J. Comput. Struct. Eng. Inst. Korea, 30(1)
pp.77~86, February, 2017
https://doi.org/10.7734/COSEIK.2017.30.1.77

pISSN 1229-3059 eISSN 2287-2302
Computational Structural
Engineering Institute of Korea

BIM 소프트웨어를 활용한 토목 시설물 IFC 확장요소기반의 정보모델링 방안

이 상 호[1†] · 박 상 일[1] · 권 태 호[1] · 서 경 완[1]

[1]연세대학교 토목환경공학과

Civil Infrastructure Information Modeling Method Based on Extended IFC Entities using BIM Authoring Software

Sang-Ho Lee[1†], Sang I. Park[1], Tae-Ho Kwon[1] and Kyung-Wan Seo[1]

[1]*School of Civil & Environmental Engineering, Yonsei Univ., Seoul, 03722, Korea*

Abstract

Industry Foundation Classes(IFC) is the core product schema for ensuring the interoperability of information in Building Information Modeling(BIM) environment. However, since current IFC is mainly focused on building structure. there are limitations in the generation of functional information of components when applied to civil infrastructures. Previous studies have proposed IFC-based new entities for the civil infrastructures, it takes long time to support them in BIM authoring software packages. In this study, we proposed practical rules to apply IFC-based information modeling using BIM authoring software and additional new entities for the civil infrastructure through attributes and information mapping. The availabilities of proposed method were examined using the rail and sleeper information models based on an extended IFC data schema for the railway infrastructures.

Keywords : *industry foundation classes(IFC), building information modeling(BIM), civil infrastructure, IFC extension, information mapping*

1. 서 론

건축 구조물을 중심으로 Building Information Modeling (BIM)의 활용성이 검증되기 시작하면서, 토목 분야에서도 건설 프로젝트에 BIM 도입의 적극성이 높아지고 있다. 이에 따라 국내 주요 공공 프로젝트에 BIM 적용이 의무화되면서 토목 시설물에 BIM을 효과적으로 적용하기 위한 방법에 대한 관심이 증가하고 있다. 공공 토목 시설물에 BIM을 적용하는 경우에는 필요에 따른 대안설계, 설계오류 감소 등을 통한 비용절감과 더불어 시공 시뮬레이션 등을 통한 작업 위험도 감소, 공기 단축 등의 효과를 볼 수 있지만, 운영 기간이 길다는 특징에 맞춘다면 사전에 생성해 놓은 정보의 재사용 및 정보 공유의 효율성 향상과 이를 활용한 신뢰도 높은 유지관리 의사결정지원이 중요한 장점이 될 수 있다. 따라서 정보관리의 관점에서 생각해 볼 때, 동일한 표준 프레임워크 내에서 정보 또는 모델을 생성하고 저장하는 것이 핵심이라고 할 수 있으며, 이를 위해 building SMART International(bSI)에서는 Industry Foundation Classes(IFC)라는 BIM을 위한 표준 데이터 스키마(ISO-TC184/SC4, 2013)를 개발하여 배포하고 있고 대부분의 BIM 소프트웨어에서는 IFC 포맷을 다루는 것을 지원한다.

그러나 현재 버전의 IFC 데이터 스키마는 건축물을 대상으로 하고 있어서 토목 시설물에 IFC를 그대로 적용하는 데에는 한계가 있다. 즉, 토목 시설물이 가지고 있는 기능적 의미에 대해서 IFC를 통해서는 정확하게 표현하기가 힘들며 이에 따라 토목 분야에서는 형상모델만을 생성하여 활용하고 있는 실정이다. 따라서 이러한 문제점을 극복하기 위해 기존 IFC에 교량

† Corresponding author:
Tel: +82-2-2123-2808; E-mail: lee@yonsei.ac.kr
Received December 19 2016; Revised December 26 2016;
Accepted December 27 2016
©2017 by Computational Structural Engineering Institute of Korea

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(http://creativecommons.org/licenses/by-nc/3.0) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

(Lebegue, 2005; Yabuki *et al.*, 2006; Lee *et al.*, 2008), 터널(Yabuki, 2009; Borrmann *et al.*, 2015; Lee *et al.*, 2015), 도로(Lee and Kim, 2011; Moon, 2014)의 시설물을 위한 요소를 새롭게 추가 확장하는 연구를 수행하였다. 언급한 연구들은 형상을 구현하는 또는 요소간의 관계를 짓는 방법은 기존의 IFC를 그대로 활용하고, 토목 시설물의 세부 객체에 대한 기능을 새롭게 정의하여 이를 적절한 위치에 배치하는 방식을 따른다. 이러한 방식은 데이터 스키마를 개발하기 위한 전형적이고 궁극적인 방향이지만 새로운 요소를 추가하거나 기존의 요소를 변경/삭제하는 경우에는 이를 정보모델에 반영하기까지 최소 3년 이상의 지연시간이 발생한다. 이러한 불편함을 효과적으로 대처하기 위해 Lee 등(2013)과 Lee 등(2014)은 IFC 프레임워크를 준수하면서 토목 시설물을 정보모델링할 수 있는 방안을 제시하였다. 그러나 언급한 연구들은 특정 구조물에 대한 방법론에 대한 연구로 일반화된 규칙을 제시하고 있지 않아 다른 구조물에 적용하기 위해서는 적지 않은 기술적 숙련도가 요구된다.

따라서 본 연구에서는 토목 시설물에 적용할 수 있는 IFC 데이터 스키마 활용 방안에 대해서 설명하고, 토목 시설물을 위한 확장 요소와 기존 IFC 요소와의 정보 매핑을 통한 확장 IFC기반의 토목 시설물 정보모델링 방법을 제시하였다. 이때 정보 매핑을 위한 일반화된 3가지 규칙을 제시하여 변환기의 개발이 가능함을 보였고, 이를 통해 철도 시설물의 궤도 및 침목 모델에 적용하여 그 활용성을 검토하였다.

