

Investigating the Automation of the Building Permits Issuing process through 3D GeoBIM information - Final Deliverable

Francesca Noardo, Teng Wu, Ken Arroyo Ohori, Thomas Krijnen and Jantien Stoter

3D Geoinformation, Delft University of Technology, Delft, The Netherlands - (f.noardo, t.wu, k.ohori, t.krijnen, j.e.stoter)@tudelft.nl

July 3, 2020

Contents

1	Introduction	2
1.1	Submitted documents as final deliverable	3
2	Methodology	3
2.1	Case study	3
3	Results	9
3.1	The chosen regulations and their formalization	9
3.1.1	Dimension regulation	9
3.1.2	Parking regulation	9
3.2	Ground truth measurement	9
3.2.1	Maximum height	10
3.2.2	Building segmentation	10
3.2.3	Storeys height	12
3.2.4	Overhangs	12
3.3	Inspection of the IFC models provided by the stakeholders in the case studies	13
3.3.1	Grouping of building elements in storeys	14
3.3.2	Storage of information in the IFC models: split models	17
3.3.3	Storage of information in the IFC models: semantics accuracy	19
3.3.4	Overlapping elements in the BIM	20
3.3.5	Ceiling-slab-floor ensemble thickness in the facade	21
3.3.6	Use and modelling of Spaces and <i>IfcSpaces</i> in the BIM	22
3.3.7	Representation of parking places	26
3.4	The implemented demonstrator	29
3.4.1	Storey overlap calculation algorithm	30
3.4.2	Overhang distance checking algorithm	32
3.4.3	Tool performance	33
3.4.4	Using the tool	35
3.5	Guidelines	35
3.5.1	Guidelines to BIM modellers	35
3.5.2	Guidelines to geoinformation modellers	36

3.5.3	Guidelines to planners	37
4	From the detail to the GeoBIM: general topics arisen	37
4.1	Georeferencing the IFC models	37
4.1.1	Georeferencing introduction	38
4.1.2	3D Georeferencing parameters calculation	39
4.1.3	Georeferencing storage within the IFC model	39
4.1.4	Proposals of concrete solutions for BIM models georeferencing	40
4.2	IfcSpaceType enumerations extension	42
5	Discussion	47
6	Conclusions	48

1 Introduction

This document describes the outcomes of the project “Investigating the Automation of the Building Permits Issuing process through 3D GeoBIM information” (1st October 2019 – 1st July 2020), funded by the Municipality of Rotterdam (H. Tezerdi and R. Manbodh).

The aim of the project was the investigation of the topic of automation of building permit issuing by means of digital tools and 3D data (namely, Building Information Models — BIM — and 3D city models). In addition, the initial findings were implemented into a demonstrator tool able to use BIM and digital geoinformation to support the Municipality officers in the checking of regulations for building permits.

The topic of digitalization and automation of building permit process is widely promoted around the world and in Europe (e.g. Noardo et al. [2019]). The European Directive 2014/24/EU¹ strongly encourages the use of BIM for public projects. A European network (the EUnet4DBP²) was recently born to collaborate and coordinate the actions about the many issues involved in such a complex topic.

Many countries and municipalities realised and calculated how much savings (in terms of time and money) an automated (or assisted) tool could allow. Especially big cities like Rotterdam, with high number of asked permits and often complex situations could have advantage of such a tool, allowing the municipality officers to focus better on the cases of non-compliance and possible exemptions.

While developing the project, starting from the specific case study in Rotterdam (see Section 2), some specific observations and issues could be pointed out and investigated (Section 3.1), and some solutions were implemented into a demonstrator (Section 3.4). From such experience, some initial guidelines are proposed in Section 3.5, to support the production of suitable input data. A proposal to two more general encountered issues: the georeferencing of IFC and the need for extension of *IfcSpaceType* enumeration according to their functions are described in Section 4, before some discussion (Section 5) and conclusions (Section 6). In the discussion section (5), a table assessing the extent and quality reached by the results of this project is added, within which it is also proposed how to improve them in further steps.

¹<https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32014L0024>

²<https://3d.bk.tudelft.nl/projects/eunet4dbp/>

The premises of the project and the initial part of the study are extensively explained within the updated Deliverable 1 of this project and the paper [Noardo et al., 2020a].

1.1 Submitted documents as final deliverable

The final deliverable of the project is composed of the following files and documents:

- This document describing the developed work and the implemented demonstrator;
- The updated Deliverable 1, corrected and integrated with respect to the version delivered in December;
- The paper [Noardo et al., 2020a] submitted for review to the conference 3D GeoInfo 2020³ entitled “GeoBIM for digital building permit process: learning from a case study in Rotterdam”, where the general methodology of the project is described, together with the initial steps regarding the interpretation and formalization of regulations and the definition of the workflow for checking the building dimension regulation;
- The folder containing the tool with user interfaces (a.exe file), the results calculated by the tool with the provided BIMs and the BIM themselves which were used (as delivered and georeferenced by us).

