleInitial>G</MiddleInitial>
    <LastName>Bayliss</LastName>
  </Name>
  <Address>
    <Street1>10 Elizabeth Place</Street1>
    <Town>Paddington</Town>
    <City>Sydney</City>
    <StateProvinceCounty>NSW</StateProvinceCounty>
    <Country>Australia</Country>
    <ZipPostCode>2021</ZipPostCode>
  </Address>
</Post>
```

그림 5. XML 문서의 예  
Fig. 5 An Example of XML Document

### 2.2.2 XML 스키마

XML 문서는 적격 문서(Well-Formed Document)와 유효한 문서(Valid Document)로 나뉜다. 적격 문서란 XML 문서의 모든 엘리먼트와 어

리뷰트들이 XML의 표준 문법 규칙을 위반하지 않으면서 스키마가 없는 문서를 말하고, 유효한 문서란 XML 문서가 스키마의 정의대로 올바르게 작성되고 검증용 파서(Parser)로 파싱할 때 오류가 없는 문서를 말한다.

W3C에서 XML 1.0 권고안을 채택할 때, XML 문서에 허용되는 구성 엘리먼트를 제한하는 메커니즘을 포함시켰는데, 그것이 XML DTD (Document Type Definitions)이다[6]. 그러나 XML 개발자들의 DTD를 구현한 경험에 의해 DTD의 수많은 단점이 표출되었고, 이를 보완하기 위하여 내용 모델을 정의하는 더욱 포괄적이고 엄격한 방법이 필요하다는 것을 알게 되었다. 그 결과 W3C는 새로운 대체 스키마 언어로 XML 스키마를 채택하였다. XML 스키마의 특징은 다음과 같다[6,8,9,10,11].

- 개념이나 객체들의 관계를 기술할 수 있다.
- 다양한 데이터 형을 지원한다.
- 네임스페이스를 지원한다.
- 별명, 엘리먼트와 어트리뷰트의 상관관계 및 계통구조, 참조, 다중 키 개념을 이용하여 개념 스키마를 구현한다.
- 열린 모델을 지원하여 XML 문서가 유효한 문서이면서 확장될 수 있도록 지원한다.

이러한 XML 스키마를 만들어 줌으로써 서로 다른 시스템에서 여러 개발자가 XML 문서를 생성할 때나 개발할 때, 이것을 참고하여 문서가 올바른지 검증할 수 있으며, 엘리먼트와 어트리뷰트 그리고 다른 엘리먼트와의 관계 등을 스키마만 가지고 파악할 수 있다[6,8].

그림 6은 그림 5의 XML 문서의 각 엘리먼트를 정의하는 스키마로써 <schema> 루트 엘리먼트의 부엘리먼트로 <Post>, <Name>, <Address> 엘리먼트를 각각 선언하였고, 전역으로 선언된 <Name> 엘리먼트와 <Address> 엘리먼트는 <Post> 엘리먼트에서 참조된다. 즉 다음 스키마의 구조를 따르는 XML 문서에서 루트 엘리먼트는 <Post>가 되고, <Name>과 <Address> 엘리먼트는 <Post> 엘리먼트의 부엘리먼트가 된다.

```

<?xml version="1.0"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema">
<element name="Post">
<complexType><sequence>
<element ref="Name"/>
<element ref="Address"/>
</sequence></complexType>
</element>
<element name="Name">
<complexType><sequence>
<element name="FirstName" type="string"/>
<element name="MiddleInitial" type="string"/>
<element name="LastName" type="string"/>
</sequence></complexType>
</element>
<element name="Address">
<complexType><sequence>
<element name="Street1" type="string"/>
<element name="Town" type="string"/>
<element name="City" type="string"/>
<element name="StateProvinceCountry" type="string"/>
<element name="Country" type="string"/>
<element name="ZipPostCode" type="string"/>
</sequence></complexType>
</element>
</schema>
    
