Future City Pilot 1 - Automating Urban Planning Using Web Processing Service Engineering Report

Table of Contents

Publication Date: 2017-10-20

Approval Date: 2017-08-17

Posted Date: 2017-06-27

Reference number of this document: OGC 16-099

Reference URL for this document: http://www.opengis.net/doc/PER/FCP1-autoUP

Category: Public Engineering Report

Editor: Mohsen Kalantari

Title: Future City Pilot 1 - Automating Urban Planning Using Web Processing Service Engineering Report

---

OGC Engineering Report

COPYRIGHT

LICENSE AGREEMENT

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD. THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to

indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable.

Abstract

Numerous and diverse technologies push cities towards open and platform-independent information infrastructures to manage human, natural, and physical systems. Future Cities Pilot 1 is an OGC interoperability initiative that aims to demonstrate how cities can begin to reap the benefits of open standards. This document reports how Web Processing Standard (WPS) of OGC was successfully used in automating urban planning processes. This document details the implementation of urban planning processes and rules concerning urban development approval processes.

Business Value

Spatial representation of land development proposals are often submitted in 2D paper/image/CAD formats. The 2D design of even complex high-rise land developments is a norm. Moreover, land development assessments are applied in isolation since the responsible agencies usually do not have the infrastructure that enables sharing spatial data and automating the assessment process. This report outlines how land development proposal assessments can be automated through the use of WPS. This report illustrates that the use of a range of open standards including WPS can benefit the multi-stakeholder land development process.

What does this ER mean for the Working Group and OGC, in general

This report is of interest for Web Processing Service SWG. It details an application of this standard in urban planning and land development assessment. The report confirms viability of WPS in urban planning and can be used as evidence-base for promoting the standard.

How does this ER relate to the work of the Working Group

This report declares how WPS is used to parametrize several spatial and non-spatial data requirements of urban planning. The WPS reported in this ER involve interaction with an urban planning rules engine and WFS.

Keywords

Urban Planning, Land Development, WPS, WFS, CityGML, IFC

Proposed OGC Working Group for Review and Approval

Web Processing Service and CityGML SWGs.

Chapter 1. Introduction

1.1. Scope

The Future Cities Pilot is an OGC initiative to demonstrate how cities can benefit from open standards in managing human, physical, and natural systems. The first pilot project (FCP1) aimed to show how 3D modeling open standards such as IFC and CityGML can be used in urban planning and management scenarios. The scenarios included 1) the application of IFC and CityGML in land development proposal assessment, 2) the integration of CityGML with dynamic data feeds such as sensor observations and the utility of CityGML in solar energy potential of buildings, and 3) the value of CityGML in modeling urban flooding.

These scenarios were proposed and sponsored by Institut National de l'Information Géographique et Forestière - IGN (France), Ordnance Survey Great Britain (UK), and virtualcitySYSTEMS GmbH (Germany). The pilot participants that developed the solutions for Scenario 1 was University of Melbourne (Australia), for Scenario 2 was Technical University of Munich (Germany) and for Scenario 3 was Remote Sensing Solutions, Inc. (U.S.A).

Scenario 1 involved two objectives. The first objective was to demonstrate that 3D data in IFC and CityGML can provide land development assessment process with better information than that of 2D spatial data. The second objective was to demonstrate how the conformance with urban planning rules can be enhanced and automated through the adoption of WPS.

This OGC® document addresses the second objective of Scenario 1 in which several urban planning rules based on WPS are designed and developed.[OGC 16-16-097], which is part of FCP1, addresses the first objective of Scenario 1.

1.2. Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Table 1. Contacts

Name	Organisation
Mohsen Kalantari	The University of Melbourne
Bart De Lathouwer	OGC

1.3. Future Work

No future work is planned to this document.

1.4. Foreward

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

Chapter 2. References

The following documents are referenced in this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

- OGC: OGC 02-058, OGC® Web Feature Service, 2002

- OGC: OGC 09-025r1, OGC® OpenGIS Web Feature Service 2.0 Interface Standard (also ISO 19142), 2010

- OGC: OGC 05-007r7, OGC® Web Processing Service, 2007

- OGC: OGC 12-009, OGC® OGC City Geography Markup Language (CityGML) Encoding Standard, 2012

- OGC: OGC 16-097, Future City Pilot 1 - Using IFC/CityGML in Urban Planning Engineering Report, 2017

- OGC: OGC 16-098, FCP1 Engineering Report, 2017

- buildingSMART: Industry Foundation Classes IFC2x Edition 3, Technical Corrigendum 1, http://www.buildingsmart-tech.org/ifc/IFC2x3/TC1/html/index.htm, 2007

Chapter 3. Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of OGC® Web Processing Service [OGC 14-065] shall apply.

3.1. Abbreviated terms

- AEC - Architecture Engineering Construction
- bSI - buildingSMART International
- BIM - Building Information Modelling
- CityGML - City Geography Markup Language
- COLLADA - COLLAborative Design Activity
- ER - Engineering Report
- FCP1 - Future City Pilot Phase 1
- GML - Geography Markup Language
- GUI - Graphical User Interface
- IFC - Industry Foundation Classes
- KML - Keyhole Markup Language
- LoD - Level of Detail
- OGC - Open Geospatial Consortium
- O&M - Observations & Measurements
- PB - Protocol Buffers
- UI - User Interface
- WAR - Web application ARchive
- WFS - Web Feature Service
- WPS - Web Processing Service
- XML - Extensible Markup Language

Chapter 4. Overview

This report outlines the use of open standards in automating urban planning process. It includes the following sections. Section "FCP1 and it requirements" sets the scene and identifies urban planning as one of the key areas that open spatial data infrastructures can be of value. Specific urban requirements of FCP1 and together with its case study data are introduced. Section "Conceptual Design" reports a requirements analysis of urban planning and identifies three areas of inefficiencies. This section then presents the design of a system based on open platforms to address these inefficiencies. Section "Automatic Urban Planning Demonstrator" details the implementation of the design including tools that are used and developed. The ER is then concluded in the "Conclusion" section.