2 Methodology

The approach that was used within the project was almost completely bottom-up. Following the framework of a general reference workflow⁴, it was chosen to start from a specific case study in Rotterdam (Section 2.1) and two regulations (namely, the building dimensions and the planning of parking places) to point out the necessary steps in detail and encountered issues to be investigated during the project.

The regulations and the involved data were analysed (Sections 3.1, 3.2 and 3.3) as a starting point to develop the supporting tool for regulation checks (Section 3.4).

Finally, some flaws in the current best practices or standards were detected during the investigation. In particular, some notes and initial proposals for the georeferencing of IFC models and extension of *IfcSpaceType* enumerations are described in Sections 4.1 and 4.2) respectively.

2.1 Case study

In consultation with the Rotterdam team, the zone “Centrum 3” of the *Waterstad bestemmingsplan* (destination plan) in the centre of Rotterdam (Maritiem district, Figure 1) was selected. The existing zoning plan was drawn up in 2015 in the spirit of the upcoming environmental law, and it was already object of a previous experiment regarding a pilot for 3D⁵, foreseeing the implementation of the environmental law, by the city of Rotterdam. Figure 1 shows the area in the *Ruimtelijkeplannen* webGIS⁶, mapping the zoning regulations for the Netherlands, showing the *bestemmingsplan* (destination zoning plan).

³<https://www.ucl.ac.uk/3dgeoinfo/>

⁴<https://3d.bk.tudelft.nl/projects/eurosdrgbobim/WorkflowDescription.pdf>

⁵<https://arcg.is/1Cu0Te>

⁶<https://www.ruimtelijkeplannen.nl/viewer/viewer>

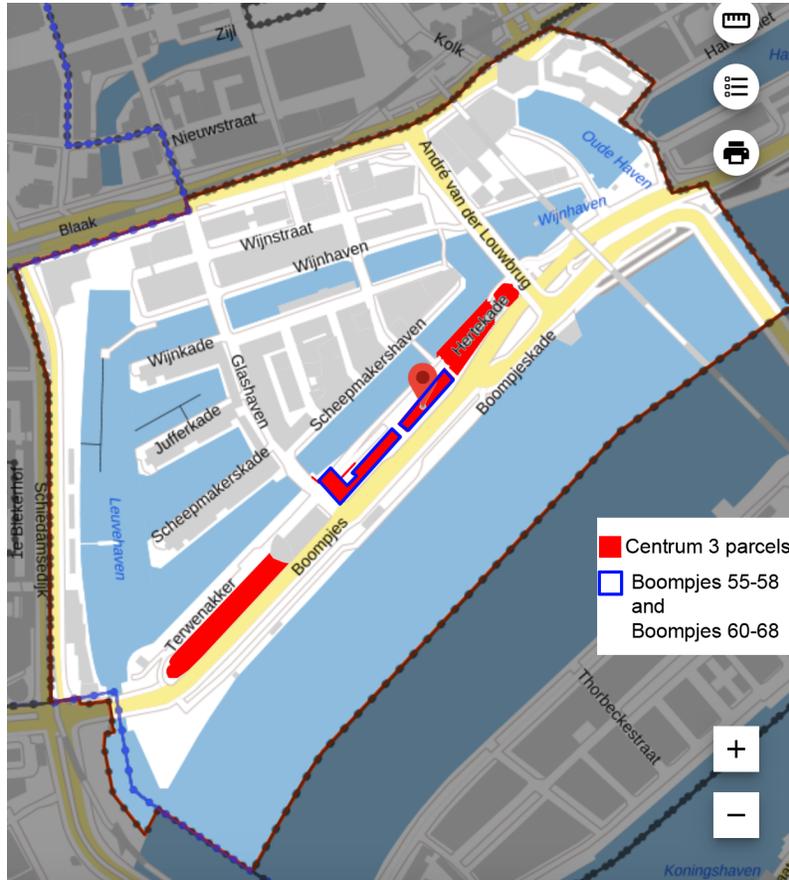


Figure 1: Map of the Maritiem district. The “Centrum 3” area is in red in the image, with addresses Boompjes 55–58 and 60–60 outlined in blue.

The dimension regulation⁷ and the parking regulation⁸ are considered for this project.