```

그림 6. XML 스키마의 예  
Fig. 6 An Example of XML Schema

### 2.3 GML

미래의 지리정보시스템의 데이터 교환 표준으로 XML을 채택하고자 하는 많은 연구가 진행 중이며, 이미 다양한 분야의 지리정보시스템들이 속속 XML을 채택하고 있다. 이러한 지리정보시스템을 위한 XML 표준화 움직임으로써 가장 대표적인 것이 바로 OGC의 GML이다[3,13,14].

GML은 OGC에 의해 개발된 공간, 비공간 정보를 포함한 지리정보 데이터를 저장하고 전송하기 위한 XML 기반의 인코딩(Encoding) 표준이다. GML은 2000년 5월 버전 1.0이 발표된 이후, 2001년 2월에 버전 2.0, 2003년 1월에 버전 3.0이 발표되었으며, 앞으로 지리정보시스템을 위한 표준으로 자리 잡을 것으로 보인다. 본 논문에서 소개하고자 하는 GML 사양은 OGC의 권고안 2.0에 기초하고 있다.

GML은 XML 스키마 후보 권고안과 XML 네임스페이스 권고안에 기반하고 있다. 이를 토대로 Feature, Geometry, Xlinks의 3가지 기본 스키마를 제공하고 있으며 이 스키마들의 관계를 UML 다이어그램으로 표현하면 그림 7과 같다. Feature 스키마는 Geometry 스키마를 참조하고 Geometry 스키마는 XLinks 스키마를 포함한다[3,13].

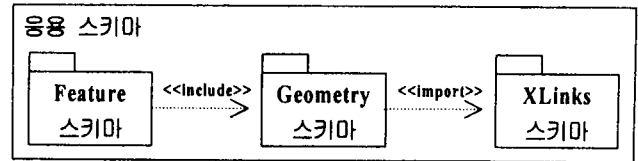


그림 7. 기본 스키마 패키지  
Fig. 7 Basic Schema Package

### 2.3.1 Geometry 스키마

Geometry 스키마에서는 Geometry 엘리먼트를 정의한다. 이를 위해 OGC 심플 Feature 사양에 따라 Curve, Surface, MultiSurface, MultiCurve를 제외한 Point, LineString, LinearRing, Polygon, MultiPoint, MultiLineString, MultiPolygon, 이들에 대한 Geometry Collection 및 Box를 제공한다[13].

GML에서는 Geometry 정보를 표현하기 위해서 소수(decimal) 타입의 x, y, z 좌표를 가지는 Point, LineString, Box, LinearRing, Polygon과 여기서 확장된 MultiPoint, MultiLineString, MultiPolygon등의 Geometry 타입을 제공한다.

GML Geometry 스키마에서 제공하는 Geometry 스키마의 구조는 본 논문에서의 공간 객체 엘리먼트의 스키마 구조와 매우 유사하다. 따라서 본 논문은 공간 객체 엘리먼트를 표현하기 위해, 필요한 공간 객체 엘리먼트의 데이터 구조인 Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon을 GML의 Geometry 스키마 구조를 수용하여 사용한다.

### 2.3.2 Feature 스키마와 Xlinks 스키마

Feature 타입, Feature 식별자(fid), 이름, 설명과 같은 공통적인 Feature 어트리뷰트를 정의한다. Feature 스키마는 Geometry 스키마를 참조로 하며, 이를 위해 <include> 엘리먼트를 사용한다[13]. 그리고 XLinks 스키마는 Linking 기능을 지원하기 위한 Xlinks 어트리뷰트를 제공한다.

### 2.3.3 응용 스키마

GML의 기본 스키마들로서는 실제 사용할 데이터를 표현하기 위한 적합한 스키마를 제공할 수 없다. 즉 기본 스키마가 지리 정보를 표현하기 위한 기본적인 형태를 제공한다면, 응용 스키마는 개발자가 GML 표준 컴포넌트를 사용해 실제 사용할 특정 타입이나 프로퍼티(Property) 타입들을 직접 정의해 놓은 스키마이다. 즉 응용 스키마 개발자는 기본 스키마를 이용함으로써 OGC표준에 대한 깊은 이해 없이도 표준에 맞는 응용을 보다 쉽게 개발할 수 있다.

그림 8은 GML 응용 스키마로써, <Dean> 엘리먼트와 부엘리먼트 <familyName>, <age>, <nickName> 그리고 발생순서와 속성을 선언하였다. 또한 gml 네임스페이스의 <gml:location> 엘리먼트를 참조하여 위치정보를 표현할 수 있다.

```
<element name="Dean" type="ex:DeanType"
  substitutionGroup="gml:Feature"/>
<complexType name="DeanType">
  <complexContent>
    <extension base="gml:AbstractFeatureType">
      <sequence>
        <element name="familyName" type="string"/>
        <element name="age" type="integer"/>
        <element name="nickName" type="string"
          minOccurs="0" maxOccurs="unbounded"/>
        <element ref="gml:location"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
</element>
```

그림 8. GML 스키마의 예  
Fig. 8 An Example of GML Schema

그림 9는 그림 8의 GML 스키마 문서의 구조를 따르는 GML 문서로써 <Dean> 엘리먼트와 부엘리먼트인 <familyName>, <age>, <nickname>과 위치정보를 가지는 <gml:location> 엘리먼트들로 구조화된 GML 문서의 예이다[13].

