Sensors and Materials, Vol. 38, No. 7 (2026) 4219–4230
MYU Tokyo

S & M 4560

Development of a City Geography Markup Language-to-land Extensible Markup Language Terrain Conversion Technique Supporting Port Facility Maintenance

JaeMyeong Kim,[1] SeungHo Jang,[2*] YeonChan In,[3] and DongGun Lee[3]

[1]College of Interdisciplinary Studies, Major of Urban Planning Engineering, Seokyeong University, 124 Seogyeong-ro, Seongbuk-gu, Seoul 02713, Republic of Korea
[2]National Land DB Department, Spatial Information Industry Promotion Agency, 242 Pangyo-ro, Bundang-gu, Seongnam-si, Gyeonggi-do 13487, Republic of Korea
[3]Department of Urban Planning, Construction and Disaster Prevention Engineering, Seokyeong University, 124 Seogyeong-ro, Seongbuk-gu, Seoul 02713, Republic of Korea

(Received November 27, 2025; accepted July 1, 2026)

Keywords: port facility, maintenance, 3D data conversion, City Geography Markup Language, Land Extensible Markup Language

The need for facility maintenance is gradually increasing owing to the aging of domestic infrastructure such as port facilities. Accordingly, various measures are being taken for maintenance, but more efficient and productive measures are needed. Recently, digital transformation has emerged as a trend in the era of the Fourth Industrial Revolution, and the importance of 3D-data-based maintenance is increasing. Aging port facilities require more structured and interoperable digital data for efficient maintenance. In this study, we propose a City Geography Markup Language (CityGML)-to-land Extensible Markup Language conversion workflow for terrain data supporting port facility maintenance. The method derives terrain-related required classes and elements from both standards, matches semantically equivalent items, and transfers the extracted attribute values into a maintenance-oriented target structure. Validation was conducted using CityGML-based terrain data extracted from the 2019 Hague city model provided by Delft University of Technology for educational and research purposes. The extracted dataset consisted of one terrain object represented by 5595 surface elements in a triangulated irregular network structure. The results confirmed the successful generation of all required target items without omission, along with file-size reduction from approximately 1.71 to 1.07 MB and a conversion time of approximately 0.112 s. Because terrain serves as the spatial reference for facility objects, the proposed workflow provides a foundational step toward future facility-level maintenance data integration.

1. Introduction

In Korea, a total of 60 ports are currently in operation, comprising 31 trade ports and 29 coastal ports. Within these ports, there are 1086 facilities, of which approximately 284 (27.7%)

*Corresponding author: e-mail: sh.jang@spacen.or.kr
https://doi.org/10.18494/SAM6187

ISSN 0914-4935 © MYU K.K.

have been in service for over 30 years. The proportion of aging port facilities has risen from 5.1% in 2000 to 14.0% in 2010 and further to 27.7% in 2020, with projections indicating it will reach 52.5% by 2030. Accordingly, Korea's Ministry of Oceans and Fisheries is devising various measures for port facility maintenance, such as establishing regeneration plans for aging ports and selecting candidate facilities for regeneration projects, by comprehensively evaluating factors including deterioration levels and safety. However, these conventional approaches require significant time and financial resources and are highly dependent on human labor, leading to reduced efficiency. Therefore, new maintenance approaches are needed to improve the efficiency and productivity of port facility management. Recently, with the emergence of information and communication technologies, digital transformation is being applied across various industrial sectors,[1] and efficient facility management incorporating these technologies is increasingly required. Modern facility maintenance relies on sensor technologies such as light detection and ranging (LiDAR), terrestrial laser scanners, and unmanned aerial vehicle (UAV)-mounted cameras to generate high-precision 3D data.[2] Since this data is typically processed in the City Geography Markup Language (CityGML) format for visualization and standardized representation of urban objects,[3] CityGML has limitations in expressing the attribute information required for maintenance. By contrast, Land Extensible Markup Language (LandXML) is a civil-engineering-centered extensible markup language-based schema capable of expressing detailed terrain and structural information,[4,5] making it more suitable for facility management applications. LandXML data are necessary to express attribute information in greater detail. Accordingly, it is essential to comply with the LandXML standard to express attribute information in greater detail and to transition from existing human-centered facility management to an efficient, automation-centered management. In this regard, research on the integration and interoperability between standards and 3D data such as LandXML, CityGML, and geographic information system (GIS) data has been actively conducted. Kang *et al.*[6] proposed an external building information modeling (BIM) reference model identifying common areas between Industry Foundation Classes (IFC) and CityGML. Park *et al.*[7] presented a BIM framework applicable to Korean environments for digital twin implementation. Hwang *et al.*[8] analyzed level-of-detail (LoD) specific object mapping, whereas El-Mekawy[9] developed a unified model for bidirectional conversion. Min *et al.*[10] performed standardization for port facility BIM construction, and Kang *et al.*[11] analyzed interoperability among LandXML, CityGML, and IFC. Kim *et al.*[12] designed data simplification and coordinate transformation scenarios, Ryu and Choo[13] compared geometric representations across these standards, noting challenges including data format structures and LoD mismatches, and Rafiee *et al.*[14] further demonstrated the practical need for BIM/GIS data interoperability through spatial analysis applications. Additionally, domestic groups have addressed foundational issues in spatial data representation and acquisition, the development of open BIM/GIS and 3D spatial and LiDAR, UAV-based 3D reconstruction, and digital elevation model generation.[15–17] However, existing studies have been primarily focused on analyzing 3D map data structures or proposing unidirectional/bidirectional integration rules and frameworks between different standards. Research specifically related to conversion elements, items, criteria, and methods between LandXML and other standards remains insufficient. Furthermore, domain-specialized