Within the area, two buildings were recently designed, specific case studies for this project. The two respective BIMs were kindly provided in IFC (v.2x3) for the tests:

PeakTower Boompjes 60-68 – Project organization: Peak Development – Architect: Team V Architectuur

TerraceTower Boompjes – Project organization: Provast – Architect: OZ Architects

The *Peak Tower* is composed by structural, architectural and facades model (Figure 2). They are correctly registered together, by means of the same reference point and orientation. They are not georeferenced, though: the *IfcCartesianPoint* referenced by *IfcSite* according to a *LoGeoRef 30* (see Section 4.1), reports coordinates (0, 0, 0), whilst the *RefLatitude* and *RefLongitude* attributes in the *IfcSite* (according to a *LoGeoRef 20*) would locate the model in a general location in Amsterdam. In Figure 3 it is

⁷https://www.ruimtelijkeplannen.nl/documents/NL.IMRO.0599.BP1054Waterstad-va01/r_NL.IMRO.0599.BP1054Waterstad-va01.html#_5_Centrum-3

⁸http://decentrale.regelgeving.overheid.nl/cvdr/xhtmloutput/Historie/Rotterdam/486392/486392_1.html and https://www.ruimtelijkeplannen.nl/documents/NL.IMRO.0599.BP1054Waterstad-va01/b_NL.IMRO.0599.BP1054Waterstad-va01_rb2.pdf

possible to see some external dimensions measured in the facades model, representing, in this case the external part of the building, which is the one useful to check the building permit dimensions.

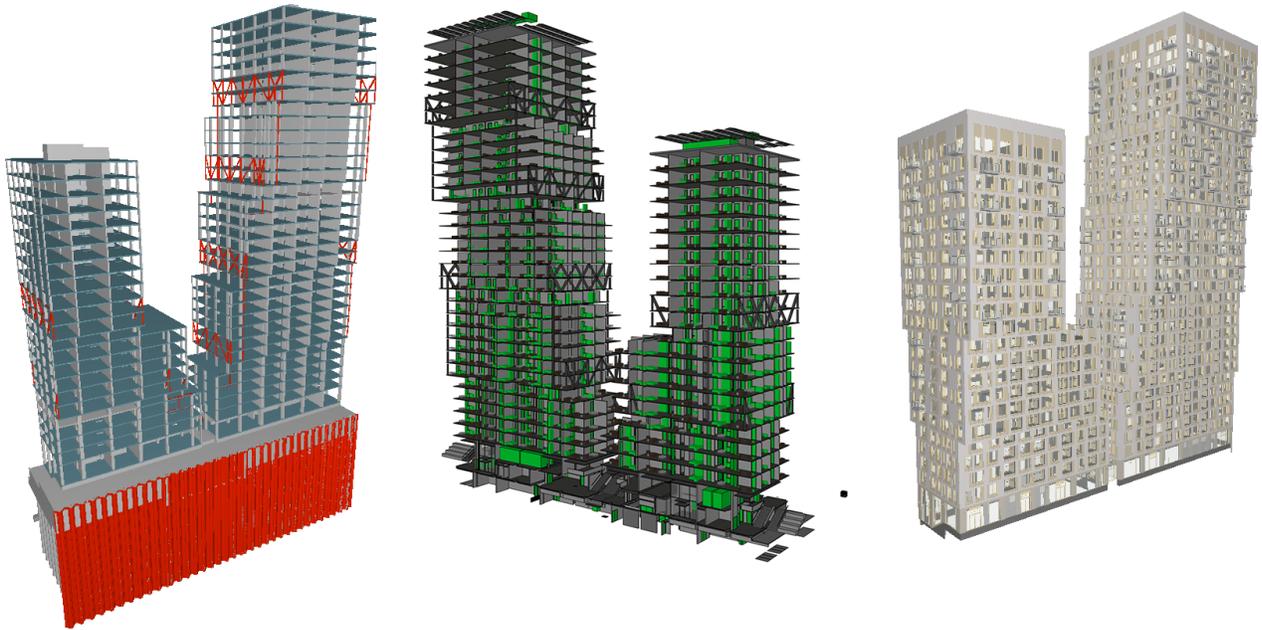


Figure 2: The three IFC models composing the BIM, from left to right, the structural, the architectural and the facades one.