2. 토목 시설물을 위한 IFC 확장 및 활용

IFC 데이터 스키마는 BIM 데이터의 소프트웨어 간 상호

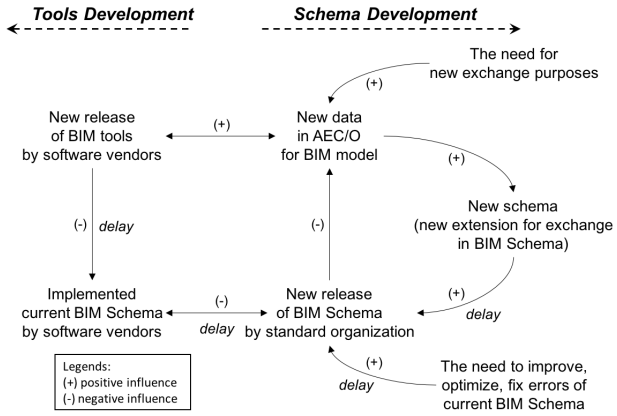

Fig. 1 BIM development casual model (inspired by Cerovsek, 2011)

운용성 확보를 위해 bSI의 Model Support Group (MSG)이 주관하여 국제적으로 공동 개발한 개념적 모델이다(ISO-TC184/SC4, 2013). IFC는 ISO 10303(STEP) Part 11 (ISO-TC184/SC4, 2004)에서 정의하는 기술(description) 언어인 EXPRESS를 활용하여 정의한다. EXPRESS 언어는 객체 지향적 특성에 완벽하게 부합하는 형태이며, 이에 따라 IFC 데이터 스키마 역시 객체지향적 성격을 가진다. 이러한 점으로 인해 특정 객체의 특성을 상속하는 하위 객체 생성(subtyping)을 통해 데이터 스키마를 더욱 정교하게 만들거나 특화시켜 나갈 수 있다. 1장에서 언급한 토목 시설물을 위한 확장 IFC 데이터 스키마 역시 EXPRESS의 이러한 특징으로 인해 가능한 것이다. 한편, 정보 호환이나 교환의 측면에서 생각해 볼 때, 사용할 정보의 저장 형태 및 저장 위치의 구조에 관한 일관된 규약을 제시하는 데이터 스키마의 개발은 무엇보다 중요하며, 이에 따라 선행되어야 할 부분이라고 할 수 있다.

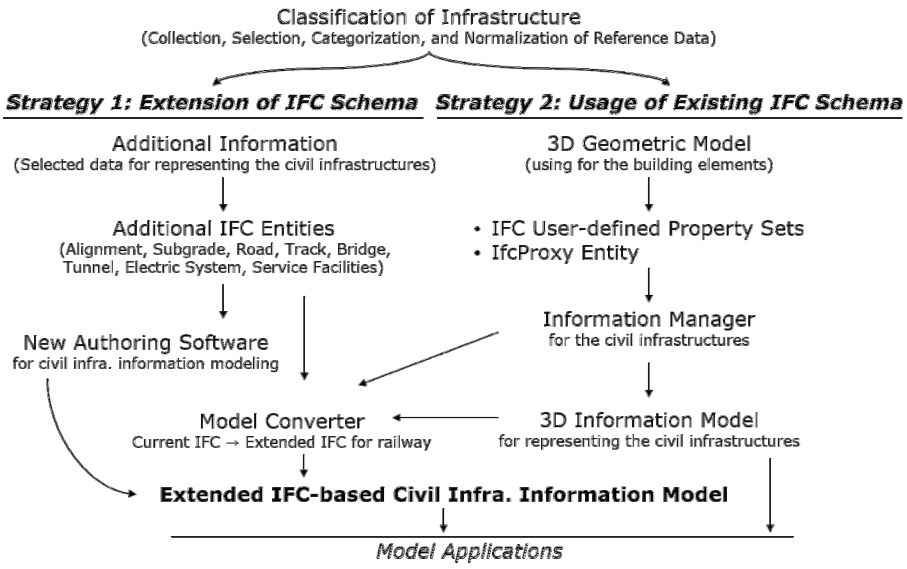

Fig. 2 2-way strategies for extended IFC-based civil infrastructure information modeling

그러나 end-user의 입장에서는 모델 객체에 저장되는 정보의 형태나 구조보다는 손쉬운 활용을 위한 정보 접근성이 보다 중요한 문제라고 할 수 있다. 이런 정보의 접근성은 주로 소프트웨어의 기능 또는 User Interface(UI)/User Experience(UX)와 깊은 관련이 있다고 할 수 있다. 즉, IFC기반의 좋은 토목 시설물 정보모델을 생성하기 위해서는 해당 시설물에 적합한 IFC기반의 데이터 스키마의 개발과 이를 지원하며 사용자 편의성이 높은 소프트웨어의 개발이 모두 이루어져야 한다. 그러나 Fig. 1과 같은 Cerovsek(2011)의 연구에서도 알 수 있는 바와 같이 새로운 스키마의 개발의 과정과 개발한 스키마를 지원하기 위한 소프트웨어의 개발은 시간적 지연이 필연적으로 발생할 수 밖에 없다. 특히 bSI에서 토목 시설물에 적합한 IFC기반의 데이터 스키마를 개발하기 위해 구성한 Infrastructure Room의 조직시점이 2013년인 것에 반해 업무에서 활용할 수 있는 스키마 및 이를 지원하는 소프트웨어가 현재까지 제공되고 있지 않다는 상황에서 알 수 있듯이, 실제 활용 가능한 스키마 및 소프트웨어의 개발 시간은 생각보다 길다. 이러한 점은 궁극적인 지향 방향인 '적절 데이터 스키마 개발 후 이를 지원하는 소프트웨어의 개발'이 완료되기 전에는 토목 시설물 구성요소의 기능적 의미가 정확하게 반영된 정보모델링을 수행하는 것이 어렵다는 것을 의미한다. 이에 따라 본 연구에서는 Fig. 2와 같이 토목 시설물을 위한 데이터 스키마를 정보모델에 반영할 수 있는 2가지 방안을 제시하였다.

Fig. 2에서 strategy 1에 따르는 흐름이 토목 시설물에 대한 IFC기반의 확장 스키마를 생성한 후에 이에 적합한 소프트웨어를 개발하고, 이를 통해 토목 시설물 정보모델을 생성하는 과정을 나타내는 것이다. 그리고 strategy 2는 토목 시설물에 적합한 데이터 스키마가 아직 개발되지 않은 상황에서의 정보 모델링 과정을 나타낸 것이다.

Strategy 1은 확장 스키마기반의 정보 모델링을 위한 전형적인 방법으로, strategy 1의 과정에서 중요하게 고려해야 하는 부분은 IFC 요소확장에 대한 논리성이다. 즉, 1) 추가되는 요소는 현재의 IFC 구조 체계에 적합한 형태이어야 하며, 2) 추가되는 확장 요소는 기존의 요소와의 기능적인 차별성을 가지고 있어야 한다. 따라서 토목 시설물을 위한 새로운 요소는 특정 시설물만을 위한 요소, 토목 시설물 전체에서 공유 가능한 요소 및 건축물을 포함한 건설 산업 전반에서 사용 가능한 요소로 구분할 수 있어야 한다.