conversion for port facility maintenance, which requires the consideration of unique attributes such as material properties, installation dates, safety grades, and environmental exposure conditions specific to maritime infrastructure, has not been specifically addressed in any study. Therefore, we propose a data conversion method of data currently expressed in the CityGML standard into the LandXML-based standard. The contributions of this study are threefold. First, terrain-related classes and elements are systematically derived from CityGML and LandXML. Second, a conversion procedure that is based on mandatory item matching rather than simple format replacement is proposed. Third, the proposed workflow is positioned as a domain-oriented foundational process for port facility maintenance, where terrain acts as the spatial reference for future facility-level integration. In practical maintenance environments, three-dimensional terrain and facility data are commonly acquired through sensing-based technologies, including LiDAR, UAV image measurement, global navigation satellite system (GNSS) surveying, and mobile mapping. Therefore, the efficient structural conversion of such sensing-derived spatial data is an important prerequisite for digital maintenance workflows.

2. Research Scope and Methodology

Both CityGML and LandXML possess class schemes capable of defining facility objects, thereby enabling the digital modeling of social infrastructure or structures. However, before directly converting facility data, it is necessary to verify the conversion of terrain data first, because terrain functions as the foundational spatial dataset on which facility objects are located, interpreted, and managed. Facility data requires superimposition onto terrain data; however, if facility data is converted while the terrain data remains in its original state, discrepancies in data formats may lead to errors. Furthermore, in actual practice, the construction of data specifically for facility management is very limited, and obtaining empirical data based on these formats is challenging. By contrast, terrain data are more consistently structured and provide clear geometric and spatial-reference information, including surface geometry, elevation values, and coordinate-related elements. For this reason, in this study, we adopt a terrain-first scope as the initial validation stage. Accordingly, terrain-related classes and elements are derived from both CityGML and LandXML. The attribute values and structural roles of these classes and elements are compared to identify semantically equivalent items and required target items for conversion. On the basis of the results of this analysis, a conversion procedure is developed to extract terrain-related information from CityGML and store it in the corresponding required items of the LandXML structure. Although the present validation is limited to terrain data, the proposed workflow is designed to be expandable to facility objects. To demonstrate this expandability, an illustrative case of mapping for representative port structures is additionally presented in this study.

3. Analysis of Data Characteristics for Data Conversion

3.1 Analysis of CityGML data characteristics

In this study, CityGML version 1.0 is used since the experimental data from the Delft University of Technology's The Hague city model was constructed using this version.[18] Note that CityGML 3.0, released in 2021, introduces a significantly different data model based on a