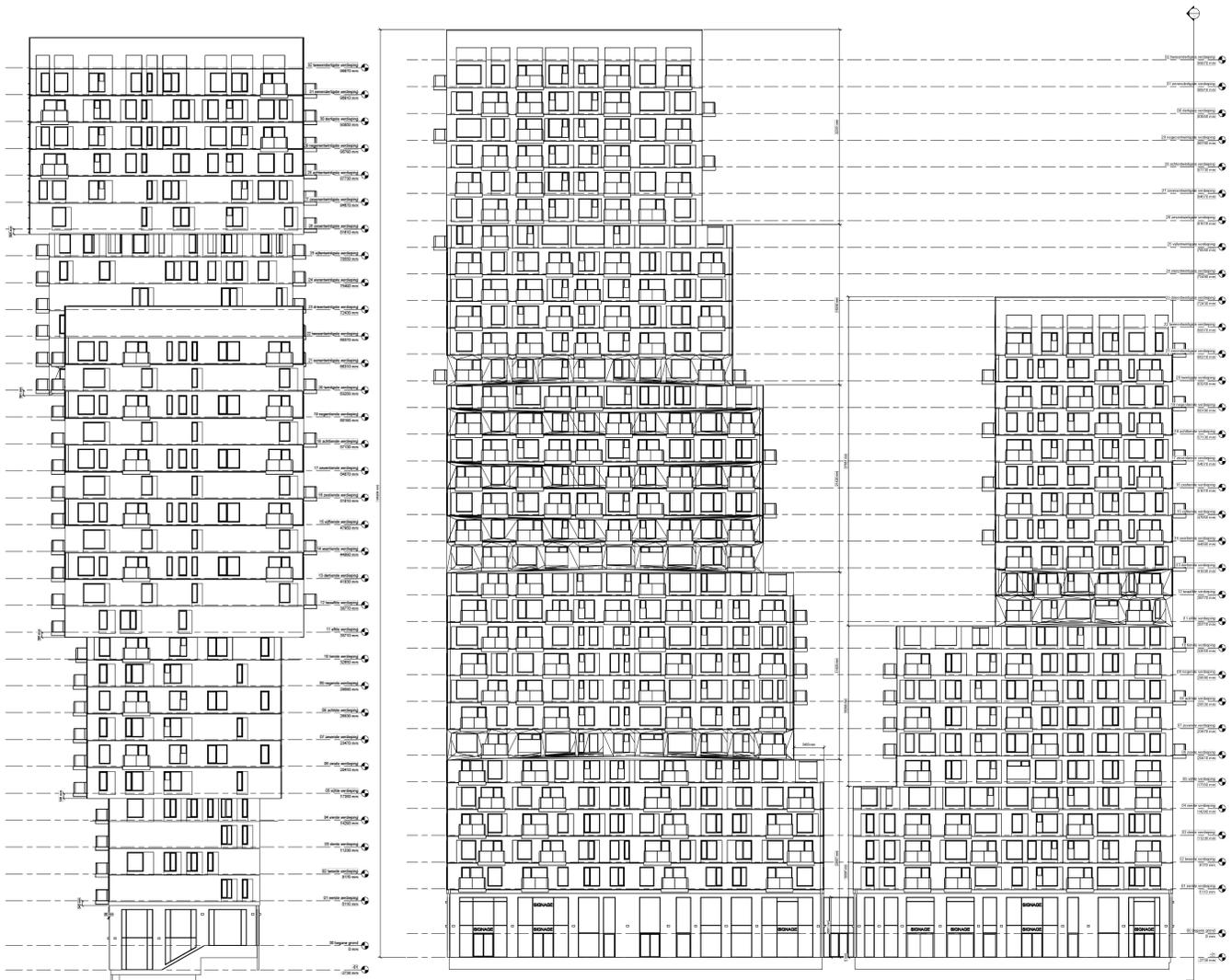


Figure 3: One of the side facades of the building, with dimensions. And building facade towards Het-erkade, with the dimensions of the segmented building parts.

The other BIM, representing the *Terraced tower*, by structural and architectural models, plus one representing the context and elements probably belonging to the work site. They are correctly registered together too; the direction is correctly stored as *TrueNorth* attribute of the *IfcGeometricRepresentation-Context* (according to *LoGeoRef 40*). In this case the *RefLatitude* and *RefLongitude* attributes in the *IfcSite* (*LoGeoRef 20*) would locate the model in the United States, but the *IfcCartesianPoint* referenced by *IfcSite* reports the right coordinates of the Rotterdam location in the Dutch projected Coordinate Reference System (93191.0497637025, 436743.261) (valid for *LoGeoRef 30*), expressed in millimeters.