Strategy 2는 사용자가 원하는 데이터 스키마가 아직 개발되어 있지 않은 상황에서 BIM 소프트웨어를 활용하여 토목 시설물에 대한 정보모델을 생성하기 위한 방법으로, strategy 2의 과정을 통한 정보모델링에서는 데이터 스키마 체계에 따른 정보 관리보다는 형상 모델의 생성 및 이와 연계되어 있는 속성의 사용자 관리에 보다 초점이 맞추어져 있다. 이때의 형상 모델은 IFC를 지원하는 BIM 소프트웨어에서의 건물 요소를 통해 생성한다. 그리고 형상 요소와 연결되는 속성은 IFC 프레임워크에서 제시하는 user-defined property sets (PSET)를 활용할 수 있다. 건물 요소를 활용한 IFC기반의 토목 시설물 정보모델링에 대한 내용은(Lee *et al.*, 2014)에 구체적으로 서술되어 있다. 결과적으로 strategy 2의 흐름을 통해 생성된 토목 시설물 정보모델은 현재의 BIM 소프트웨어와 이에 따르는 건물 요소를 활용한 것으로, 토목 시설물에 적합한 데이터 스키마 구조를 따르기 위한 모델을 생성하기 위해서는 소프트웨어 독립적인 형태의 모델 변환기(Fig. 2에서의 'Model Converter')를 거치도록 하였다. 모델 변환기의 주 기능은 IFC user-defined property sets이 포함된 현재 버전의 IFC 데이터 스키마를 기반으로 생성된 IFC Physical File(IPF)을 입력 자료로 하여 토목 시설물을 위해 추가로 정의한 요소를 포함하는 IFC 확장 데이터 스키마기반의 정보모델로 변환하는 것이다.

3. IFC PSET과 확장 IFC 요소의 일반화된 매핑 알고리즘

IFC에서는 객체지향적인 개념에 따라 형상과 속성을 독립적으로 생성하고 이를 결합하여 정보를 갖는 객체로 표현한다. 형상과 속성을 표현하는 방법에 대해서는 전체 IFC 프레임워크의 기저에 있는(IFC 내의 다른 요소들이 참조할 수 있는) 'Resource Layer'에서 다루고 있다(Fig. 3 참조). 즉, 'Resource Layer'에서는 기능적 의미에 대한 부분은 배제하고 현상 자체에 대한 부분만을 표현한다. 결과물의 측면에서 보면 건축물과 토목 시설물은 다르게 표현되지만, 결과물을 생성하는

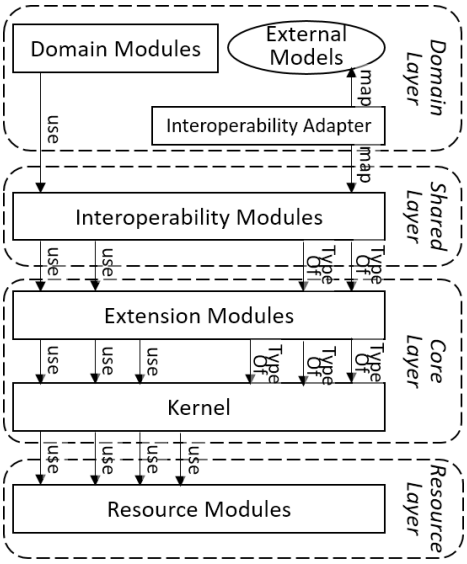

Fig. 3 Layering concept of IFC architecture(See, 1999)

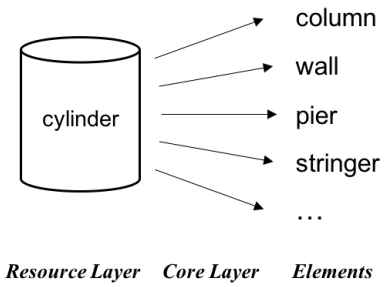

Fig. 4 Concept of relations between 'Resource Layer' and structure elements

중간 과정에서의 구성요소 생성의 측면에서 생각해 보면 Fig. 4와 같이 동일한 과정을 거친다고 할 수 있다. 요약하자면, 토목 시설물을 위해 IFC 요소를 확장한다고 하더라도 이는 객체의 기능적인 정보를 효과적으로 다루기 위한 것으로 IFC의 'Resource Layer'와 'Core Layer'에 속하는 요소들은 그대로 활용할 수 있고, 그대로 활용하는 것이 바람직하다. 즉, 본 연구에서 제시한 방법론을 따르면, 새로운 요소를 추가한 IFC 데이터 스키마기반의 IPF를 다루는데 있어서 초점을 맞춰 고려야해 할 부분은 새롭게 추가된 요소 자체와 그 요소가 포함하고 있는 속성이라고 할 수 있다.

시설물에서 물리적으로 존재하는 객체는 IFC 데이터 스키마의 IfcElement의 하위요소에서 표현한다. IfcElement의 하위요소에는 적용하는 시설물 분야(IfcBuildingElement / IfcCivilElement / IfcGeographicElement), 기능(IfcDistributionElement / IfcFurnishingElement / IfcTransportElement) 또는 형태(IfcElementAssembly / IfcElementComponent / IfcFeatureElement / IfcVirtualElement)에 따라서 하위요소를 구분하여 정의하고 있는데, 토목 시설물을 위한 새로운 물리적 요소는 IfcCivilElement의 하위요소에 추가하는 것이 합리적이라고 할 수 있다(IfcCivilElement가 IfcElement의 하위요소 또는 IfcBuildingElement와 같은 위상인 것이 적절한지는 본 연구에서는 논외로 한다). 이에 따라 IfcCivilElement와 표현하는 시설물만 다를 뿐 개념과 위상이 동일한 IfcBuildingElement의 속성 구성을 살펴보면 Table 1과 같다.