modular conceptual schema, which would require a substantially different mapping approach. The LandXML matching performed in this study follows the CityGML 1.0 schema structure, and future work will address compatibility with CityGML 2.0 and 3.0. CityGML is primarily used to represent and visualize three-dimensional geospatial objects. For terrain representation, the relevant hierarchy is centered on the CityGML objects ReliefFeature and ReliefComponent. Terrain can be expressed through several component types, including triangulated irregular network (TIN) Relief, Mass Point Relief, Breakline Relief, and Raster Relief. Among these, TIN Relief is the most suitable for detailed surface representation because it expresses terrain through triangulated geometry with explicit coordinate values. ReliefComponent objects that represent terrain surfaces consist of TIN Relief, Mass Point Relief, Breakline Relief, and Raster Relief. TIN Relief represents the terrain surface using a triangle-based irregular connectivity structure, including length and coordinate information for the lines connecting the start and end points of each triangle. This structure enables the most precise representation among various terrain forms and is effective for accurate terrain depiction as the terrain can be described through a geometry-based structure. The Mass Point Relief method constructs terrain in a 3D point cloud format, where each point indicates a location via x, y, and z coordinate values. The BreakLine Relief method represents terrain boundaries or sections with sudden elevation changes using lines, allowing for the precise depiction of abrupt changes between continuous contour lines. Finally, the Raster Relief method constructs terrain based on images in a grid or general matrix format and is frequently utilized in terrain representation. The components for each class corresponding to these methods are presented in Table 1.

3.2 Analysis of LandXML data characteristics

LandXML data are organized into terrain information through DataSet-level metadata and terrain-oriented classes such as Land Feature, Land Element, Land Surface, and Land Layer. Compared with CityGML, this structure is more suitable for maintenance-oriented data organization because it can store terrain identity, surface representation, and related descriptive attributes in a more explicit engineering format. In the target structure used in this study, DataSet, Land Feature, Land Element, and Land Surface are treated as the principal target classes for terrain conversion. In particular, Land Surface supports TIN-based terrain representation through patch-oriented surface elements, which enables the direct reorganization of triangle-based geometry extracted from CityGML. Land Element represents information regarding specific land types and classifications, and may include unique identifiers, types,

Table 1
Classes defined on CityGML: Relief.

Class	Description
Relief Feature	A feature representing the terrain surface
Relief Component	TIN, raster, grid, and mass point components
TIN Relief	Terrain described using a TIN
MassPoint Relief	Terrain represented as a set of 3D mass points
Breakline Relief	Terrain expressed with 3D breaklines
Raster Relief	Terrain stored in raster or gridded form

materials, and attributes. Among these, the identifier is a mandatory item, whereas type and material are defined optionally. Land Surface expresses characteristics such as contours, boundaries, and terrain steps, and is typically represented using a TIN structure. The TIN used here constructs a continuous terrain surface by connecting triangular faces. This network has a simple triangular structure, and it also allows for the definition of material properties beneath the land surface. Land Layer represents the spatial information of layers existing either below or above the land surface and is divided into Solid Layer, Surface Layer, and Linear Layer. Solid Layer represents a volume with constant thickness along with material properties, while Surface Layer is composed of a structure containing upper and lower boundary surfaces. Linear Layer consists of a sequence of two-dimensional linear elements and may include attribute information such as directionality, positional values, distances, and intersection coordinates. This structure serves as a means to systematically organize the classes of terrain representation elements within the LandXML standard; the respective classes and attributes are presented in Table 2.

4. Classes and Elements for Data Conversion

To achieve data format conversion, it is crucial to clearly understand the terrain-related classes and items included in both standards. However, since each standard follows a distinct framework, differences exist in represented information, composition, and structure. Consequently, even when conveying identical attribute information, there are structural disparities in representation methods, as well as in the nomenclature of classes and items. For instance, although representing the same attribute information, the class related to TIN data is defined as TIN Relief in CityGML, whereas it is defined as Land Surface in LandXML. Despite representing identical information, the conversion between the two cannot be processed through simple matching because of the divergent structural definition methods and class classification criteria of each standard. Accordingly, for data conversion, it is necessary to identify interoperable classes and items through attribute and structural comparison, and to match classes and items that possess equivalent values. In this study, the conversion logic was organized into three mapping types, namely, direct semantic matching, structural reorganization, and default-value generation. Direct semantic matching was used when source and target items shared equivalent meanings such as name, description, or boundary-related spatial-reference information. Structural reorganization was required when TIN-based terrain geometry represented under TINRelief in CityGML had to be reorganized into LandSurface, Patches, and SimpleTrianglePatch in the LandXML structure. In addition, several target items required by the LandXML schema but not explicitly available in the source data were generated using predefined

Table 2
Classes defined on LandXML: Relief.