```
<Dean>
  <familyName>Smith</familyName>
  <age>42</age>
  <nickName>Smithy</nickName>
  <nickName>Bonehead</nickName>
  <gml:location>
    <gml:Point>
      <gml:coord>
        <gml:X>1.0</gml:X>
        <gml:Y>1.0</gml:Y>
      </gml:coord>
    </gml:Point>
  </gml:location>
</Dean>
```

그림 9. GML 문서의 예  
Fig. 9 An Example of GML Document

### III. 전자해도용 XML 스키마 설계

본 장에서는 S-57 전자해도 표준 형식을 위한 전자해도용 XML 스키마에 대하여 설명한다. 제안한 전자해도용 XML 스키마에서는 S-57 전자해도 표준 형식과 대응되는 각 엘리먼트와 어트리뷰트들을 정의한다.

#### 3.1 전자해도용 XML 스키마의 구조

전자해도용 XML 스키마는 특정 객체 엘리먼트, 어트리뷰트, 부엘리먼트 집합 A, B, C, 공간 객체 엘리먼트 선언으로 구성된다.

표 6. S-57 전자해도 표준 형식과 XML 스키마의 대응관계

Table. 6 The Correspondence Relation between S-57 Standard Format and XML Schema

S-57 전자해도 표준 형식	전자해도용 XML 스키마
특징 객체	특징 객체 엘리먼트
식별자	어트리뷰트
어트리뷰트 집합 A	부엘리먼트 집합 A
어트리뷰트 집합 B	부엘리먼트 집합 B
어트리뷰트 집합 C	부엘리먼트 집합 C
공간 객체	공간 객체 엘리먼트

표 6과 같이 S-57 전자해도 표준 형식의 특징 객체를 전자해도용 XML 스키마에서는 특징 객체 엘리먼트로 선언하고, S-57 전자해도 표준 형식의 식별자는 전자해도용 XML 스키마에서 어트리뷰트로 선언하였으며, S-57 전자해도 표준 형식

의 어트리뷰트 집합 A, B, C는 전자해도용 XML 스키마에서 부엘리먼트 A, B, C로 선언하였다. 표 6에서 정의한 대응관계 규칙에 따르면, 각 전자해도용 XML 스키마는 그림 10과 같은 계층 구조로 설계된다. 모든 스키마 문서의 루트 엘리먼트가 되는 <schema> 엘리먼트가 있으며, 모든 XML 문서는 오직 하나의 루트 엘리먼트만을 허락하기 때문에 <ENCObjectSet> 엘리먼트를 선언하여 <ENCObjectSet> 엘리먼트가 전자해도용 XML 문서에서 루트 엘리먼트가 되도록 한다.

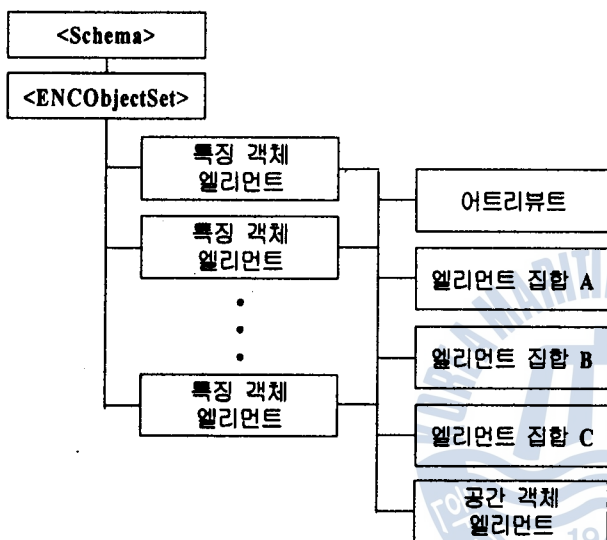


그림 10. 전자해도용 XML 스키마의 구조  
Fig. 10 Structure of ENC XML Schema

<ENCObjectSet> 엘리먼트는 내부에 특징 객체 엘리먼트들을 부엘리먼트로 갖는다. 특징 객체 엘리먼트는 내부에 어트리뷰트와 부엘리먼트 집합 A, B, C, 그리고 공간 객체 엘리먼트를 갖는다.

### 3.2 특징 객체 엘리먼트 정의

전자해도용 XML 스키마에서 특징 객체 엘리먼트는 S-57 전자해도 데이터에서의 특징 객체에 대응된다. 그러나 전자해도용 XML 스키마의 특징 객체 엘리먼트는 S-57 전자해도 표준 형식의 객체 클래스와 같은 개념이다. 즉 특징 객체 엘리먼트는 S-57 전자해도 표준 형식의 객체 클래스 구조의 루트 엘리먼트가 된다.

특징 객체 엘리먼트를 선언하면서 사용한 엘리먼트 이름은 S-57 전자해도 표준 형식에서 특징 객체를 위해 정의된 6자 약어이다. 특징 객체 엘리먼트의 한 예로 'LIGHTS' 라는 특징 객체 엘리먼트 선언은 다음과 같다.