Chapter 5. FCP 1 and its requirements

Digitally enhancing cities is essential for sustainable, prosperous, healthy and inclusive future for citizens by using advanced digital technologies. However, the digital enhancement requires diverse and numerous technologies that operate in space and time. The enhancement of cities cannot, therefore, be realised unless open standards platform is used for communicating spatial and temporal data.

Various cities around the world have successfully created 3D digital city models. These models have the potential to be used in several aspects of cities. For instance, current land development assessments can be significantly enhanced by using the 3D models of the proposed development and existing digital models. 3D models combined with real-time data of buildings' temperature and temperature change, energy and water use can provide information, knowledge and insight to enhance financial, environmental, and social outcomes for citizens living in cities. Or, 3D models can be used to better understand the dynamic of floods in cities and assessment of its damage on the buildings

Considering the potential of open standards in realising such scenarios, FCP 1 aimed at demonstrating standards-based location enabled information technology to advance a range of city services, improve governance and enable innovative citizen and consumer services. The pilot was required to prove the ability of spatial data infrastructures to support indicators of quality of life, civic initiatives and resilience. Several scenarios were considered in the pilot including urban planning, social services and urban resilience. This following outlines the FCP 1's urban planning requirements.

5.1. Urban Planning Requirements

The use of BIM models encoded in IFC will become mandatory for major land development projects. The FCP 1's urban planning requirement, therefore, included the use of BIM in the development approval process. It was required to automatically validate the proposed development against urban planning rules. In particular, it was required to use contextual information such as the surrounding city model, cadastre and road network to facilitate the validation process. Concerning the city models, it was required to consider mapping IFC to various levels of detail of CityGML including LoD 2, LoD 3 and LoD 4. It was also required to provide facilities for human inspection and verification of proposal such that urban planners can view the building project within the existing 3D model of the city and store in databases.

5.2. Data

The following dataset was provided for urban planning scenario

- BIM model (IFC) of the proposed development
- Existing CityModel of the area of interest (CityGML profile, according to Ref3DNat recommendations, in LOD2 – or 3 in close future – with textures)
- Roads and buildings footprint.
- Cadastre

Chapter 6. Conceptual Design

6.1. Requirements Analysis

Urban planning systems aim to establish affordable, socially-inclusive, environmentally-friendly environments for human settlement. Urban planning is an interplay of place, people, and purpose. Planning for where to live, work, and enjoy is therefore inherently a spatial process. The use of geospatial information system in urban planning is now well established. The use of small and large scale geospatial data such as planning zones, green areas, street networks, and property boundaries is an indispensable part of the planning processes for land development.

In urban planning, land development is considered to be a multi-stakeholder process, in which several agencies are involved. The agencies may include privately operated land surveying, land development, or architecture design firms or publicly-funded organizations such as municipalities, road authorities, land registries, or mapping agencies. Each agency plays a unique role in the land development approval process, but the process itself is about preparing development proposals and the application of rules and criteria defined in the regulation, some of which are location-dependent, against the proposals. Where the rules are location-dependent, the agencies use spatial data for assessing land development proposals. Spatial representation of development proposals may be prepared and designed using CAD or GIS software solutions but are often submitted in paper/image formats for assessment.

Some technology inefficiencies are evident in this process. First, even if proposals are submitted as data, the format may not be readily usable by processing agencies (1). Second, the 2D design of even complex high-rise land developments is a norm in representing proposals (2). Third, land development rules are applied in isolation since the responsible agencies usually do not have the infrastructure that enables sharing spatial data (3a) and automating rules (3b).

6.2. System Architecture

An open platform design is presented in this report to help, firstly, with transforming proposed development data into a format that is usable by several processing agencies (inefficiencies 1 and 3b) and secondly, with integrating data from several agencies for applying location-dependent rules against the proposal (inefficiency 3a). [OGC 16-16-097], which is part of FCP1, addresses (inefficiency 1 and 3) in a separate report.

The design enables execution of data transformation and location-dependent urban planning rules functionalities across a network of stakeholders by providing standardized inputs and outputs. The design involves four layers: User Interface, Logic, Service, and Data (Figure 1). The layers include 11 components including development approval workflow user interface, several database systems capable of managing various data formats, urban planning rules engine, data transformation engine, and web services for accessing data and processing.
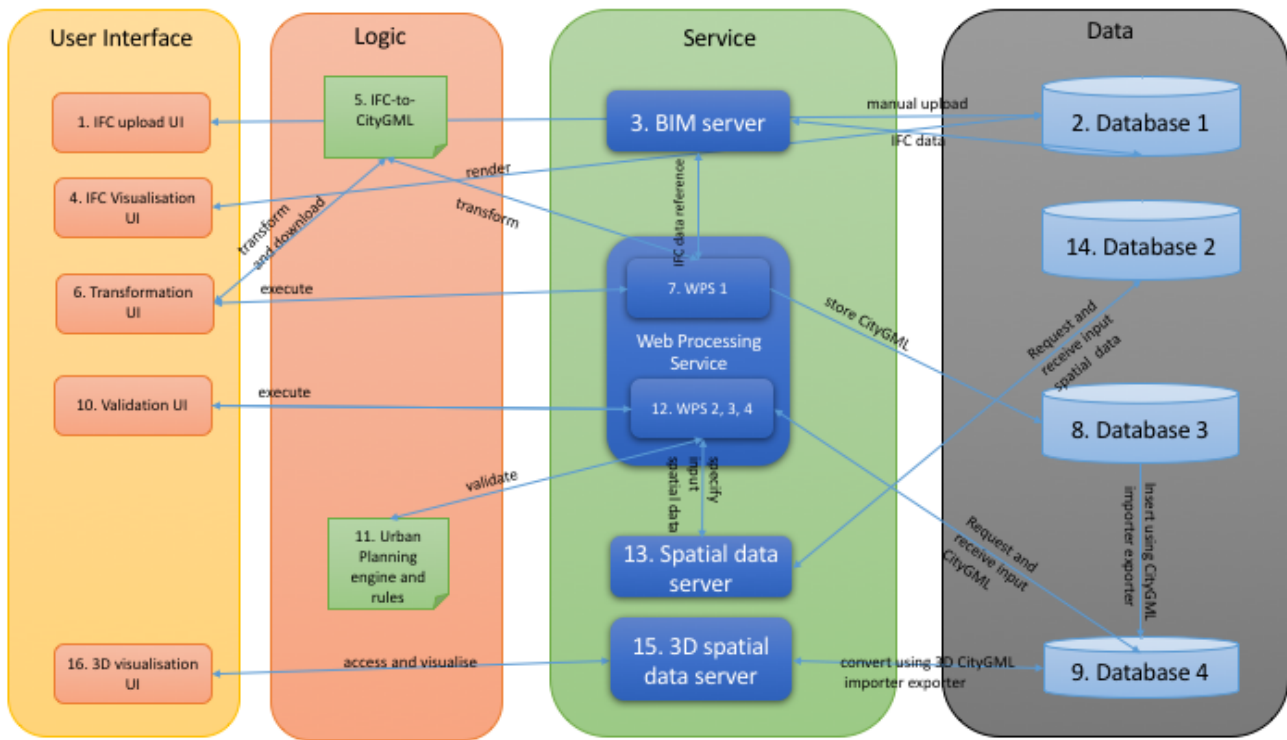
Figure 1: System Architecture