Class	Description
Land Feature	A class for defining the characteristics of land within the LandInfraDataSet
Land Element	Represents types and characteristics of land, such as embankments, riverbeds, and roads
Land Surface	Contains information for specifying the boundaries between different terrain surfaces
Land Layer	Includes properties and information about each terrain layer

default values to preserve schema completeness. On the basis of these principles, terrain-related classes and elements were derived from both standards, and semantically compatible items were matched to define the essential targets for conversion. Terrain in the CityGML standard consists of the following classes: Core, Relief Feature, Relief Component, TIN Relief, Mass Point Relief, Break Line Relief, and Raster Relief. Among these, Core, Relief Feature, and Relief Component are mandatory classes. The data structure is completed by including one or more of the specific terrain representation classes: TIN Relief, Mass Point Relief, Break Line Relief, or Raster Relief. Most CityGML data utilize TIN Relief to represent terrain surfaces. The terrain-related classes and elements of CityGML are presented in Table 3.

Terrain in the LandXML standard is structured around DataSet with other primary classes consisting of Land Feature, Land Element, Land Surface, and Land Layer. Among these, the DataSet, Land Feature, Land Element, and Land Surface are mandatory classes, whereas Land Layer is utilized when support for 2D forms is required. DataSet includes metadata serving as header information for the entire data to describe the relevant terrain information, and it contains elements such as features, DataSet IDs, and identifiers. Land Feature is a subtype of the Feature class within DataSet and includes Land Element, Land Surface, and Land Layer. Land Element is a subclass of Land Feature and includes elements such as unique identifiers and types for the land element. The terrain-related classes and elements of LandXML are presented in Table 4.

For conversion between the two data standards presented above, it is necessary to compare and match the attribute values of terrain-related classes and items, and to identify the essential classes and items required for the process. The essential classes and items for this conversion are presented in Table 5.

Table 3.
CityGML Relief classes.

Class	Elements			
Core	CityobjectMember Description	FeatureID Date	Name BoundedBy	Valid Identifier
Relief Feature	ReliefComponent	LoD		
Relief Component	TIN Relief Breakline Relief	MassPoint Relief LoD	Raster	Relief
TIN Relief	Triangulated Surface GML	Triangle Triangle Patches	Linearization	Exterior

Table 4
LandXML Relief class.

Class	Elements			
DataSet	Feature DataSetID	Name	DateTime	Description
Land Feature	LandElement LandSurface	LandLayer State	LandSurfaceID	Scope
Land Element	ElementID	ElementTyper	Material	
Land Surface	Spatial Representation	Patches Material	PosList	SimpleTrianglePatch

Table 5
Required classes and elements for conversion to LandXML.

Class	Elements (CityGML→ LandXML)	
Core→DataSet	CityObjectMember → FeatureDataSetID FeatureID → DataSetID Identifier → Identifier Description → Description	Date/Valid → DateTime DataSet Version → DataSet Version Language → Language BoundedBy → DefaultCRS
Relief Feature→ Land Feature	TINRelief → Spatial Representation TriangulatedSurface GML → LandSurfaceID LoD → Scope Exterior → State	Triangle Patches → Patches Triangle → SimpleTrianglePatch posList srsDimension → posList

The classes requiring conversion are Core and Relief Feature. Core is converted to DataSet, and Relief Feature to Land Feature. For Core, key elements such as Object, ID, Identifier, Name, Description, and Creation Date are mapped to the corresponding DataSet elements. Any mandatory LandXML elements not defined in CityGML are assigned default values. For Relief Feature, elements including TIN Relief, Relief Surface ID, LoD, and TIN are converted into the corresponding Land Feature elements.

5. Data Conversion Method

To convert data into LandXML format, terrain-related classes, components, and their corresponding attribute information must be included. Accordingly, we designed a sequential conversion process based on the essential classes and items outlined in Table 5. The workflow proceeds as follows. First, CityGML input data are read and terrain-related classes and elements are extracted. The corresponding attribute values are then retrieved at the element level. Next, the extracted source-side items are matched to LandXML target-side items in accordance with semantic equivalence and structural compatibility, and required target items not available in the source data are generated using predefined default values. Finally, the converted output is validated to confirm the completeness of all required classes, elements, and attribute values. If omission or errors are detected, the data are classified as unusable; otherwise, they are classified as valid converted terrain data. The overall CityGML to LandXML conversion process is illustrated in Fig. 1.