```
<xs:element name="LIGHTS"></xs:element>
```

제안한 전자해도용 XML 스키마는 특징 객체 엘리먼트 내부에 어트리뷰트 선언, 엘리먼트 A, B, C 선언, 그리고 공간 객체 엘리먼트 선언을 가진다.

### 3.3 어트리뷰트 정의

전자해도용 XML 스키마에서 각 특징 객체 엘리먼트들이 가지는 어트리뷰트는 S-57 전자해도 표준 형식에서의 식별자와 대응된다. S-57 전자해도 표준 형식의 식별자는 해당 특징 객체 엘리먼트의 산술코드나, 그룹, 식별번호 등과 같은 의미를 가지고 있으므로 특징 객체 엘리먼트를 위한 어트리뷰트로 선언하는 것이 가장 적절하다. 어트리뷰트는 모두 6개이며 모든 특징 객체 엘리먼트들은 동일한 어트리뷰트를 가진다. 제안한 전자해도용 XML 스키마는 어트리뷰트들을 그룹으로 선언하고, 이를 각 특징 객체 엘리먼트에서 참조한다.

```
<xs:attributeGroup name = "AttributeGroup">
<xs:sequence>
<xs:attribute name = "GRUP" type = "xs:integer"/>
<xs:attribute name = "OBJL" type = "xs:integer"/>
<xs:attribute name = "RVER" type = "xs:integer"/>
<xs:attribute name = "AGEN" type = "xs:integer"/>
<xs:attribute name = "FIND" type = "xs:integer"/>
<xs:attribute name = "FIDS" type = "xs:integer"/>
</xs:sequence>
</xs:attributeGroup>
```

그림 11. 어트리뷰트의 XML 스키마 정의  
Fig. 11 XML Schema Definition for Attribute

어트리뷰트 선언을 재사용하여 효율적인 스키마 설계를 위해 어트리뷰트들은 그룹으로 선언한다. 제안한 어트리뷰트 정의는 그림 11과 같다.

### 3.4 부엘리먼트 집합 정의

전자해도용 XML 스키마의 부엘리먼트 집합은

S-57 전자해도 표준 형식의 어트리뷰트 집합에 대응된다. 본 논문에서는 전자해도 XML 문서가 복잡해지는 것을 막기 위하여 S-57 전자해도 표준 형식의 어트리뷰트 집합을 전자해도용 XML 스키마에서 부엘리먼트로 선언한다. 각 엘리먼트의 선언에 있어서 사용한 엘리먼트 이름은 S-57 전자해도 표준 형식에서 어트리뷰트들을 위해 정의된 6자 약어이다.

제안한 전자해도용 XML 스키마에서 모든 특징 객체 엘리먼트는 엘리먼트 집합 A, B, C의 일부를 부엘리먼트로 가진다.

### 3.4.1 부엘리먼트 집합 A 정의

전자해도용 XML 스키마의 부엘리먼트 집합 A는 S-57 전자해도 표준 형식의 어트리뷰트 집합 A이다. 특징 객체 엘리먼트들은 서로 다른 부엘리먼트를 가진다. 따라서 부엘리먼트 집합 A에 해당하는 엘리먼트들은 그룹화하여 재사용하기 어렵기 때문에, 부엘리먼트 집합 A는 'LIGHTS' 특징 객체 엘리먼트에서와 같이, 모두 특징 객체 엘리먼트 내부에 각각 부엘리먼트로 선언한다. 그림 12는 부엘리먼트 집합 A의 XML 스키마 정의의 일부이다.

```
<xs:element name = "CATLIT" type = "xs:string"/>
<xs:element name = "COLOUR" type = "xs:string"/>
<xs:element name = "DATEND" type = "xs:string"/>
<xs:element name = "DATSTA" type = "xs:string"/>
<xs:element name = "EXCLIT" type = "xs:integer"/>
<xs:element name = "HEIGHT" type = "xs:float"/>
<xs:element name = "LITCHR" type = "xs:integer"/>
<xs:element name = "LITVIS" type = "xs:string"/>
...
```

그림 12. 부엘리먼트 집합의 XML 스키마 정의  
Fig. 12 XML Schema Definition for Subelement Set

### 3.4.2 부엘리먼트 집합 B 정의

전자해도용 XML 스키마의 부엘리먼트 집합 B는 S-57 전자해도 표준 형식의 어트리뷰트 집합 B이다. 부엘리먼트 집합 B에 해당하는 부엘리먼트들은 모두 7개이며, 부엘리먼트 집합 B 선언은 그룹으로 묶어 각 특징 객체 엘리먼트에서 참조하도록 한다. 또한 재사용성을 높이기 위해 부엘리

먼트 집합 B 선언은 4개의 그룹으로 나누어 선언한다. 각 특징 객체 엘리먼트들은 부엘리먼트 집합 B의 4가지 그룹 중의 한 그룹을 부엘리먼트로 가진다. 그림 13은 부엘리먼트 집합 B의 4가지 그룹에 대한 정의의 일부이다.