Of significance to this ER, to address inefficiencies 1 and 3, a number of WPS based rules were designed for demonstration including 1) transforming IFC to CityGML, 2) checking building height , 3) checking building location, and 4) checking building footprint.

The components are described as below.

- Component 1 is a part of the UI and is a web client that enables creating projects and uploading IFC files.
- Component 2 is a database system that stores the content of IFC files.
- Component 3 is a web service to provide access to IFC files of Component 2.
- Component 4 is another aspect of the UI which is a web client for rendering and inspecting an IFC file and its elements.
- Component 5 is the IFC to CityGML transformation function.
- Component 6 is a web client to enable users to call Component 5 and download the resulted CityGML or execute WPS 1.
- Component 7 (WPS 1) is a WPS based function that calls Component 5 and stores the resulted CityGML in a file server which is Component 8.
- Component 9 is a database to store the CityGML data of the proposed development and existing city model.
- Component 10 is the urban planning validation UI that allows executing urban planning rules against the development proposal data.
- Component 11 is the urban planning rules engine.
- Component 12 (WPS 2, 3, and 4) includes WPS based functions for specifying inputs and output parameters in executing urban planning rules using Component 11.

- Component 13 is to access non-CityGML spatial data that are stored in Component 14 which is a spatial database.

- Component 15 is a 3D data web service.

- Component 16 is a 3D data web client for rendering the proposed development together with data about the surrounding environment.

Chapter 7. FCP 1 - Automatic Urban Planning Demonstrator

In this section of the ER, the implementation of the system is detailed. First, the software components that were utilized are listed, and their utility is introduced. Then, the properties of the WPS based functions including their purpose, input, and output parameters are outlined.

7.1. Software components

BIMServer: The Building Information Model server (BIMserver.org) platform enables creating web based applications for BIM on the open standard IFC. IFC data are interpreted by a smart core and stored as objects in an underlying database. BIMserver is based on plugins in an open framework. The BIMserver software is free and open source (GNU Affero GPL) (BIMserver, 2017). It can support dynamic collaboration processes in urban planning where several players such as land developers, land registration officers, mapping agencies, and urban planners play a role. It has core server features like revisions, authorization, compare, query, model checking, merging, etc. BIMserver has open interfaces and network protocols (SOAP, PB, JSON), uses open standards, is built as a plugin framework for easy fine-tuning, has an admin configuration GUI, and developers documentation and SDKs. The core of the software is based on the open standard IFC. Components 1 to 5 of the demonstrator are built on BIMserver and its plug-ins. Screen captures of Components 1 and 4 are presented in Figures 2 and 3.
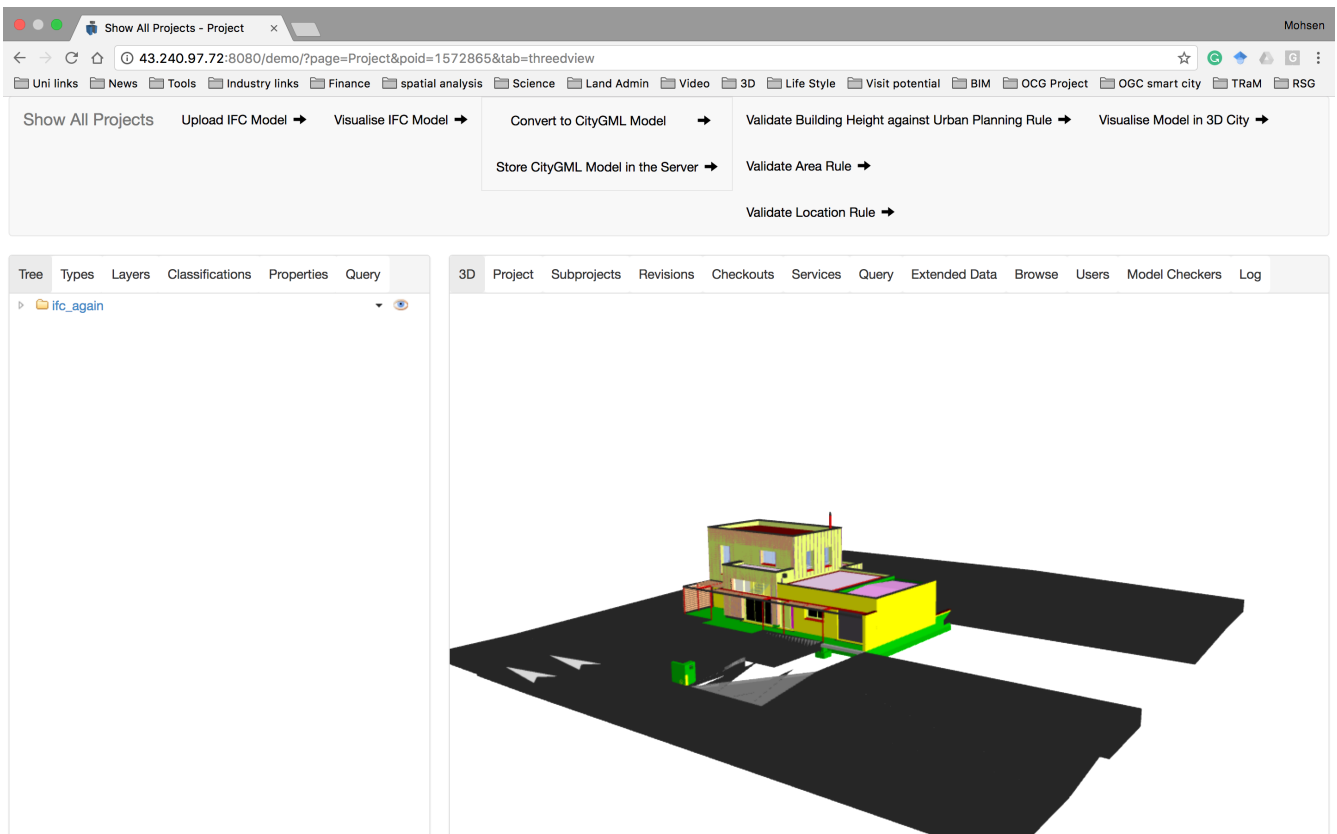