6. Conversion Results

6.1. Validation DataSet

For validation, the 2019 Hague city model provided by Delft University of Technology was used as the source DataSet. The original CityGML DataSet covers an area of approximately 787.36 × 671.85 × 13.48 m and includes one terrain DataSet composed of 5595 triangular surfaces, together with 844 building objects and 1653 building component objects. The total size of the original DataSet is approximately 22.7 MB. Because the scope of the present study is limited to terrain-oriented validation, only terrain-related information was extracted from the

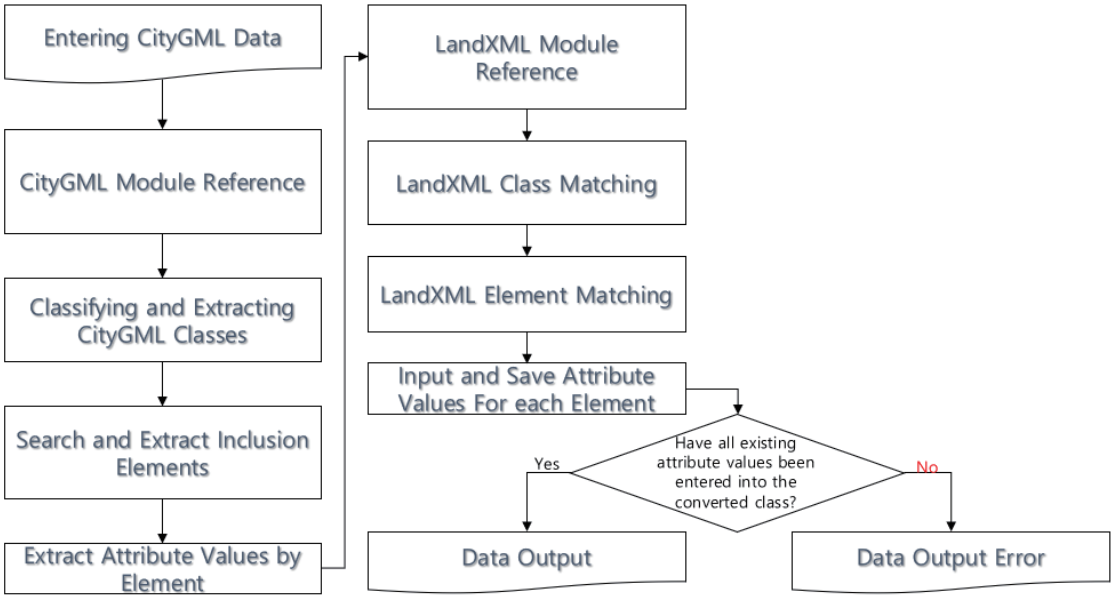

Fig. 1. (Color online) CityGML data before conversion, and the LandXML conversion process.

original city model. After extraction, the terrain-only source file had a size of approximately 1.71 MB. The extracted terrain data consisted of a single TIN Relief component at LoD 2, and each triangle was represented by coordinate values with srs Dimension=3. The source CityGML terrain data and the corresponding LandXML conversion process used for validation are shown in Fig. 2

6.2 Coordinate reference system handling

The coordinate reference system information was preserved during the conversion process by extracting the srsName attribute from the CityGML gml:Envelope element and transferring it to the defaultCRS item in the target structure. In the validation DataSet, CRS was defined as urn:ogc:def:crs:EPSG::28992, a projected coordinate system used in the Netherlands. The coordinate units are meters, and the coordinate axes represent easting (X), northing (Y), and elevation (Z). No geometric reprojection was performed during conversion. Instead, the original coordinate values were transferred directly from the CityGML posList to the corresponding LandXML posList entries. To validate geometric consistency, the source and converted data of the coordinate values of five randomly selected triangles were compared. The results of the comparison confirmed that the coordinates were transferred without numerical loss, and only the redundant closing vertex used in the source triangle description was removed in accordance with the target patch convention