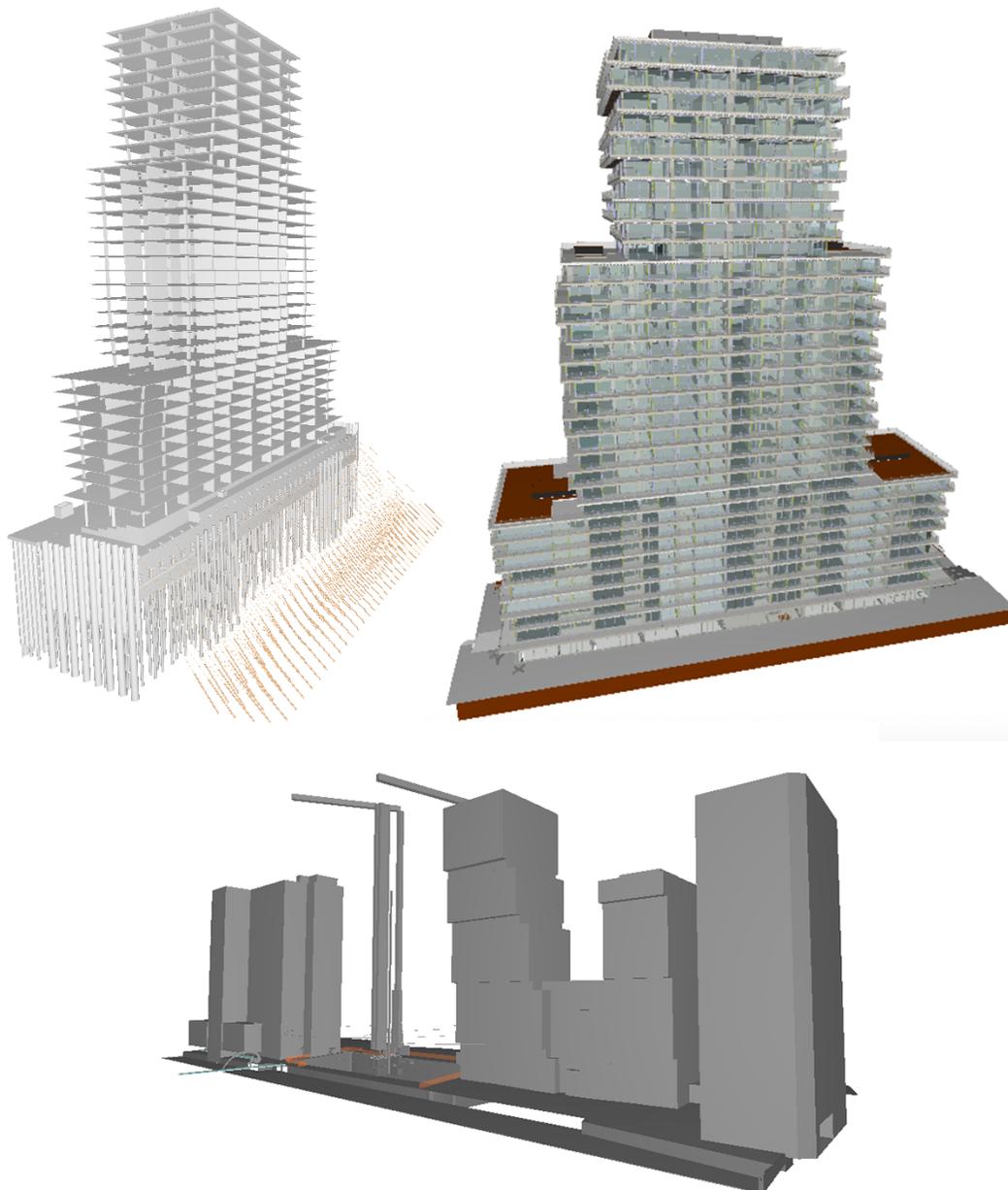


Figure 4: The three IFC models composing the BIM, clockwise, the structural, the architectural and the context one.