Figure 2: UI to upload IFC file.

Figure 3: UI to render IFC file

52°North Web Processing Service: This component enables the deployment of standardized geo-processing on the web (52°North, 2017). It features a pluggable architecture for processes and data encodings. It is used to implement components 7 and 12 which are about WPS 1, 2, 3, and 4 for automating development assessment proposal. In this project, the WPS version 1.0.0 has been used.

GeoServer: This is a software server that allows viewing and editing geospatial data based on OGC standards. In implementing the demonstrator, GeoServer's Web Feature Service (WFS) capability is used to serve the topographic and cadastral data that was required in WPS based urban planning functions. In this demonstrator, WFS implementation 1.0.0 is used.

3D City Database: This is a free 3D geo database to store, represent, and manage virtual 3D city models on top of a spatial relational database based on CityGML schema (3DCityDB, 2017). In this project, it has been used to store the existing city model of the study area, and the resulted CityGML.

3D City Database Importer/Exporter: This is an application interface for the 3D City Database (3DCityImporterExporter, 2017). This application offers a wide range of configuration options (e.g., based on the schema, feature id, bounding box) to validate and import CityGML files into the 3D City Database. It also allows exporting the content of the database to file format such as KMZ, KML, or COLLADA. When exporting, several aspects of the data such as LoD, bounding box, and appearance can be configured. This tool has been extensively used in preparing data for this demonstrator.

3D City Database Web Feature Service: This service allows web-based access to 3D City Database. It is a platform-independent, database-independent service that is provided as a Java WAR and Java libraries that render necessary dependencies for the WFS service (3DCityWFS, 2017). This service was used to access the CityGML data that was required in parameterizing data requirements of

WPS 3 and 4.