```
<xs:group name = "ElementSetB1">
<xs:sequence>
<xs:element name = "IMFORM" type = "xs:string"/>
<xs:element name = "NINFOM" type = "xs:string"/>
<xs:element name = "NTXTDS" type = "xs:string"/>
<xs:element name = "PICREP" type = "xs:integer"/>
<xs:element name = "SCAMAX" type = "xs:integer"/>
<xs:element name = "SCAMIN" type = "xs:integer"/>
<xs:element name = "TXTDSC" type = "xs:string"/>
</xs:sequence>
</xs:group>
<xs:group name = "ElementSetB2">
<xs:sequence>
...
</xs:sequence>
</xs:group>
<xs:group name = "ElementSetB3">
<xs:sequence>
<xs:element name = "IMFORM" type = "xs:string"/>
<xs:element name = "NINFOM" type = "xs:string"/>
...
</xs:sequence>
</xs:group>
<xs:group name = "ElementSetB4">
<xs:sequence>
...
</xs:sequence>
</xs:group>
```

그림 13. 부엘리먼트 집합 C의 XML 스키마 정의  
Fig. 13 XML Schema Definition for Subelement Set B

### 3.4.3 부엘리먼트 집합 C 정의

전자해도용 XML 스키마의 부엘리먼트 집합 C는 S-57 전자해도 표준 형식의 어트리뷰트 집합 C이다. 부엘리먼트 집합 C의 엘리먼트는 모두 4개이며, 엘리먼트의 재사용을 위해 그룹으로 묶어 전역으로 정의한다. 모든 특징 객체 엘리먼트들은 부엘리먼트 집합 C를 부엘리먼트로 갖는다. 그림 14는 부엘리먼트 집합 C에 대한 정의이다.

```
<xs:group name = "ElementSetC">
<xs:sequence>
<xs:element name = "RECDAT" type = "xs:string"/>
<xs:element name = "RECIND" type = "xs:string"/>
<xs:element name = "SORDAT" type = "xs:string"/>
<xs:element name = "SORIND" type = "xs:string"/>
</xs:sequence>
</xs:group>
```

그림 14. 부엘리먼트 집합 C의 XML 스키마 정의  
Fig. 14 XML Schema Definition for Subelement Set C

### 3.5 공간 객체 엘리먼트 정의

전자해도용 XML 스키마의 공간 객체 엘리먼트는 S-57 전자해도 표준 형식의 공간 객체이다. 공간 객체는 특정 객체 엘리먼트의 위치 정보를 표현하는 부엘리먼트이다. 공간 엘리먼트는 모두 6가지형으로 정의하며, <Point>와 <MultiPoint>, <LineString>과 <MultiLineString>, 그리고 <Polygon>과 <MultiPolygon>이다. <Point>, <LineString>, <Polygon> 엘리먼트는 부엘리먼트로 <Coordinates> 엘리먼트를 참조하고, <MultiPoint>, <MultiLineString>, <MultiPolygon> 엘리먼트는 부엘리먼트로 <Coordinates3D> 엘리먼트를 참조한다. 재사용성을 높이기 위해 공간 객체 엘리먼트의 각 엘리먼트들은 그룹으로 묶어 전역으로 정의하여 각 특정 객체 엘리먼트에서 참조할 수 있도록 한다. 공간 객체 엘리먼트의 구조는 그림 15와 같다.

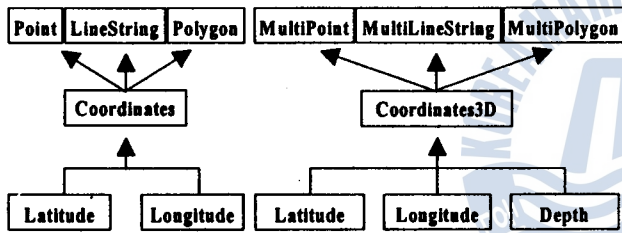


그림 15. 공간 객체 엘리먼트의 구조  
Fig. 15 Structure of Spatial Object Element

GML 2.0의 Geometry 스키마에서 지원하는 각 엘리먼트의 구조와 본 논문의 공간 객체 엘리먼트의 데이터 구조가 유사하여 공간 객체 엘리먼트 정의의 각 엘리먼트 이름과 구조는 GML 2.0의 Geometry 스키마 구조를 수용하여 사용한다[13, 15]. 공간 객체 엘리먼트 선언에서 <Coordinates> 엘리먼트는 부엘리먼트로 <Latitude> 엘리먼트와 <Longitude> 엘리먼트를 선언하며, <Coordinates3D> 엘리먼트는 <Latitude> 엘리먼트와 <Longitude> 엘리먼트 그리고 <Depth> 엘리먼트를 부엘리먼트로 선언한다.

### 3.6 통합된 전자해도용 XML 스키마

그림 16은 제안한 각 스키마들을 통합한 전자해도용 XML 스키마 정의의 일부이다. 그 중 'LIGHTS' 특정 객체 엘리먼트 선언과 관련된 부

분을 보면, 구조는 <schema> 엘리먼트 선언과 <ENCObjectSet> 엘리먼트 선언, <LIGHTS> 엘리먼트 선언, 그리고 그 내부에 어트리뷰트 선언 참조, 부엘리먼트 집합 A 선언, 부엘리먼트 집합 B, C 선언 참조, 공간 객체 엘리먼트 정의 참조 등의 계층적 구조를 가진다.