Table 1 IfcBuildingElement entity and its attributes

Entity Name	Explicit Attribute		Defined by
Ifc Building Element	(GlobalId, OwnerHistory, Name, Description,	→	IfcRoot
	ObjectType,	→	IfcObject
	ObjectPlacement, Representation,	→	IfcProduct
	Tag)	→	IfcElement

단, IfcBuildingElement는 abstract 형태로 정의된 요소로 실제 IPF에서는 나타나지 않으며, 따라서 IfcBuildingElement보다 세부적인 의미를 나타내는 하위요소가 실제 값으로 나타난다. 예를 들어 시설물에서의 물리적인 '보'를 나타내는 IfcBeam은 Table 1에서 언급한 IfcBuildingElement의 속성(inherited explicit attributes)과 IfcBeam 자체에서 정의한 'PredefinedType' 속성(local explicit attributes)이 더해져서 표현된다. 이에 따라 IFC 물리적 요소 *PE(i)*의 속성 리스트 PE^a는 다음 식 (1)과 같이 일반화하여 표현할 수 있다.

$$PE(i)^a = \mathrm{LIST}(I(i)^a, L(i)^a) \tag{1}$$

여기서, $I(i)^a$는 *PE(i)*가 상속받은 속성 리스트(inherited explicit attributes)를 의미하며, $L(i)^a$는 *PE(i)*에서 정의한 속성 리스트(local explicit attributes)를 의미한다. 건축물의 기본 요소만을 고려하는 경우에서의 *PE*는 다음과 같이 표현할 수 있다.

$$\begin{aligned} PE(building) &:= subtypesof(IfcBuildingElement) \\ &= \{IfcBeam, IfcBuildingElementProxy, IfcChimney, \\ &\quad IfcColumn, IfcCovering, IfcCurtainWall, IfcDoor, \\ &\quad IfcFooting, IfcMember, IfcPile, IfcPlate, \\ &\quad IfcRailing, IfcRamp, IfcRampFlight, IfcRoof, \\ &\quad IfcShadingDevice, IfcSlab, IfcStair, IfcStairFlight, \\ &\quad IfcWall, IfcWindow\} \end{aligned}$$

*PE(building)*은 1) 건축물을 포함하는 시설물의 형태를 나타내기 위한 가장 핵심적이고 기본적인 요소이며, 2) 따라서 BIM 소프트웨어에서 객체를 생성하기 위한 기본 UI에 우선적으로 포함시키고 있는 것들이다. 따라서 소프트웨어를 활용한 토목 시설물에 대한 3차원 모델링 역시 *PE(building)*의 요소를 활용하거나 *PE(building)*의 요소를 변형한 형태의 요소를 활용하는 것이 대부분이다. 즉, 토목 시설물을 위한 새로운 요소를 IfcCivilElement의 하위 요소로 생성하면 그 속성은 식 (1)의 $L(i)^a$만 달라진다. 이에 더하여 *PE(building)* 원소의 L^a는 IfcDoor, IfcPile, IfcStairFlight, 및 IfcWindow를 제외하고는 모두 'PredefinedType'이다. 따라서 $L(i)^a$의 갯수가 1인 경우에는 Rule 1에 따라 IPF 텍스트 변환을 통해 BIM 소프트웨어를 기반으로 생성한 객체를 토목 시설물을 위한 확장요소로 매핑하는 것이 가능하다.

Rule 1:
if $n(L(j)^a) = 1$

use k in BIM authoring software
then, $k_{EntityName} \mapsto j_{EntityName}$
and $In(L(k)^a) \mapsto In(L(j)^a)$ in IPF

여기서, $j \in PE(civil)$이고, $k \in PE*(building)$이다. *Entity Name*은 IFC 요소명을 나타내는 것으로, $k_{EntityName}$은 IfcBeam 등과 같이 요소의 명칭을 의미한다. 또한 본 연구에서 $\mapsto$ 기호는 정보의 매핑을 뜻한다. $In(x)$는 x의 실제값(instance)를 의미하고, $PE*(building)$은 다음과 같이 정의된다.

$$PE*(building) = PE(building) - \{IfcDoor,\ IfcPile,\ IfcStairFlight,\ IfcWindow\}$$

전술한 바와 같이 IFC에서의 대부분의 물리적 요소는 L^a의 원소의 개수가 하나이다. 그러나 생성하는 객체의 특성에 따라 다루어야 하는 속성은 여러 개가 필요한 경우도 있다. 이러한 경우에는 $PE(building)$의 원소와 $PE(civil)$의 원소의 텍스트 변환을 통한 매핑의 과정에서 정보의 손실을 가져온다. 따라서 2장에서 서술한 IFC user-defined PSET을 매개체로 $PE(civil)$에서 정의한 정보를 생성한다. 즉, Fig. 5와 같이 $PE(civil)$에서 필요한 속성을 IfcProperty의 하위요소를 통해 생성한 후 이를 $PE(building)$의 요소와 연결한 후에 IPF 모델을 만들고, $PE(civil)$에서 해당 요소에 할당하는 것이다.

이러한 과정에서 중요하게 고려해야 할 부분은 PSET에 정보를 저장할 때, 일관된 규칙을 설정하고 이를 따르는 것이다. 이에 따라 $L(IfcPropertySingleValue)^a = LIST(Name,\ Description,\ NominalValue,\ Unit)$로 정의할 때, 본 연구에서는 다음과 같이 Rule 2를 제시하였다.

Rule 2:
$L(j)^a = LIST(att(j)_1,\ att(j)_2,\ \ldots\ ,att(j)_y)$,
if $n(L(j)^a) \geq 2$,
use k as physical element
and
$In(att(IfcPropertySingleValue)_{name}) = "Physical"$
and
$\Phi = In(att(IfcPropertySingleValue)_{NominalValue}) = "j_{GeneralName}{:}In(att(j)_1)-In(att(j)_2)-\ldots-In(att(j)_y)"$
in BIM authoring software
then, $\{\Phi.Split(-)\} \mapsto In(L(j)^a)$

여기서, y는 속성(attribute)구분자이며, $att(j)$는 j 요소의 L^a 중에서 특정 속성을 의미한다. 그리고 $Split(x)$는 x 구분자로 특정 문장을 여러 단어로 분리하는 함수를 나타낸다.

토목 시설물의 공간요소에 적용하기 위한 속성은 전술한 바와 같이 BIM 소프트웨어의 물리적 요소에 할당하며, 따라서 공간요소의 속성은 해당 공간에 속하는 모든 물리적 요소에 할당해야 한다. 공간요소는 시설물에서 공간이 차지하는 영역 및 속성을 정의하기 위해 IFC에서 생성해 놓은 요소로, IfcSpatial Element 및 그 하위요소에서 다루고 있다. 건축물의 경우 건물 자체, 건물의 층, 건물 내부의 거실, 방, 복도 등에 대한 영역이 공간요소로 다루어지고 있는 부분이라고 할 수 있다. 그러나 토목 시설물 분야에 있어서는 '공간' 또는 '생활 공간'이라는 영역보다는, 시설물의 구조적인 기능에 보다 초점을 맞추고 있어, 토목 시설물을 다루는 모델러는 건축물을 다루는 모델러에 비해