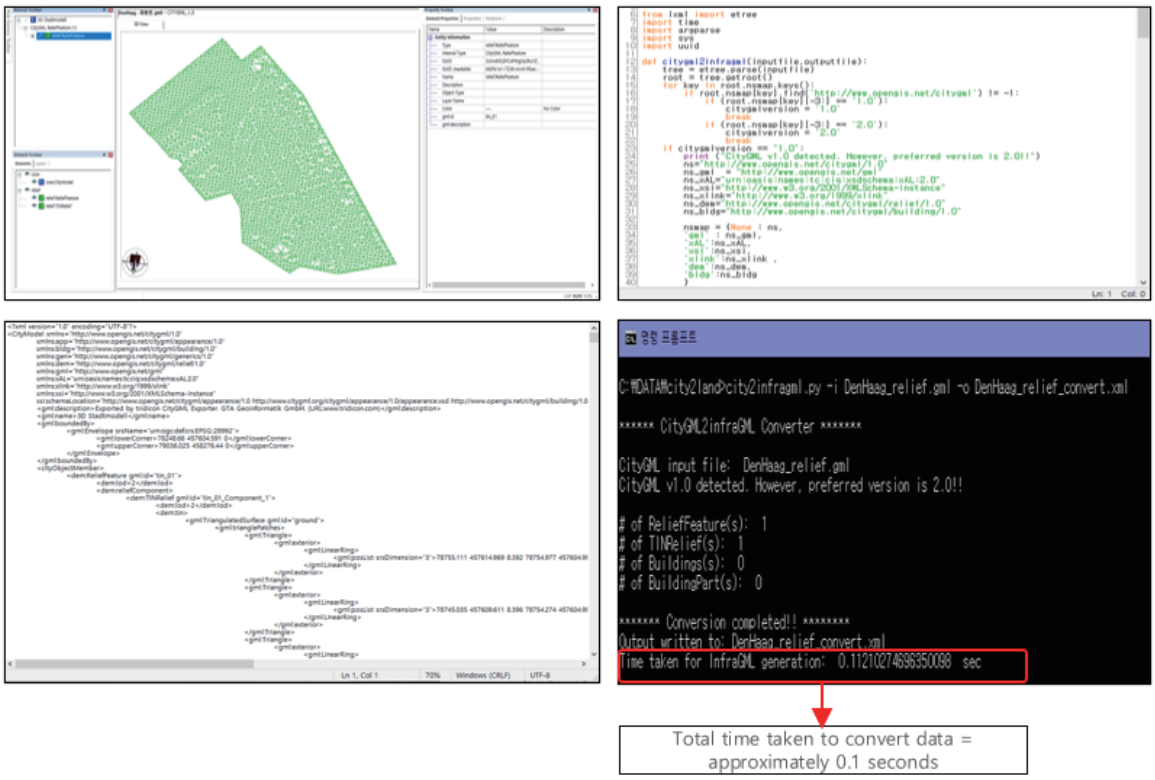

Fig. 2. (Color online) CityGML data before conversion and the LandXML conversion process.

6.3 Conversion performance and result analysis

To validate the conversion process proposed in Sect. 5, conversion and verification were conducted utilizing the extracted Hague terrain DataSet. The experimental results demonstrated a successful conversion from the CityGML format to the LandXML format, and it was confirmed that the converted data satisfied all essential classes and items defined in Sect. 4. The proposed method converted the extracted terrain DataSet from the CityGML format to the LandXML format in approximately 0.112 s. The converted file size was approximately 1.07 MB, compared with approximately 1.71 MB for the terrain-only CityGML input. This corresponds to a compression ratio of approximately 1.59:1 and suggests that the target schema can represent the terrain data in a more compact maintenance-oriented structure. Structural inspection of the converted output confirmed that all required target classes and elements were successfully generated. DataSet-level metadata, terrain-reference items, and patch-based terrain representation items required in the target structure were all present. In addition, no omission or attribute errors were detected in the required items. With 5595 triangles processed in 0.112 s, the effective throughput was approximately 49955 triangles per second, indicating that the methodology can be applied efficiently to larger terrain DataSets. Details of the conversion results are shown in Table 6, and the conversion results are shown in Fig. 3.

Table 6
Conversion results.

Item	Source data	Converted data	Result
Data Format	*.GML (CityGML)	*.XML (LandXML)	Converted
Geometry Type	One TIN-based terrain	One TIN-based terrain	Preserved
Surface Elements	5595	5595	Preserved
Data Capacity	About 1.71 MB	About 1.07 MB	Reduced
Time Required	—	About 0.112 s	Efficient
Required Items	Source items available	All target items generated	Satisfied
Attribute Errors	—	No errors detected	Acceptable
CRS Handling	EPSG:28992	Same CRS preserved	Consistent