```

<?xml version = "1.0" ?>
<xs:schema xmlns:xs = "http://www.w3.org/2001/XMLSchema">
<xs:element name= "ENCObjectSet">
  <xs:complexType>
    <xs:element name="LIGHTS" >
      <xs:complexType>
        <xs:sequence>
          <xs:element name = "CATLIT" type = "xs:string"/>
          <xs:element name = "COLOUR" type = "xs:string"/>
          ...
          <xs:group ref = "ElementSetB2"/>
          <xs:group ref = "ElementSetC"/>
          <xs:choice>
            <xs:group ref = "Point"/>
            <xs:group ref = "MultiPoint"/>
          </xs:choice>
        </xs:sequence>
        <xs:attributeGroup ref = "AttributeGroup"/>
      </xs:complexType>
    </xs:element>
    ...
  </xs:complexType>
  <xs:group name = "ElementSetB1">
    <xs:sequence>
      <xs:element name = "IMFORM" type = "xs:string"/>
      <xs:element name = "NINFOM" type = "xs:string"/>
      ...
    </xs:sequence>
  </xs:group>
  ...
</xs:schema>
    
```

그림 16. 전자해도 데이터용 XML 스키마  
Fig. 16 XML Schema for ENC Data

## IV. 변환 프로그램 구현

S-57 전자해도는 IHO/IEC 8211 코드의 이진 파일이다. 이는 유용한 해양 지리 정보를 이용하는데 있어서 많은 걸림돌이 되어 왔다. S-57 전자해도 파일을 보다 손쉽게 이용할 수 있도록 전자해도용 XML 스키마 구조를 따르는 전자해도 XML 문서로 변환하는 프로그램이 필요하며, 본 장에서는 변환 프로그램 구조에 관하여 설명한다.



4.1 변환 프로그램 구조