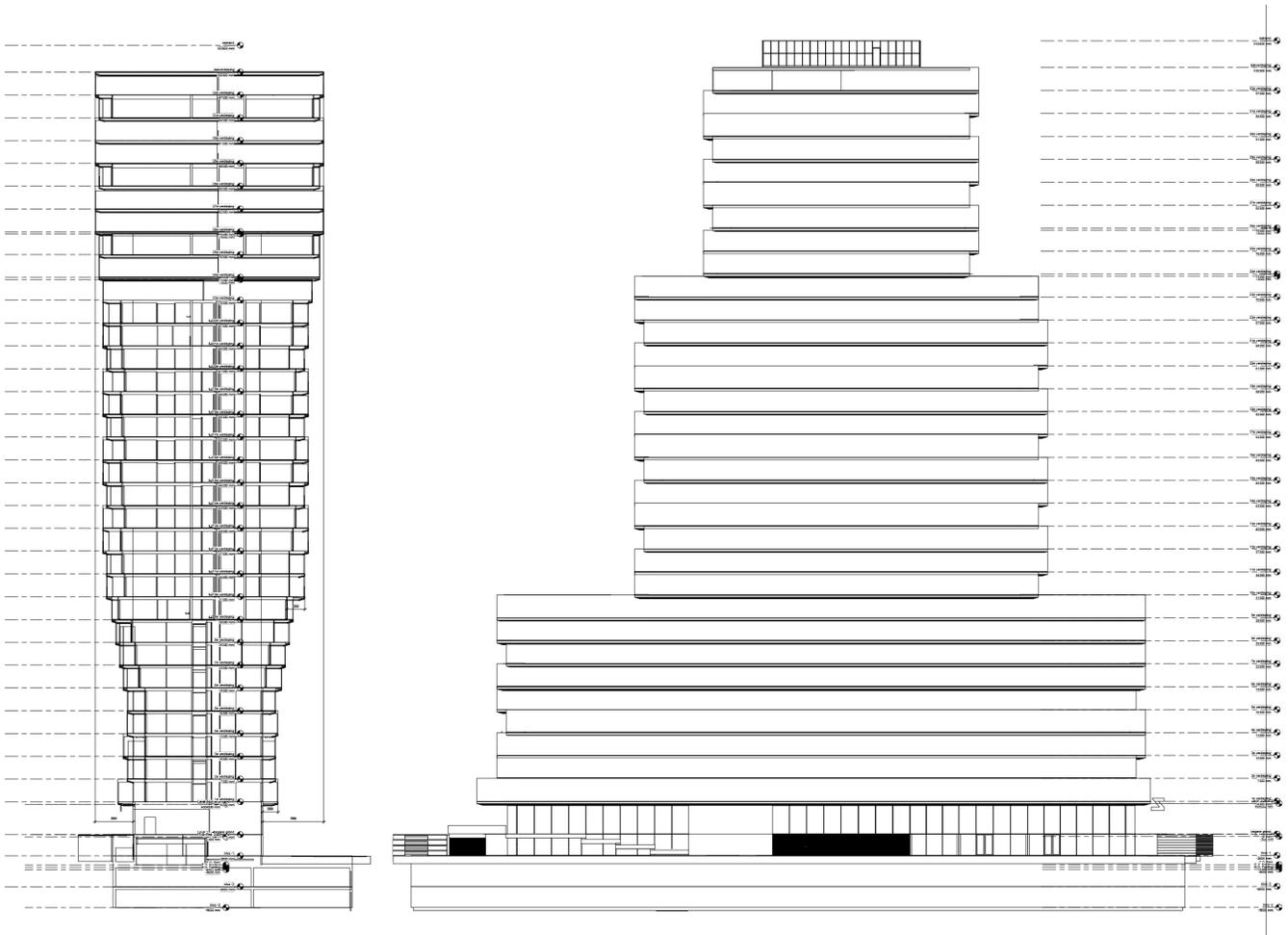


Figure 5: One of the side facades of the building, with dimensions. And building facade towards Het-erkade, with the dimensions of the segmented building parts.

The Rotterdam CityGML model in addition was used as 3D city model. It contains the CityGML data of tree crown, tree trunk, charging stations street lamps and Basisregistratie Grootchalige Topografie (BGT). The source date of the CityGML data is 10-01-2020. The CRS of the CityGML is EPSG:28992-Amersfoort/RD New.

The geometry type of the tree crown, tree trunk, charging stations street lamps is points containing x,y coordinates and height. The BGT contains the footprints of buildings, generic city object, land use, plant cover, road network and water body. And the geometry type of BGT is 2.5D polygon. There is one attribute stored the height information of the building polygons, but it is empty in the dataset.

In addition, Actueel Hoogtebestand Nederland (AHN3) point cloud is used to calculate the ground height of the BIM.

3 Results

3.1 The chosen regulations and their formalization

The first step towards the development of a tool automating the building permit process is the formalization of regulations to be checked, to make them machine-readable. This process was quite challenging, since their text, at present in natural language, is still open to many interpretations. First step was therefore the disambiguation of the rule (especially the dimensions rule) with the help of municipality officers usually dealing with the checks.

3.1.1 Dimension regulation

An extensive explanation of the dimension regulation⁹ and its translation in a more formal language is explained in Deliverable 1 of this project [Noardo et al., 2020b] and by Noardo et al. [2020a].

3.1.2 Parking regulation

The interpretation of regulation regarding the provision of parking places¹⁰ was easier, and an excel sheet supporting the municipality officer in the calculation was already available. Although some complexity to this is given by its connection to further rules, which are not written, but rely on the practice knowledge of the municipality experts, it is potentially easier to understand the stated requirements unambiguously. An initial formalization of the regulation was proposed in the Deliverable 1 of the project [Noardo et al., 2020b].

The procedure can be summarized according to the following steps:

1. Define the building units and their function;
2. Calculate the area of each apartment;
3. Calculate the number of the min parking units (“*MinPP*”) by means of the formulas proposed in Deliverable 1 [Noardo et al., 2020b];
4. Checking if the designed parking units meet the regulation.

3.2 Ground truth measurement

In order to set the ground truth for the considered checks, the BIMs were measured in Revit and other BIM viewer (e.g. the Solibri Model Viewer).

While it was possible for the dimensions regulation, as explained in the Sections 3.2.1 – 3.2.4, for the parking regulation the use of BIM can be tricky. Even using a manual procedure, the representation is not set to support the checks easily. The Section 3.3.7 gives more details about the encountered issues. Some initial guidelines are proposed in Section 3.5.1 to support automation for this task.