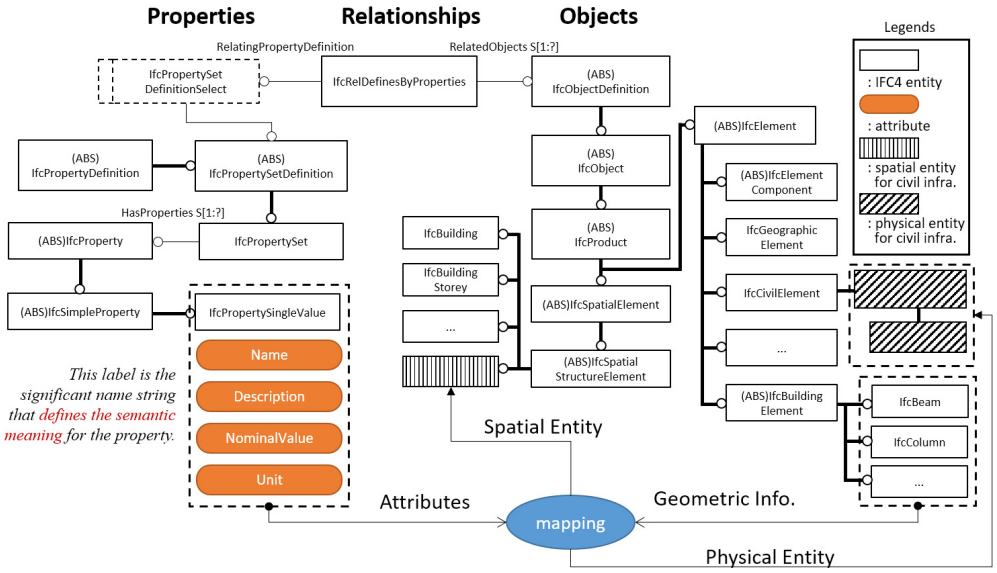

Fig. 5 Object, property and relationship in IFC architecture

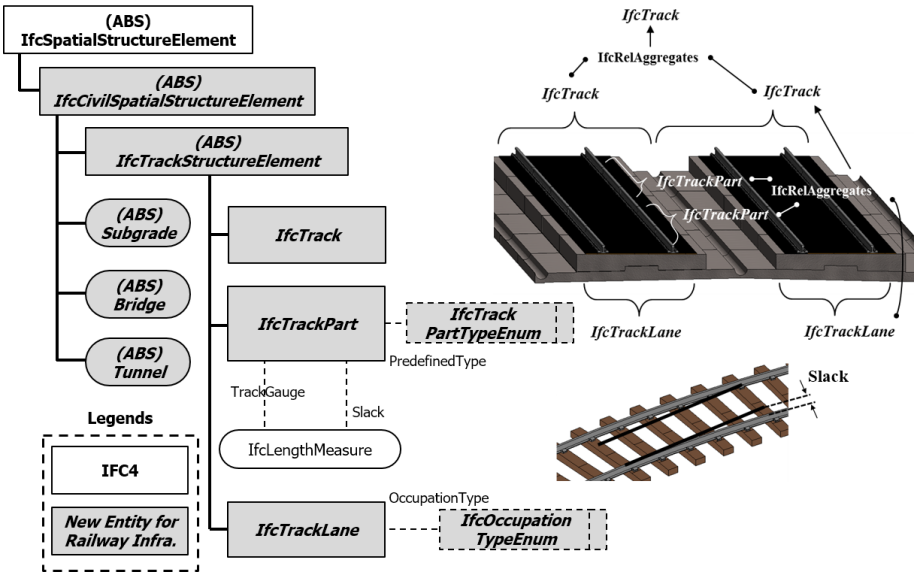

Fig. 6 Simple example of the additional IFC spatial entities for the railway infrastructure

상대적으로 공간에 대한 개념적 이해가 부족하다. 특히 공간에 대한 요소는 일반적으로 BIM 소프트웨어에서 객체로 직접 모델링하기보다는 소프트웨어 내에서 필요 정보를 속성으로 다루면서 IPF로 내보내기 할 때 내부적인 파서(parser)에서 다루는 경우가 많다. 따라서 본 연구에서의 토목 시설물을 위한 공간요소의 생성을 위한 Rule 3은 Rule 2와 유사한 방식으로 하되, BIM 소프트웨어에서 정보를 생성할 때는 *PE*(building)*과 연결된 PSET을 활용하였다. Rule 3은 다음과 같이 정의하였다.

Rule 3:
use k for spatial element
and
$In(att(IfcPropertySingleValue)_{name}) = "Spatial$
and
$\Pi = In(att(IfcPropertySingleValue)_{NominalValue}) =$
$"\varphi_{id}{:}In(att(\varphi_1)_1)-In(att(\varphi_1)_2)-\ldots$
$-In(att(\varphi_1)_y)-In(att(\varphi_2)_1)-\ldots-In(att(\varphi_m)_y)"$
in BIM authoring software
then, $\{\Pi Split(-)\} \mapsto In(L(\varphi_m)^a)$ in IPF

여기서, $\varphi \in SE(civil)$이며, $SE(civil)$은 토목 시설물의 공간요소를 나타내기 위한 IFC 추가요소에 대한 집합이다.

4. 철도 궤도 및 침목을 위한 IFC 요소 확장과 알고리즘 적용을 통한 정보모델의 생성

본 연구에서는 제시한 방법론을 활용하여 철도 시설물의 궤도 및 침목에 적용하였다. 철도 시설물을 위한 IFC 요소의 확장은 1) 기존의 IFC4를 기본으로 하여 현재의 IFC는 변형하지 않고 필요항목을 추가하는 방향으로 하며, 2) 철도 시설물에 대한 공간적, 물리적 객체를 포함할 수 있도록 하였다. 다만, 본 연구에서 제시하는 철도시설물의 궤도 및 침목에 대한 요소는 기본 설계에서 활용하는 수준의 낮은 상세수준(Level of Detail, LoD)이며, 이에 따라 실제 업무에서 활용을 위한 것이라기 보다는 전술한 방법론을 검토하기 위한 목적으로 하였다.

Fig. 6은 철도 궤도부에 대한 공간요소 정보관리를 위한 것으로, IfcTrackStructureElement의 하위요소로 IfcTrack, IfcTrackPart 및 IfcTrackLane의 요소를 추가해 주었다.