전자해도 변환 프로그램의 구성은 그림 18과 같다. 먼저 S-57 전자해도 이진 파일은 객체 클래스 단위로 순차적인 구조를 가지고 있으므로 S-57 전자해도 파일을 입력 데이터로 받는다. 다음으로 입력으로 받은 S-57 전자해도를 객체 클래스 단위로 분리하여 텍스트 문서로 변환한다. 마지막으로 텍스트 문서로 변환된 전자해도 데이터와 제안된 전자해도용 XML 스키마 문서의 구조를 비교하여 문서를 제안된 전자해도

용 XML 스키마의 구조를 따르는 전자해도 XML 문서로 변환한다.

4.2 변환된 XML 문서

그림 17은 제안한 방법을 통하여 변환된 전자해도 XML 문서의 'LIGHTS' 특징 객체 엘리먼트의 일부를 나타내었다. 'LIGHTS' 특징 객체 엘리먼트는 6가지의 어트리뷰트, 엘리먼트 집합 A, B, C 그리고 공간 객체 엘리먼트들로 구성되어 있으며 각각의 어트리뷰

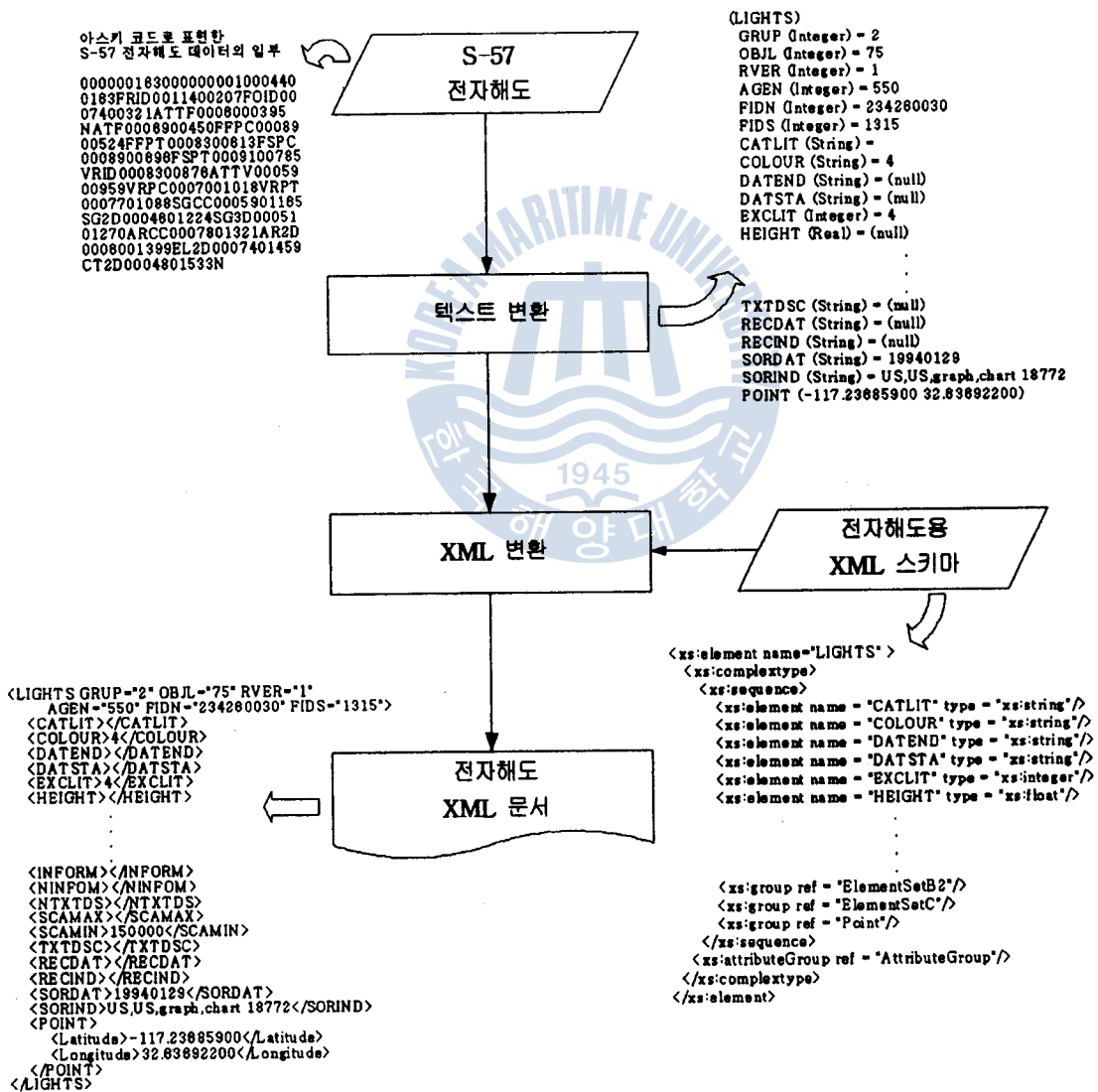


그림. 18 변환 프로그램 흐름도  
 Fig. 18 Flowchart for Translation Program

트와 엘리먼트들에는 변환되기 전의 S-57 전자해도 데이터가 가지고 있던 값들을 가지고 있다.

변환된 XML 전자해도 문서에서 객체 클래스 내의 빈 엘리먼트는 객체 클래스에서 데이터의 가치를 고려하여 삭제하지 않았다. 변환된 전자해도 XML 문서를 각 사용자 요구에 맞게 이용할 때 빈 엘리먼트에 값을 지정하거나 또는 무시할 수 있을 것이다.

```

<?xml version = "1.0"?>
<ENCObjectSet xmlns:xsi= "http://www.w3.org/2001/XMLSchema-
instance" xsi:noNamespaceSchemaLocation="ENCXML.xsd">
<LIGHTS GRUP= "2" OBJL= "75" RVER= "1" AGEN= "550"
FIDN= "234260030" FIDS= "1315">
<CATLIT></CATLIT>
<COLOUR>4</COLOUR>
<DATEND></DATEND>
<DATSTA></DATSTA>
<EXCLIT>4</EXCLIT>
...
<SORDAT>19940129</SORDAT>
<SORIND>US, US, graph, chart 18772</SORIND>
<POINT>
<Latitude>-117.23685900</Latitude>
<Longitude>32.63692200</Longitude>
</POINT>
</LIGHTS>
...
</ENCObjectSet>
    