3D-DB-Web-Map-Client: This is a web client from 3D visualization (3DCityWebClient, 2017). It is developed based on Cesium Virtual Globe with WebGL as the underlying technology. This component has been customized to visualize the proposed development and surrounding area. It is important to note that the spatial data of interest were converted to KML and KMZ for visualization by 3D-DB-Web-Map-Client. Screenshots of this UI are presented in Figures 4 and 5.
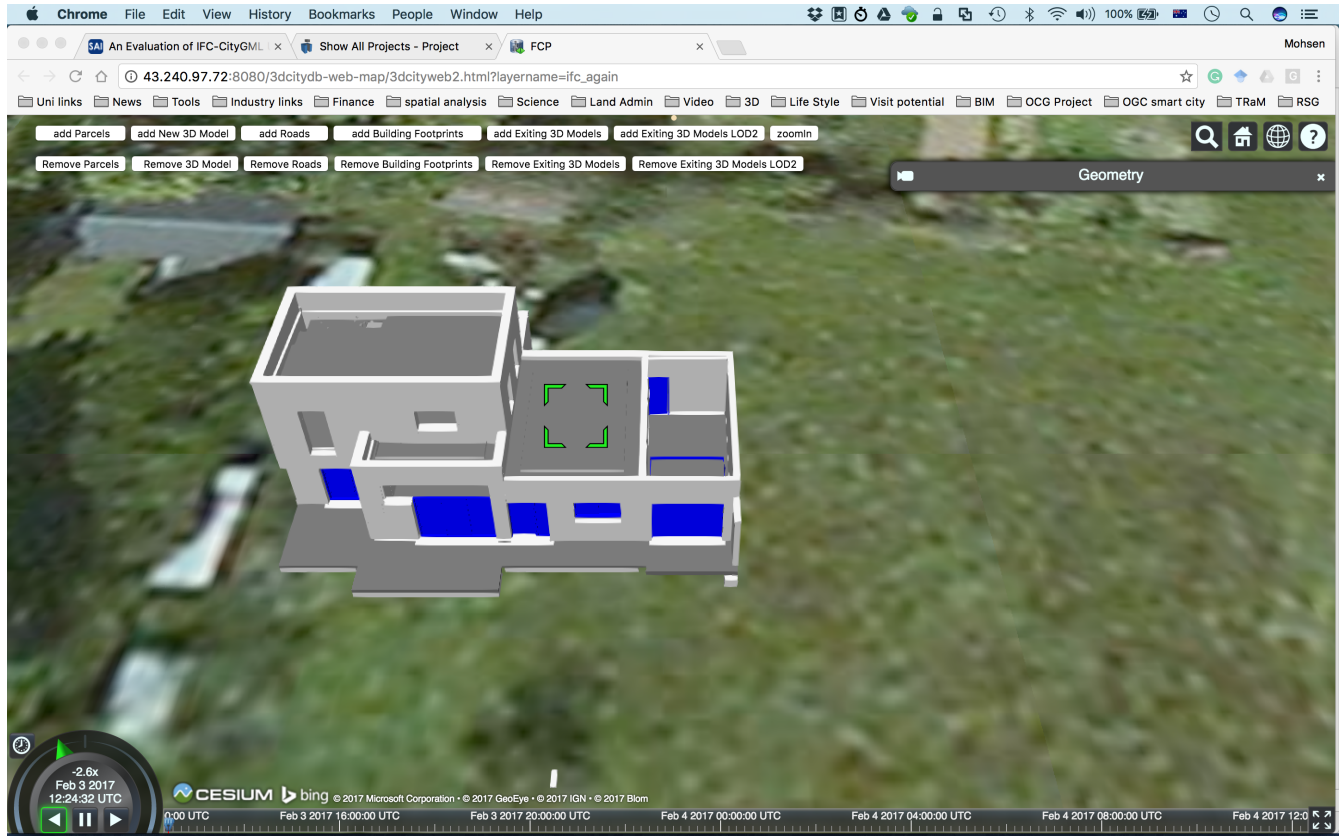
Figure 4:CityGML after conversion from IFC in 3D-DB-Web-Map-Client
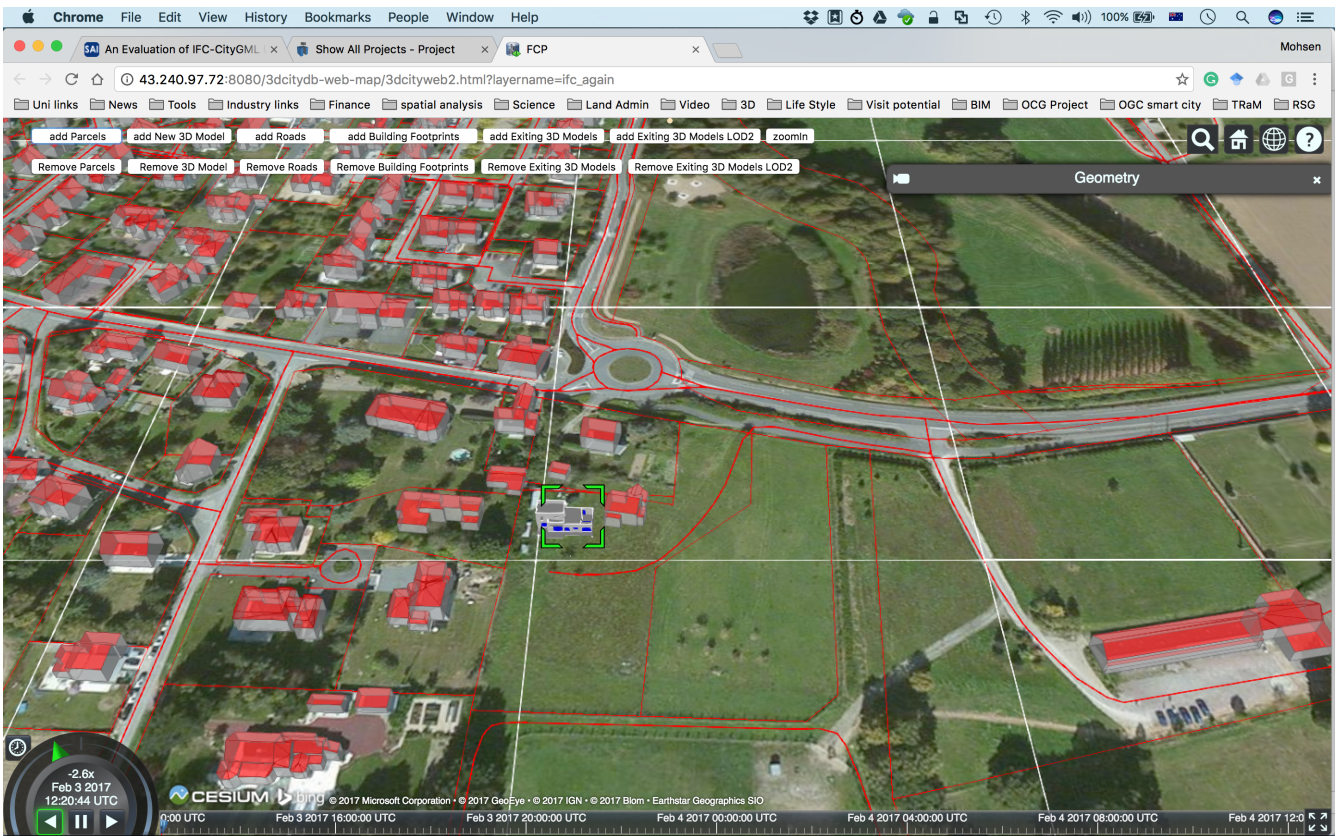
Figure 5:CityGML after conversion from IFC and its surrounding in 3D-DB-Web-Map-Client

7.2. Implementation

This section lists the WPS based functions, describes their purpose, details input and output parameters, and provides request and output examples for each.

7.2.1. WPS 1

Name: WPS BIMGenerateAndPersist

Purpose: To execute IFC to CityGML Transformation

Description: This service calls the IFC to CityGML transformation module of the demonstrator. It carries the name of the project that contains the IFC file as the input parameter and returns access url of the resulted CityGML.

Input: Name of IFC file project in BIMserver

Output: Access URL of generated CityGML file

Request example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wps:Execute service="WPS" version="1.0.0"
xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsExecute_request.xsd">

<ows:Identifier>org.n52.wps.server.algorithm.test.BIMGenerateAndPersistGML</ows:Identi
fier>
  <wps:DataInputs>

        <wps:Input>
            <ows:Identifier>ProjectNameInBIMServer</ows:Identifier>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">IFC file name</wps:LiteralData>
            </wps:Data>
        </wps:Input>
  </wps:DataInputs>
  <wps:ResponseForm>
    <wps:ResponseDocument status="false" storeExecuteResponse="false">
      <wps:Output asReference="false">
        <ows:Identifier>GeneratedGMLAccessURL</ows:Identifier>
      </wps:Output>
            <wps:Output asReference="false">
                <ows:Identifier>Status</ows:Identifier>
            </wps:Output>
```

Output Example:

```
<wps:ExecuteResponse xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ows="http://www.opengis.net/ows/1.1"xsi:schemaLocation="http://www.opengis.net/w
ps/1.0.0 http://schemas.opengis.net/wps/1.0.0/wpsExecute_response.xsd"
serviceInstance="http://localhost:8080/wps/WebProcessingService?REQUEST=GetCapabilitie
s&SERVICE=WPS" xml:lang="en-US" service="WPS" version="1.0.0">
    <wps:Process wps:processVersion="1.0.0">