IfcTrackPart와 IfcTrackLane은 각각의 L^a를 가지며, IfcTrack은 이들을 통합하는 기능을 한다. 이때 본 연구에서는 IfcTrackPartTypeEnum과 IfcOccupationTypeEnum은 다음과 같이 정의하였다.

```
TYPE IfcTrackPartTypeEnum =
  ENUMERATION OF
   (MAINTRACK, STATIONTRACK,
    BRIDGETRACK, TUNNELTRACK,
    USERDEFINED, NOTDEFINED);
END_TYPE;

TYPE IfcOccupationTypeEnum =
  ENUMERATION OF (UPLINE, DOWNLINE,
    USERDEFINED, NOTDEFINED);
END_TYPE;
```

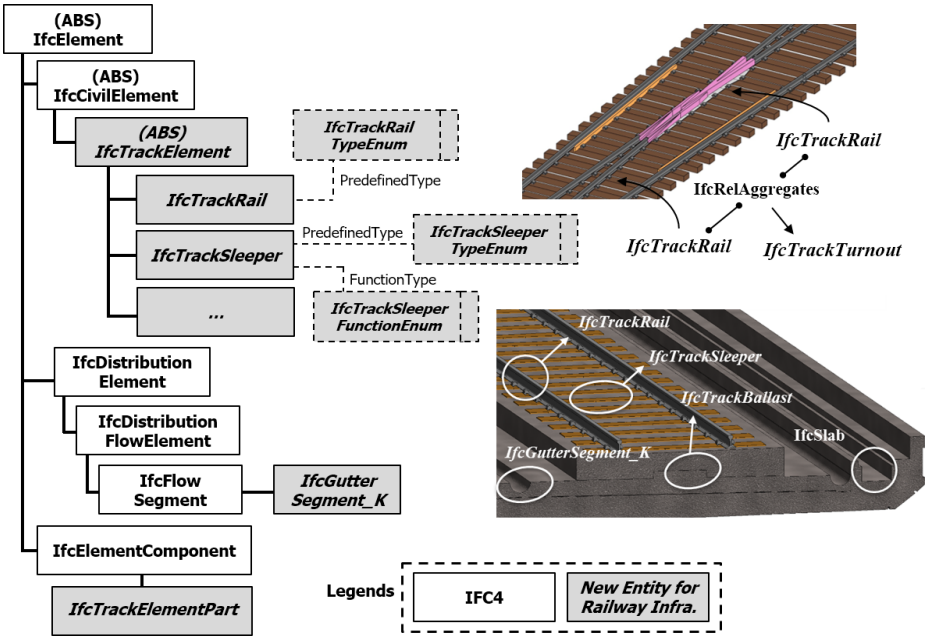

Fig. 7 Simple example of the additional IFC physical entities for the railway infrastructure

Fig. 7은 철도 궤도부에 대한 물리적 요소에 대한 추가 항목을 나타낸 것으로, IfcTrackElement의 하위에 실제 요소를 배치하였으며, IfcGutterSegment_K는 Moon(2014)의 연구의 결과를 차용하였다. 이때, IfcTrackRail은 L^a로 'PredefinedType' 하나의 속성을 가지고 있어 Rule 1을 적용할 수 있으며, IfcTrackSleeper는 L^a가 2개인 경우로, Rule 2를 적용할 수 있다.

Fig. 8은 궤도와 침목을 중심으로 BIM 소프트웨어를 활용하여 모델링한 것을 IPF로 생성하여 Solibri Model Checker를 통해 나타낸 것이며, Table 2는 이에 대한 IPF 내부 데이터의 일부를 나타내고 있는 것이다.

Table 2에서 #282147, #306690는 위치가 다른 두 개의 침목에 대한 물리적 정보를, #6229는 궤도에 대한 물리적 정보를 담고 있는 요소이며, #282148, #306691, #6230는 각각 공간적 정보를 담고 있다. 침목의 경우, *PE*(building)*중 IfcSlab의 요소를 활용하고 있으며, 궤도는 IfcBeam을 활용하고 있다. 물리적 요소와 공간적 요소는 #317927과 #318316의 IfcRelContainedInSpatialStructure로 연결되어 있으며, 공간적 요소는 모두 IfcBuildingStorey로 나타내고 있다. 전술한 바와 같이 궤도는 하나의 L^a만을 가지고 있어 Rule 1을 적용할 수 있으며, 침목은 Rule 2를 적용해야 한다. 공간적 정보는 궤도와 침목 모두 Rule 3을 적용하였다. 이러한 내용을 바탕으로 본 연구에서는 Fig. 6과 Fig. 7의 새로운 요소를 반영할 수 있는 모델 변환기 개발을 통해 Table 3과 같은 철도의 궤도 및 침목에 대한 정보모델의 IPF를 생성하였다.

궤도의 경우, Table 2와 Table 3을 비교했을때, #6214의 물리적 요소가 IfcBeam에서 IfcTrackRail로 변경되었다. 또한 Table 2에서 PSET으로 처리가 되었던 'Predefined Type'의 'MAINRAIL' 속성은 물리적 요소 내의 속성에서 다루고 있다. Table 2에서의 경우 공간요소에 대한 정보는, 실제 활용요소는 #140인 IfcBuildingStorey에서 다루고 있지만, 궤도를 위한 것은 아니며 궤도와 관련한 공간정보는 #6230에서 표현된다. Table 3에서는 IfcTrack의 항목을 통해 공간정보를 다루고 있는데, IfcTrack(#318820)은 IfcTrackPart(#318818)와 IfcTrackLane(#318819)이 IfcRelAggregates(#318821)를 통해 결합되어 IfcRelContainedInSpatialStructure(#318316)를 통해 IfcTrackRail(#6214)과 연결되어 있다. 이때, 'Sleeper 1'과 'Rail'은 동일한 공간에 배치되어 있기 때문에 동일한 공간요소를 공유하고 있다.