⁹https://www.ruimtelijkeplannen.nl/documents/NL.IMRO.0599.BP1054Waterstad-va01/r_NL.IMRO.0599.BP1054Waterstad-va01.html#_5_Centrum-3

¹⁰http://decentrale.regelgeving.overheid.nl/cvdr/xhtmloutput/Historie/Rotterdam/486392/486392_1.html and https://www.ruimtelijkeplannen.nl/documents/NL.IMRO.0599.BP1054Waterstad-va01/b_NL.IMRO.0599.BP1054Waterstad-va01_rb2.pdf

3.2.1 Maximum height

As measured in *Revit*, the maximum height of the **Peak tower** model, considering the entrance door towards Boompjes is **103.47 m** and 106.20 m starting from the entrance towards Heterkade. Considering the BIM design, the last storey is completely over the limit.

We can see from the BIM that the last 3 meters in facade (5.97 m if we consider the element rising the highest) are occupied by installations and technical rooms. For this reason, an exemption could be granted and the design approved anyway, as foreseen by the Article 19.2¹¹. The 0 is set at the entrance towards Boompjes, which is the highest one.

In the **Terrace tower** BIM, the maximum height from the entrance door towards Boompjes, which is the higher one, is **100.46 m** to the roof top. According to the possible exemptions described in Article 19.2, it should be permissible. Alternatively, we can measure until the top of the highest elements in the building, including installations, lifts and solar panels. It is 103.82 m, according to the possible exemptions described in Article 19.2, could be permissible anyway. This non-compliance should be notified by a final perfect tool, together with the notification that in those cases a derogation could be granted.

One more check should be necessary, considering the road elevation adjacent to such entrance, instead of the entrance itself.

3.2.2 Building segmentation

The considered regulation foresees the division of the building in two parts: a base and a top part occupying half of the footprint area of the previous one and allowed to overhang towards two of the enclosing streets.

Therefore, first need is to detect those two parts. Later on it will be possible to check if they respect the dimensions established by the regulation.

The IFC models we got for the experiment, as usual, are divided in building storeys. The entity *IfcBuildingStorey* groups the elements belonging to the same floor, approximately from slab to slab. The software exporting the IFC file usually consider all the elements at the same level as belonging to the same storey, therefore the same floor of two different towers (as it happens in this case with the *Peak tower* BIM) are grouped within the same *IfcBuildingStorey*. It is necessary to consider this when implementing automatic solutions based on *IfcBuildingStorey* grouping. The IFC model does not directly prescribes a way to group parts of the building differently.

The Boompjes BIM can be segmented by a human like in Figure 6, with the lower building part (P1, orange in the picture) measuring 6.9 m and the second one (P2, pink in the picture) 19.597 m. Another criteria to detect the two parts of the building (the base and the top part) in this case could also be the separation of the building where it is composed by only one contiguous part or both of the two towers. However, since this represents already a further interpretation of the regulation, we didn't use it.

¹¹https://www.ruimtelijkeplannen.nl/documents/NL.IMR0.0599.BP1054Waterstad-va01/r_NL.IMR0.0599.BP1054Waterstad-va01.html#_19_Algemeneafwijkingregels

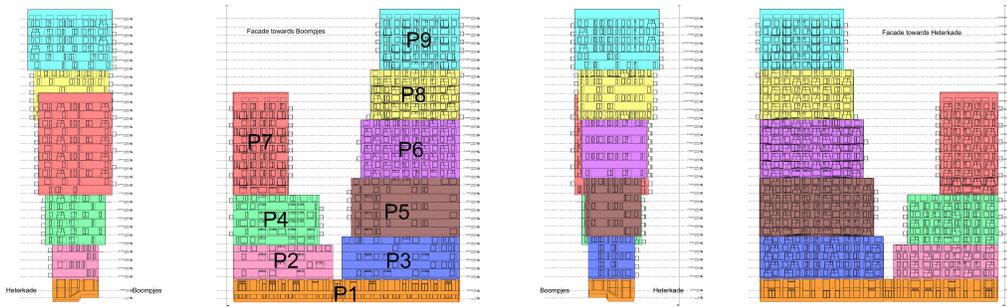


Figure 6: Facades of the Boompjes BIM segmented intuitively by human judgement.

The Terrace tower BIM Figure 7 can be segmented in three parts. The height of the first one is 31.3 m, apparently outside the 17 m limit.