<ows:Identifier>org.n52.wps.server.algorithm.test.BIMGenerateAndPersistGML</ows:Identi
fier>

<ows:Title>org.n52.wps.server.algorithm.test.BIMGenerateAndPersistGML</ows:Title>
    </wps:Process>
    <wps:Status creationTime="2017-02-10T08:38:15.768Z">
        <wps:ProcessSucceeded>Process successful</wps:ProcessSucceeded>
    </wps:Status>
    <wps:ProcessOutputs>
        <wps:Output>
            <ows:Identifier>GeneratedGMLAccessURL</ows:Identifier>
            <ows:Title>GeneratedGMLAccessURL</ows:Title>
            <wps:Data>
                <wps:LiteralData
dataType="xs:string">http://43.240.97.72:8080/gendata/citygml/ifc_again.gml</wps:Liter
alData>
            </wps:Data>
        </wps:Output>
        <wps:Output>
            <ows:Identifier>Status</ows:Identifier>
            <ows:Title>Status</ows:Title>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">Generated
Successfully</wps:LiteralData>
            </wps:Data>
        </wps:Output>
    </wps:ProcessOutputs>
</wps:ExecuteResponse>
```

7.2.2. WPS 2

Name: Validate Building Height

Purpose: To execute height validation rule of the urban planning rules engine

Description: This service checks the height of the building in the CityGML file and compares it against the threshold specified by the user. It carries the access URL of the CityGML file and the height limit as input parameters. It returns a message including the calculated height of the building and whether the proposal conforms to the rules.

Input parameters: Access URL of generated CityGML file and height threshold

Output parameter: Result of validation

Request example:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wps:Execute service="WPS" version="1.0.0"
  xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
  xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsExecute_request.xsd">
  <ows:Identifier>org.n52.wps.server.algorithm.test.BIMValidateGML</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier
xmlns:ns1="http://www.opengis.net/ows/1.1">GMLFileAccessURL</ows:Identifier>
      <wps:Reference
        xlink:href="http://43.240.97.72:8080/gendata/citygml/ifc_again.gml"
        mimeType="text/plain" />
    </wps:Input>

     <wps:Input>
          <ows:Identifier>HeightThreshold</ows:Identifier>
          <wps:Data>
              <wps:LiteralData dataType="xs:string">6</wps:LiteralData>
          </wps:Data>
      </wps:Input>
  </wps:DataInputs>

  <wps:ResponseForm>
    <wps:ResponseDocument status="false" storeExecuteResponse="false">
      <wps:Output asReference="false">
        <ows:Identifier>ValidationResult</ows:Identifier>
      </wps:Output>
      <wps:Output asReference="false">
        <ows:Identifier>ValidationResultDetail</ows:Identifier>
      </wps:Output>

    </wps:ResponseDocument>
```

Output example

```xml
<wps:ExecuteResponse xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ows="http://www.opengis.net/ows/1.1"xsi:schemaLocation="http://www.opengis.net/w
ps/1.0.0 http://schemas.opengis.net/wps/1.0.0/wpsExecute_response.xsd"
serviceInstance="http://localhost:8080/wps/WebProcessingService?REQUEST=GetCapabilitie
s&SERVICE=WPS" xml:lang="en-US" service="WPS" version="1.0.0">
```

```
    <wps:Process wps:processVersion="1.0.0">

<ows:Identifier>org.n52.wps.server.algorithm.test.BIMValidateGML</ows:Identifier>
        <ows:Title>org.n52.wps.server.algorithm.test.BIMValidateGML</ows:Title>
    </wps:Process>
    <wps:Status creationTime="2017-02-10T08:43:41.714Z">
        <wps:ProcessSucceeded>Process successful</wps:ProcessSucceeded>
    </wps:Status>
    <wps:ProcessOutputs>
        <wps:Output>
            <ows:Identifier>ValidationResult</ows:Identifier>
            <ows:Title>ValidationResult</ows:Title>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">Building height doesn't follow PLU
rules (greater than 6), buildingHeight is 6.988655716180801</wps:LiteralData>
            </wps:Data>
        </wps:Output>
        <wps:Output>
            <ows:Identifier>ValidationResultDetail</ows:Identifier>
            <ows:Title>ValidationResultDetail</ows:Title>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">bldg:FloorSurface : 53 zMinimList
[0.0, -0.5399999916553497, -0.3999999761581421, -0.5999999940395355, -1.0, -1.0,
-0.5999999940395355, -1.0, -1.0, -0.5399999916553497, -0.5999999940395355,
-0.5999999940395355, -0.5399999916553497, -0.3199999928474426, -0.3199999928474426,
-0.1600000262260437, -0.3399999737739563, 2.859999895095825, -0.20000000298023224,
-0.12999999523162842, 2.1500000953674316, 2.0799999237060547, 2.0999999046325684,
2.1500000953674316, 2.1500000953674316, 3.1500000953674316, 2.1500000953674316,
2.1500000953674316, 2.1500000953674316, 2.1500000953674316, 2.1500000953674316,
2.680000066757202, 2.680000066757202, 2.450000047683716, 2.450000047683716,
2.450000047683716, 3.285000858306885, 2.5999999046325684, 2.5799999237060547,
2.5799999237060547, 2.5799999237060547, 2.900000953674316, 3.059999942779541,
2.880000114440918, 3.0399999618530273, 5.2200000286102295, 5.15000009367432,
5.15000009367432, 5.15000009367432, 5.15000009367432, 5.15000009367432,
5.15000009367432, 5.4000000953674325] zMinimum -1.0 bldg:RoofSurface : 24 zMaxList
[0.0, 3.5663058310747147, 3.5663058310747147, 3.5663058310747147, 3.5663058310747147,
3.5663058310747147, 3.5663058310747147, 3.8622469566762447, 3.8622469566762447,
3.5663058310747147, 3.5663058310747147, 3.5663058310747147, 3.5663058310747147,
3.5663058310747147, 3.5663058310747147, 3.8622469566762447, 5.988655716180801,
5.988655716180801, 5.988655716180801, 5.988655716180801, 5.988655716180801,
5.988655716180801, 5.988655716180801, 5.988655716180801] zMaximum
5.988655716180801</wps:LiteralData>
            </wps:Data>
        </wps:Output>
    </wps:ProcessOutputs>
</wps:ExecuteResponse>
```

7.2.3. WPS 3

Name: Validate ratio of building area to land area.