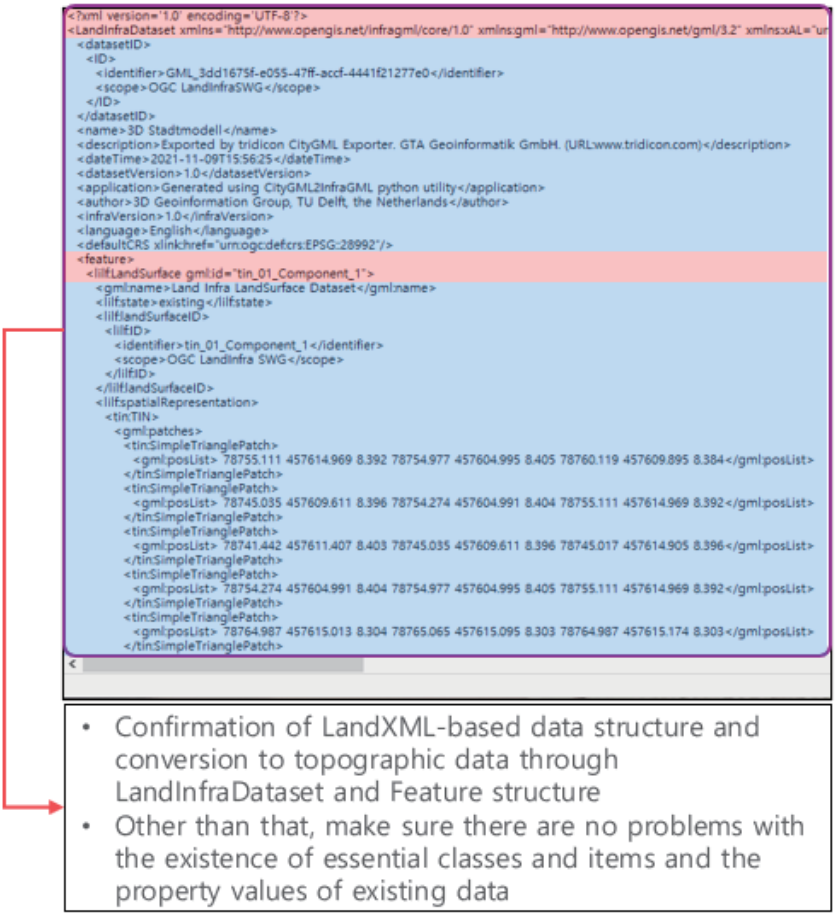

Fig. 3. (Color online) Conversion results.

7. Application Potential for Port Facility Data

In this study, the transformation methodology was validated primarily using topographic data because of limitations in obtaining 3D data at the individual facility level; however, the proposed approach can be structurally extended to port facility objects. To demonstrate the scalability of the proposed terrain-based workflow, a simple conceptual model of representative port structures, such as quays and breakwaters, was presented. In this example, object identifier,

Table 7
Example of mapping for port facility maintenance attributes.

Example Facility	CityGML-Side attribute	LandXML-Side target item
Common	gml:id	identifier
Common	class / function	elementType
Quay wall	measured material	material
Quay wall	installationYear	description or user-defined property
Quay wall	safetyGrade	description or user-defined property
Breakwater	material	material
Breakwater	inspectionDate	dateTime or user-defined property
Breakwater	maintenanceStatus	state

facility type, material, installation year, safety rating, and inspection date were considered as representative maintenance attributes. Although we could not validate transformation at the entire facility level owing to limitations in data availability, the proposed concept demonstrates how mapping logic can be extended from terrain reference data to facility maintenance objects. This example supports the applicability of the proposed workflow to port facility maintenance, extending beyond structural validation at the terrain level. An example of mapping is presented in Table 7.

8. Conclusions

In this study, we proposed and validated a terrain-oriented CityGML-to-landXML conversion workflow as a foundational process supporting port facility maintenance. The method derived terrain-related required classes and elements from both standards, matched semantically corresponding items, and reorganized the source data into a maintenance-oriented target structure through semantic matching, structural reorganization, and default-value supplementation. The validation results confirmed that all required target items were successfully generated without omission or attribute errors. The converted output preserved a TIN-based terrain structure represented by 5595 triangles, reduced the terrain file size from approximately 1.71 to 1.07 MB, and completed the conversion in approximately 0.112 s. These results demonstrate the technical feasibility and efficiency of the proposed conversion method. The significance of this study lies in its role as a foundational process for future maintenance-data integration because terrain functions as the spatial reference layer for facility objects. From the perspective of sensing applications, the proposed method also contributes to improving the structured usability of three-dimensional spatial data derived from LiDAR, UAV photogrammetry, mobile mapping, GNSS surveying, and related sensing technologies. However, the present validation is limited to terrain data. In future studies, the workflow should be extended to facility-level objects, and the conversion of maintenance attributes for representative port structures such as quay walls and breakwaters should be verified. Additional work is also needed to compare the method with other conversion approaches and to examine broader interoperability with related infrastructure standards.