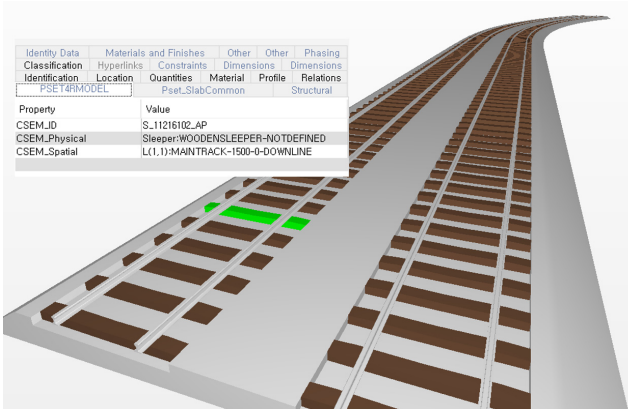

Fig. 8 Rail and sleeper model using BIM authoring software and IFC PSET

Table 2 IPF of BIM authoring software and IFC PSET-based rail and sleeper model

Component	IFC Model Data
Sleeper 1	... #134= IFCBUILDINGSTOREY('3AeNd0ZeX7Ex6qOnFOe5J4',#42,'B.O.',$,$,#132,$,'B.O.',.ELEMENT.,-4600.); #282137= IFCSLAB('0JNphQBD17Qsipq0msoG',#42,'S:146785',$,'S450',#282103,#282134,'146785',.FLOOR.); #282146= IFCPROPERTYSINGLEVALUE('CSEMID',$,IFCTEXT('S11216102_AP'),$); #282147= IFCPROPERTYSINGLEVALUE('CSEM_Physical',$,IFCTEXT('Sleeper:WOODENSLEEPER-NOTDEFINED'),$); #282148= IFCPROPERTYSINGLEVALUE('CSEM_Spatial',$,IFCTEXT('L(1,1):MAINTRACK-1500-0-DOWNLINE'),$); #282199= IFCPROPERTYSET('0JN_phQBD17QsioVK0msoG',#42,'PSET4RMODEL',$,(#282146,#282147,#282148)); #282204= IFCRELDEFINESBYPROPERTIES('0JN_phQBD17QsioFK0msoG',#42,$,$,(#282137),#282199); #317927= IFCRELCONTAINEDINSPATIALSTRUCTURE('2qYKvFEvH4cPghyPJWjrgR',#42,$,$,(#281575,...,#282137,..., #306680,...,#317749),#134); ...
Sleeper 2	... #306680= IFCSLAB('0JNphQBD17Qsipq0mnh3',#42,'S:150322',$,'S450',#306644,#306677,'150322',.FLOOR.); #306689= IFCPROPERTYSINGLEVALUE('CSEMID',$,IFCTEXT('S11216102_AEE'),$); #306690= IFCPROPERTYSINGLEVALUE('CSEM_Physical',$,IFCTEXT('Sleeper:WOODENSLEEPER-NOTDEFINED'),$); #306691= IFCPROPERTYSINGLEVALUE('CSEM_Spatial',$,IFCTEXT('L(2,2):MAINTRACK-1500-5-UPLINE'),$); #306742= IFCPROPERTYSET('0JN_phQBD17QsioVK0mnh3',#42,'PSET4RMODEL',$,(#306689,#306690,#306691)); #306747= IFCRELDEFINESBYPROPERTIES('0JN_phQBD17QsioFK0mnh3',#42,$,$,(#306680),#306742); ...
Rail	... #140= IFCBUILDINGSTOREY('3AeNd0ZeX7Ex6qOnFOe5Jg',#42,'T.O.',$,$,#139,$,'T.O.',.ELEMENT.,-4300.); #6214= IFCBEAM('0AlirFECXEcw156flFMcN8',#42,'W:138786',$,'W:131181',#6083,#6210,'138786',.BEAM.); #6228= IFCPROPERTYSINGLEVALUE('CSEMID',$,IFCTEXT('S51216102_EP'),$); #6229= IFCPROPERTYSINGLEVALUE('CSEM_Physical',$,IFCTEXT('Rail:MAINRAIL'),$); #6230= IFCPROPERTYSINGLEVALUE('CSEM_Spatial',$,IFCTEXT('L(1,1):MAINTRACK-1500-0-DOWNLINE'),$); #6306= IFCPROPERTYSET('0AlirFECXEcw1578FFMcN8',#42,'PSET4RMODEL',$,(#6228,#6229,#6230)); #6311= IFCRELDEFINESBYPROPERTIES('0AlirFECXEcw157OFFMcN8',#42,$,$,(#6214),#6306); #318316= IFCRELCONTAINEDINSPATIALSTRUCTURE('2qYKvFEvH4cPghyPJWjrgr',#42,$,$,(#237,...,#6214,...,#281451),#140); ...

Table 3 IPF of exteded IFC-based rail and sleeper model

Component	Model Data Based on Extended IFC
Sleeper 1	... #282137= IFCTRACKSLEEPER('0JNphQBD17Qsipq0msoG',#42,'S:146785',$,'S450',#282103,#282134,'146785',.WOODENSLEEPER.,.NOTDEFINED.); #318316= IFCRELCONTAINEDINSPATIALSTRUCTURE('2qYKvFEvH4cPghyPJWjrgr',#42,$,$,(#237,...,#6214,...,#282137),#318820); #318818= IFCTRACKPART('0eEdX7qOefW0548FOJgE04',#42,'L(1,1)',$,$,#139,$,'L(1,1)',.ELEMENT.,.MAINTRACK.,1500.,0.); #318819= IFCTRACKLANE('X7E84GdE7xndDgFEe48eXE',#42,'L(1,1)',$,$,#139,$,'L(1,1)',.ELEMENT.,.DOWNLINE.); #318820= IFCTRACK('3AeNd0ZeX7Ex6qOnFOe5Jg',#42,'L(1,1)',$,$,#139,$,'L(1,1)',.COMPLEX.); #318821= IFCRELAGGREGATES('37AmaAzHzEEfEeOYAisCen',#42,$,$,#318820,(#318818,#318819)); ...
Sleeper 2	... #306680= IFCTRACKSLEEPER('0JNphQBD17Qsipq0mnh3',#42,'S:150322',$,'S450',#306644,#306677,'150322',.WOODENSLEEPER.,.NOTDEFINED.); #318822= IFCRELCONTAINEDINSPATIALSTRUCTURE('2qYKvvH4cPgyPJrgr5RF',#42,$,$,(#287877,...,#306680,...),#318825); #318823= IFCTRACKPART('0eEd8XqOefW05_8FOJgE04',#42,'L(2,2)',$,$,#139,$,'L(2,2)',.ELEMENT.,.MAINTRACK.,1500.,5.); #318824= IFCTRACKLANE('XSr84GE7xndDgEe48eXE',#42,'L(2,2)',$,$,#139,$,'L(2,2)',.ELEMENT.,.UPNLINE.); #318825= IFCTRACK('3AeNd0_eX7Ex6q8nFOe5gg',#42,'L(2,2)',$,$,#139,$,'L(2,2)',.COMPLEX.); #318826= IFCRELAGGREGATES('g7Am_AzHzG3fEeOYAisCen',#42,$,$,#318825,(#318823,#318824)); ...
Rail	... #6214= IFCTRACKRAIL('0AlirFECXEcw156flFMcN8',#42,'W:138786',$,'W:131181',#6083,#6210,'138786',.MAINRAIL.); ... * spatial information: the same as 'Sleeper 1'

침목의 경우, Table 2에서는 물리적 요소는 IfcSlab 로 나타나 있으며, Table 3에서는 IfcTrackSleeper로 변경되어 있다. PSET으로 처리된 'PredefinedType'과 'FunctionType' 속성은 IfcTrackSleeper의 내부 속성으로 변경되어 저장되었으며, 공간적인 요소는 궤도와 동일한 형태로 IfcTrackPart와 IfcTrackLane이 결합된 IfcTrack을 통해 표현되고 있다. 이때 'Sleeper 1'과 'Sleeper 2'는 다른 영역의 공간에 속하기 때문에 'Sleeper 1'의 IfcTrack(#318820)과 'Sleeper 2'의 IfcTrack(#318825)은 각각 다르게 정의된다.