Purpose: To execute building footprint of the urban planning rules engine.

Description: This service computes the ratio of the proposed building footprint to the area of the land parcel containing it. It carries the WFS get feature requests of the proposed development and parcel data as input parameters. It also includes a user specified threshold for the ratio. It returns a message including the calculated ratio and whether the proposal conforms to the rules.

Input parameters: WFS get feature requests of the footprint of the proposed development (empowered by virtualCitySystems) and parcel data (empowered by Geoserver).

Output parameter: Result of validation

Request example:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wps:Execute service="WPS" version="1.0.0"
  xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
  xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsExecute_request.xsd">
  <ows:Identifier>org.n52.wps.server.algorithm.test.BIMValidateCESGML</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>ParcelWFSFeatureAccessURL</ows:Identifier>
      <wps:Data>

<wps:LiteralData><![CDATA[http://144.6.224.184:8080/geoserver/OGCPROJ/ows?service=WFS&
version=1.0.0&request=GetFeature&typeName=OGCPROJ:parcelle&featureid=4960&outputFormat
=json]]></wps:LiteralData>
      </wps:Data>
    </wps:Input>

     <wps:Input>
            <ows:Identifier>AreaThreshold</ows:Identifier>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">0.4</wps:LiteralData>
            </wps:Data>
        </wps:Input>
    <wps:Input>
      <ows:Identifier>FootprintWFSFeatureAccessURL</ows:Identifier>
      <wps:Data>

<wps:LiteralData><![CDATA[http://fcp1.virtualcitymap.de:8080/mse/wfs?SERVICE=WFS&VERSI
ON=2.0.0&REQUEST=GetFeature&STOREDQUERY_ID=urn:ogc:def:query:OGC-
WFS::GetFeatureById&ID=GML_7b1a5a6f-ddad-4c3d-a507-3eb9ee0a8e68]]></wps:LiteralData>
      </wps:Data>
    </wps:Input>
  </wps:DataInputs>
```

```
  <wps:ResponseForm>
    <wps:ResponseDocument status="false"
      storeExecuteResponse="false">
      <wps:Output asReference="false">
        <ows:Identifier>ValidationResult</ows:Identifier>
      </wps:Output>
      <wps:Output asReference="false">
        <ows:Identifier>ValidationResultDetail</ows:Identifier>
      </wps:Output>

    </wps:ResponseDocument>

  </wps:ResponseForm>
</wps:Execute>
```

Output example

```
<wps:ExecuteResponse xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ows="http://www.opengis.net/ows/1.1"xsi:schemaLocation="http://www.opengis.net/w
ps/1.0.0 http://schemas.opengis.net/wps/1.0.0/wpsExecute_response.xsd"
serviceInstance="http://localhost:8080/wps/WebProcessingService?REQUEST=GetCapabilitie
s&SERVICE=WPS" xml:lang="en-US" service="WPS" version="1.0.0">
    <wps:Process wps:processVersion="1.0.0">

<ows:Identifier>org.n52.wps.server.algorithm.test.BIMValidateCESGML</ows:Identifier>
        <ows:Title>org.n52.wps.server.algorithm.test.BIMValidateCESGML</ows:Title>
    </wps:Process>
    <wps:Status creationTime="2017-02-10T08:56:00.120Z">
        <wps:ProcessSucceeded>Process successful</wps:ProcessSucceeded>
    </wps:Status>
    <wps:ProcessOutputs>
      <wps:Output>
        <ows:Identifier>ValidationResult</ows:Identifier>
        <ows:Title>ValidationResult</ows:Title>
        <wps:Data>
          <wps:LiteralData dataType="xs:string">CES rule (Coefficient d'Emprise au
Sol) is followed for this building in relation to its parcel. 0.03 <=
0.4</wps:LiteralData>
        </wps:Data>
      </wps:Output>
      <wps:Output>
        <ows:Identifier>ValidationResultDetail</ows:Identifier>
        <ows:Title>ValidationResultDetail</ows:Title>
        <wps:Data>
          <wps:LiteralData dataType="xs:string">Parcel area in sqm
is:5153.227294921875, Footprint area in sqm is:179.220703125, cesCalculated
is:0.03,</wps:LiteralData>
        </wps:Data>
      </wps:Output>
    </wps:ProcessOutputs>
</wps:ExecuteResponse>
```

7.2.4. WPS 4

Name: Validate building is within land boundary

Description: This service computes the location of the proposed building footprint to establish if it is contained by the land parcel. It carries the WFS get feature requests of the proposed development footprint and parcel data as input parameters. It returns a message whether the proposal conforms to the rules.

Purpose: To execute building location rule of the urban planning rules engine

Input parameters: WFS get feature requests of the foot print of the proposed development and parcel data

Output parameter: Result of validation

Request example:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wps:Execute service="WPS" version="1.0.0"
  xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
  xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsExecute_request.xsd">

<ows:Identifier>org.n52.wps.server.algorithm.test.BIMValidatePolyInPolyGML</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>ParcelWFSFeatureAccessURL</ows:Identifier>
      <wps:Data>

<wps:LiteralData><![CDATA[http://144.6.224.184:8080/geoserver/OGCPROJ/ows?service=WFS&
version=1.0.0&request=GetFeature&typeName=OGCPROJ:parcelle&featureid=4960&outputFormat
=json]]></wps:LiteralData>
      </wps:Data>
    </wps:Input>
    <wps:Input>
      <ows:Identifier>FootprintWFSFeatureAccessURL</ows:Identifier>
      <wps:Data>

<wps:LiteralData><![CDATA[http://fcp1.virtualcitymap.de:8080/mse/wfs?SERVICE=WFS&VERSI
ON=2.0.0&REQUEST=GetFeature&STOREDQUERY_ID=urn:ogc:def:query:OGC-
WFS::GetFeatureById&ID=GML_7b1a5a6f-ddad-4c3d-a507-3eb9ee0a8e68]]></wps:LiteralData>
      </wps:Data>
    </wps:Input>
  </wps:DataInputs>

  <wps:ResponseForm>
    <wps:ResponseDocument status="false"
      storeExecuteResponse="false">
      <wps:Output asReference="false">
        <ows:Identifier>ValidationResult</ows:Identifier>
      </wps:Output>
      <wps:Output asReference="false">
        <ows:Identifier>ValidationResultDetail</ows:Identifier>
      </wps:Output>

    </wps:ResponseDocument>

  </wps:ResponseForm>
</wps:Execute>
```