References

1 T. W. Kang, C. H. Hong, J. R. Hwang, and H. S. Choi: Spat. Inf. Soc. **20** (2012) 91. https://doi.org/10.12672/ksis.2012.20.5.091
2 S. H. Jang: Master's thesis, Seokyeong University (2022).
3 OGC: OGC CityGML Encoding Standard, Version 2.0 (OGC 12-019) (2012). https://www.ogc.org (accessed November 2025).
4 OGC: OGC LandInfra/InfraGML Encoding Standard (OGC 16-104r1) (2017). https://www.ogc.org (accessed November 2025).
5 LandXML: LandXML Home. http://www.landxml.org (accessed November 2025).
6 T. W. Kang, H. S. Choi, J. R. Hwang, and C. H. Hong: J. Korean Soc. Surv. Geod. Photogramm. Cartogr. **30** (2012) 559. https://doi.org/10.7848/ksgpc.2012.30.6-1.559
7 S. H. Park, J. S. Han, R. Choi, and C. W. Cho: Korean J. Comput. Des. Eng. **24** (2019) 361. https://doi.org/10.7315/CDE.2019.361
8 J. R. Hwang, T. W. Kang, and C. H. Hong: Spat. Inf. Res. **20** (2012) 49. https://doi.org/10.12672/ksis.2012.20.5.049
9 M. El-Mekawy: Integrating BIM and GIS for 3D City Modelling: The Case of IFC and CityGML, Licentiate thesis, Royal Institute of Technology (KTH), Stockholm (2010).
10 B. K. Min, D. H. Park, Y. G. Jang, and I. J. Kang: J. Korean Cadastral Inf. Soc. **20** (2012) 83. https://doi.org/10.7319/kogsis.2012.20.3.083
11 T. W. Kang, J. H. Yoon, W. S. Lee, and H. S. Choi: J. Korean BIM Soc. **3** (2013) 21. https://doi.org/10.13161/kibim.2013.3.1.021
12 J. E. Kim, T. W. Kang, and C. H. Hong: J. Korea Acad.-Ind. Coop. Soc. **15** (2014) 5333. https://doi.org/10.5762/KAIS.2014.15.8.5333
13 J. R. Ryu and S. Y. Choo: Spat. Inf. Res. **22** (2014) 27. https://doi.org/10.12672/ksis.2014.22.5.027
14 A. Rafiee, E. Dias, S. Fruijtier, and H. Scholten: Procedia Environ. Sci. **22** (2014) 397. https://doi.org/10.1016/j.proenv.2014.11.037
15 J. R. Ryu: Ph.D thesis, Kyungpook National university (2016).
16 S. H. Park and J. Y. Lee: J. GIS Assoc. Korea **17** (2009) 277.
17 C. S. Woo, J. S. Yoon, J. I. Shin, and K. S. Lee: J. Korean Soc. For. Sci. **96** (2007) 251.
18 Delft University of Technology: 3D City Models, https://3d.bk.tudelft.nl (accessed November 2025).

About the Authors

Jae Myeong Kim received his Ph.D. degree from the University of Seoul in 2012. He is currently an associate professor in the Department of Urban Planning and Engineering at Seokyeong University. His research interests encompass automation and data related to smart construction, construction automation, and construction data platforms. (jm927k@skuniv.ac.kr)

Seung Ho Jang received his master's degree from Seokyeong University in 2022. He is currently working at the National Land DB Department of the Spatial Information Industry Promotion Institute. His research interests encompass indoor spatial information and quality verification. (sh.jang@spacen.or.kr)

Yeon Chan In received his master's degree from Seokyeong University in 2024. From 2024 to 2025, he worked at the National Land DB Department of the Spatial Information Industry Promotion Institute. He is currently a Ph.D. student in the Department of Urban Planning, Construction and Disaster Prevention Engineering at Seokyeong University. (jg525g@skuniv.ac.kr)

Dong Gun Lee received his master's degree from Seokyeong University in 2025. He is currently a Ph.D. student in the Department of Urban Planning, Construction and Disaster Prevention Engineering at Seokyeong University. (porminse5164@skuniv.ac.kr)