5. 결 론

본 연구에서는 현재 IFC로 토목 시설물을 정확하게 표현할 수 없는 한계점을 극복하기 위해, 기존의 건물을 위한 물리적 요소와 IFC user-defined property sets을 활용한 BIM 소프트웨어 기반의 정보모델에서 토목 시설물을 위해 새롭게 생성한 요소에 정확하게 매핑할 수 있는 방안을 제시하였다. 이를 위하여 첫째, IFC의 기본적인 체계를 분석하여 정보 매핑을 위한 적절한 위치를 선정하였다. 둘째, 토목 시설물을 위한 새로운 물리적 요소와 공간적 요소를 기존 IFC 기반의 정보와 매핑 할 수 있는 세 가지 규칙을 제시하였다. 마지막으로 제시한 방법을 적용하기 위해 철도 궤도 및 침목에 대한 IFC 확장 스키마 모델을 제시하였고, 이에 따라 BIM 소프트웨어로 정보모델을 생성하여 제시한 방법론을 적용하였다.

BIM이 갖는 가치 중 가장 중요한 것 중에 하나가 정보의 상호운용성이고, 이를 지원하기 위한 핵심이 중립 표준 포맷인 IFC이다. 그러나 현재의 IFC는 건축물에만 한정되어 있다. 많은 연구자들 및 기관이 토목 시설물에 적합한 형태의 IFC 요소를 확장해 나가고 있지만, 실제 새롭게 개발된 요소가 BIM 소프트웨어에 적용되기 까지는 적지 않은 시간이 소요된다. 또한 IFC 자체도 지속적으로 보완 및 최적화가 필요하며, 이에 따라서 지속적으로 요소가 추가/삭제되거나 또는 위치의 변경이 이루어진다. 본 연구에서 제안한 새로운 IFC 요소에 정보를 매핑할 수 있는 방안을 활용하면 토목 시설물에서도 현재의 BIM 소프트웨어의 기능을 온전하게 활용할 수 있을 뿐만 아니라 BIM 활용 요구사항에 대해서도 효과적으로 대응할 수 있을 것으로 기대된다.

감사의 글

본 연구는 국토교통부 철도기술연구사업의 연구비지원(16RTRP-B104237-02)에 의해 수행되었습니다.

References

Borrmann, A., Kolbe, T. H., Donaubauer, A., Steuer, H., Jubierre, J. R., Flurl, M. (2015) Multi Scale Geometric Semantic Modeling of Shield Tunnels for GIS and BIM Applications, *Comput. Aided Civil & Infrastruct. Eng.*, 30(4), pp.263~281.

Cerovsek, T. (2011) A Review and Outlook for a 'Building Information Model' (BIM): A Multi-Standpoint Framework for Technological Development, *Adv. Eng. Inform.*, 25(2), pp.224~244.

ISO-TC184/SC4 (2004) *ISO 10303-11: 2004 Industrial Automation Systems and Integration - Product Data Representation and Exchange — Part 11: Description Methods: The EXPRESS Language Reference Manual.*

ISO-TC184/SC4 (2013) *ISO 16739:2013 Industry Foundation Classes (IFC) for Data Sharing in the Construction and Facility Management Industries.*

Lebegue, E. (2005) *IFC-Bridge V2 Data Model*, IAI French Chapter, IAI Project CI-2.

Lee, J.-H., Lee, J.-H., Kim, H.-J., Lee, S.-H. (2008) An Extended Data Model based on the IFC for Representing Detailed Design Information of Steel Bridge Members, *J. Comput. Struct. Eng. Inst. Korea,* 21(3), pp.253~263.

Lee, S.-H., Kim, B.-G. (2011) IFC Extension for Road Structures and Digital Modeling, *Proced. Eng.*, 14, pp.1037~1042.

Lee, S.-H., Park, S. I., Park, J. (2013) BIM and Its Application to Civil Engineering: How to overcome the Limitations of Current BIM Technologies, *Civil Engineering Conference in Asia Region: Embracing the Future through Sustainability*, Jakarta, Indonesia, K42~K55.

Lee, S.-H., Park, S. I., Park, J. (2015) Development of an IFC-based Data Schema for the Design Information Representation of the NATM Tunnel, *KSCE J. Civil Eng.*, 20(6), pp.2112~2123.

Lee, S.-H., Park, S. I., Park, K.-Y. (2014) IFC Property Set-based Approach for Generating Semantic Information of Steel Box Girder Bridge Components, *J. Korean Soc. Civil Eng.*, 34(2), pp.687~697.

Moon, H. (2014) Development of IfcRoad in Korea, site: iug.buildingsmart.org(Dec. 01, 2016).

See, R. (1999), *IFC Object Model Guide R2 V2.* International Alliance for Interoperability.

Yabuki, N. (2009) Representation of Caves in a Shield Tunnel Product Model, *The 7th European Conference on Product and Process Modelling*, Sophia Antipolis, France, pp.545~550.

Yabuki, N., Lebegue, E., Gual, J., Shitani, T. (2006) International Collaboration for Developing the Bridge Product Model 'IFC-BRIDGE', *Joint International Conference on Computing and Decision Making in Civil and Building Engineering*, Montreal, Canada, pp.1927~1936.

요 지

Industry Foundation Classes(IFC)는 Building Information Modeling(BIM)을 위한 표준 데이터 스키마로 정보의 상호운용성 확보를 위한 핵심이지만, 건물만을 대상으로 하고 있어 토목 시설물에 적용하기에는 한계가 있다. 이에 따라 기존 IFC에 토목 시설물을 위한 새로운 요소를 추가하는 연구가 진행되었지만, 상용 소프트웨어가 해당 기능을 추가하기 전에는 새로운 스키마를 활용할 수 없다. 본 연구에서는 토목 시설물에 적용하기 위한 IFC 데이터 스키마 활용 방안을 제시하고, 토목 시설물을 위한 확장 요소와 기존 IFC 요소와의 정보 매핑을 통한 확장 IFC기반의 토목 시설물 정보모델링 방법을 제시하였다. 그리고 철도의 궤도 및 침목에 대한 IFC 확장 스키마를 제시하고 제시한 방법을 적용하여 그 활용성을 검증하였다.

핵심용어 : Industry Foundation Classes(IFC), Building Information Modeling(BIM), 토목 시설물, IFC 확장, 정보 매핑