Output example

```
<wps:ExecuteResponse xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ows="http://www.opengis.net/ows/1.1"xsi:schemaLocation="http://www.opengis.net/w
ps/1.0.0 http://schemas.opengis.net/wps/1.0.0/wpsExecute_response.xsd"
serviceInstance="http://localhost:8080/wps/WebProcessingService?REQUEST=GetCapabilitie
s&SERVICE=WPS" xml:lang="en-US" service="WPS" version="1.0.0">
    <wps:Process wps:processVersion="1.0.0">

<ows:Identifier>org.n52.wps.server.algorithm.test.BIMValidatePolyInPolyGML</ows:Identi
fier>

<ows:Title>org.n52.wps.server.algorithm.test.BIMValidatePolyInPolyGML</ows:Title>
    </wps:Process>
    <wps:Status creationTime="2017-02-10T08:58:45.327Z">
        <wps:ProcessSucceeded>Process successful</wps:ProcessSucceeded>
    </wps:Status>
    <wps:ProcessOutputs>
        <wps:Output>
            <ows:Identifier>ValidationResult</ows:Identifier>
            <ows:Title>ValidationResult</ows:Title>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">Building footprint polygon is
contained by Parcel polygon</wps:LiteralData>
            </wps:Data>
        </wps:Output>
        <wps:Output>
            <ows:Identifier>ValidationResultDetail</ows:Identifier>
            <ows:Title>ValidationResultDetail</ows:Title>
            <wps:Data>
                <wps:LiteralData dataType="xs:string">parcelCoord result
:[[Ljava.lang.Double;@110f080a, footprintCoord result :[[Ljava.lang.Double;@114c4197,
polygonInPolygon result :23,</wps:LiteralData>
            </wps:Data>
        </wps:Output>
    </wps:ProcessOutputs>
</wps:ExecuteResponse>
```

Chapter 8. Conclusions and future work

8.1. Conclusions

Digital enhancement of cities requires technologies that can record and represent temporal and spatial dynamics of their natural, human, and physical system.

Collecting, recording, and maintaining higher-dimensional spatial data such as 3D data of the cities are vital with the growing dominance of multi-story buildings.

The availability of several open spatial data infrastructures for dealing with 3D data has become evident as a result of developing this demonstrator. The open infrastructures that are utilized in this project are freely available, well supported, and more importantly, developed based on open standards such as IFC, CityGML, WPS, and WFS. Such advancement in the open platforms development paints a promising future for the fast adoption of open standards in cities that use numerous and diverse technology for digital enhancement.

In this project, the WPS 1.0.0 was successfully used in the urban planning processes. Such a use case demonstrates that processing the spatial extent of stratified spaces and their impact on the urban built environment could become automatic when using a 3D representation of the data. Automatic processing of multi-store building proposals and surroundings that is based on WPS would decrease the amount of effort for urban planners to establish economic, social, environmental, and legal implications of the developments.

8.2. Future works

This project demonstrated the utility of OGC's WPS 1.0.0 standard in the domain of urban planning. The project showed that a range of 2D, 3D spatial and non-spatial factors could be parameterized in the WPS based processing.

However, several areas were identified that could constitute the future work for the Future City Pilot project. The future work can consider the following interoperability considerations.

1. Using WPS for implementing the rule concerning the distance between the proposed building and the street.

2. Using Simplu3D-rules Web API. This project could not utilize the API due to other priority tasks of IGN. The project could not also adopt Simplu3D developed based on IGN openmsource library of Geoxygen due to constraints on resources and other project priorities.

This level of standardization is significant as a multi-stakeholder operation such as urban planning needs standard-based infrastructures that facilitate sharing spatial data and location-dependent processing.

While successful, the demonstration of the WPS utility was undertaken for urban planning requirements that in nature were static. Managing cities go beyond examining whether a building proposal meets the immediate regulatory requirements. It is now expected that geospatial standards can assist with modeling and simulating the future impacts of land and infrastructure

developments in the cities. At the same time, the incorporation of time-series data in the city models lays the groundwork for processing live data that is collected in the cities.

The potential of WPS standard can be investigated in application domains such as urban growth, emergency management, and event planning that requires dynamic data (OGC SWE) or relies on simulation and modeling (OGC CDB) data.

Appendix A: Revision History

Table 2. Revision History

Date	Editor	Release	Primary clauses modified	Descriptions
August 15, 2016	M. Kalantari	0.1	all	Table of content
February 10, 2017	M. Kalantari	0.2	all	Initial version
February 11, 2017	M. Kalantari	0.3	all	Draft version
February 28, 2017	M. Kalantari	0.4	all	Considering feedback by IGN and Project Manager
March 9, 2017	M. Kalantari	0.5	all	Considering second round of feedback by IGN
March 21, 2017	M. Kalantari	1.0	all	Considering feedback by OGC

Appendix B: Bibliography

[1] 3DCityDB, : http://www.3dcitydb.org/3dcitydb/d3ddatabase/ (2017)

[2] 3DCityImporterExporter, : http://www.3dcitydb.org/3dcitydb/d3dimpexp/ (2017).

[3] 3DCityWFS, : http://www.3dcitydb.org/3dcitydb/dwfs/ (2017).

[4] 3DCityWebClient, : http://www.3dcitydb.org/3dcitydb/d3dwebclient/ (2017).

[5] 52°North, : http://52north.org/communities/geoprocessing/wps/index.html (2017).

[6] BIMServer,: http://bimserver.org/ (2017).

[7] GeoServer,: http://geoserver.org/ (2017).