```

그림 17. 전자해도용 XML 문서의 예  
Fig. 17 An Example of XML Document for ENC

### V. 결론 및 향후 연구과제

S-57 전자해도는 국제수로기구에서 제정한 전자해도정보 표준 형식이다. S-57 전자해도 표준 형식은 ECDIS에서 사용될 것을 목표로 설계되었기 때문에 이 데이터를 다루기 위해서는 전용의 시스템이나 브라우저를 필요로 하며, 이는 해양지리정보 데이터로서 S-57 전자해도가 가지고 있는 유용한 정보를 이용함에 있어 큰 제약 조건으로 작용하였다. 이러한 문제점은 S-57 전자해도를 XML과 같은 범용의 데이터 형식으로 바꿈으로써 해결 가능하다.

따라서 본 논문에서는 S-57 전자해도 표준 형

식에 대응되는 XML 스키마를 제안하였고, S-57 전자해도를 XML 문서로 변환하는 프로그램을 구현하였다. 제안한 방법을 이용하여 XML 문서로 변환된 S-57 전자해도는 인터넷을 이용한 데이터 접근, 데이터 교환, 사용자 요구에 따른 다양한 활용 등이 가능하다. 또한 해도 데이터를 편집할 경우에도 간단한 문서편집기를 사용함으로써 사용자의 편의성이 증가하게 되며, 인터넷을 통하여 다양한 사용자들이 정보를 공유할 수 있다는 장점도 지니게 된다.

차후, 변환된 전자해도 XML 문서의 관리를 위한 데이터베이스를 개발이 필요하다. 또한 웹서비스를 위한 프로그램을 개발한다면 인터넷을 통해 누구나 쉽게 전자해도 데이터를 이용할 수 있을 것이다.

### 참고 문헌

- [1] M. B. Brown, "Developments in The NOAA Electronic Navigational Chart Program", Proc. of the US Hydrographic Conf. 1999, 4.
- [2] S. J. Chang, "Design And Preliminary Test On The Integration of Weather Data in ECDIS for Marine Navigation", Journal of Marine Science and Technology, Vol.9, No.1, pp.21-24, 2001.
- [3] 권준희, 윤용익, "지리정보시스템을 위한 XML", 정보처리학회지 8권 3호, pp.61-69, 2001. 5.
- [4] International Hydrographic Bureau, IHO Transfer Standard for Digital Hydrographic Data, Edition 3.1, 2000. 11.
- [5] G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide, Addison Wesley, 1997.
- [6] 황병연, 김연혜, "XML 스키마 발전 동향", 정보처리학회지 8권 3호, pp.3-9, 2001. 5.
- [7] 김영숙, 조성호, XML BIBLE, 삼양출판사, 2001.
- [8] S. Mohr, F. Norton, N. Ozu, L. Stokes-Rees, J. Tennison, and K. Williams, Professional XML Schemas, Wrox, 2002.
- [9] K. Cagle, D. Hunter, C. Dix, R. Kovack, J.

Pinnock, and J. Rafter, Beginning XML, 2nd ed., Wrox, 2002.

[10] W3C, XML Schema Part 0: Primer, 2001. 5.

[11] W3C, XML Schema Part 1: Structures, 2001. 5. <http://www.w3.org/TR/xmlschema>

[12] W3C, XML Schema Part 2: Datatypes, 2001. 5. <http://www.w3.org/TR/xmlschema>,

[13] OGC, Geography Markup Language(GML) Specification, v2.0, 2000.

[14] OGC, Geography Markup Language(GML) Specification, v3.0, 2003.

[15] B. Ronai, P. Sliogeris, M. D. Plater, and K. Jankowska, Development and Use of Marine XML within the Australian Oceanographic Data Centre to Encapsulate Marine Data, IOC of UNESCO, 2002. 8.

저자 소개



**이성대(Seong-Dae Lee)**

1999년 한국해양대학교 컴퓨터공학과 (공학사)

2001년 한국해양대학교 컴퓨터공학과 (공학석사)

2001년~현재 한국해양대학교 컴퓨터공학과 박사과정

1994.12~1997.1 미래정보CIM

※ 관심분야 : 데이터베이스, 데이터마이닝, XML

**강형석(Hyung-Suk Kang)**

2000년 한국해양대학교 컴퓨터공학과 (공학사)

2003년 한국해양대학교 컴퓨터공학과 (공학석사)

※ 관심분야 : 데이터베이스, XML



**박휴찬(Hyu-Chan Park)**

1985 서울대학교 전자공학과 (공학사)

1987 한국과학기술원 전기및전자공학과 (공학석사)

1995 한국과학기술원 전기및전자공학과 (공학박사)

1987~1990 금성반도체

1997~현재 한국해양대학교 부교수

※ 관심분야 : 데이터베이스, 모델링 방법론, XML