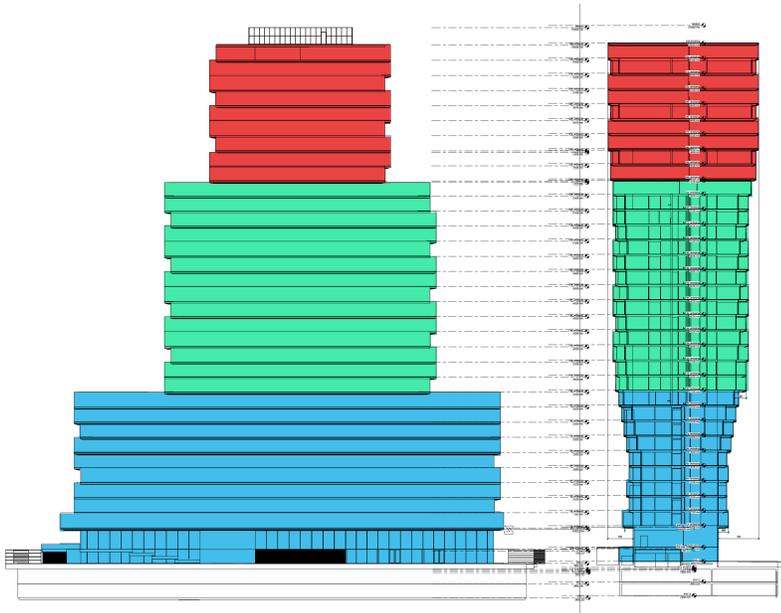


Figure 7: Facades of the Terrace tower BIM segmented as intuitive by a human eye.

Looking at Figure 6 and Figure 7 it is possible to notice that the facade is not completely plain, but it is articulated by the several facade elements, windows, and slight differences between the floors. From the measurements in Figure 8, it is possible to notice how a human eye considers the 1.2 m protruding parts as still part of the same block. Instead, the bigger protrusions, such as in Figure 8 (a), are detected as actual discontinuities. Other kinds of considerations with respect to the proportions between the total dimensions and the protruding part could be further explored.

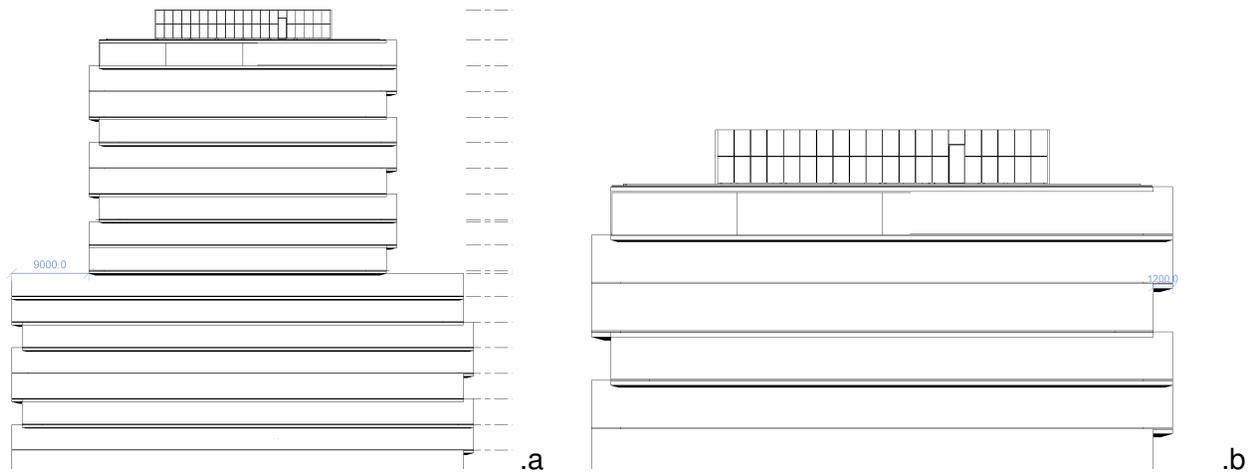


Figure 8: From the measurements in the pictures, we can notice how a human eye considers the 1.2 m protruding parts (b) as still part of the same block. Instead, the bigger protrusions (a) are detected as actual discontinuities.

3.2.3 Storeys height

An important dimension for the automation of the checks could be the thickness of the floors, and more in general, the distance between the storey height, usually stored within the *Elevation* attribute of the *IfcBuildingStorey* and the lower element of the storey jutting out in the facade. This could allow the use of *IfcBuildingStoreys* as reference or starting point to implement automatic procedures and infer at least approximate dimensions. It is challenging to find a fair approximation for all buildings, since it depends on a huge number of factors, such as the chosen construction and structural system, the added isolation, the possible location of installation ducts or halls, specific facade systems and so on. However, a very inaccurate approximation, according to the BIMs delivered for the case study could be around 50 cm (Figure 9). One more measurement, from the other facade in the *Terrace Tower* BIM, showed 31 cm, since the elements considered by the other side were less visible. This is an example of both the variability of this parameter and the difficulty to reach a very high accuracy for a human when inspecting the BIM.

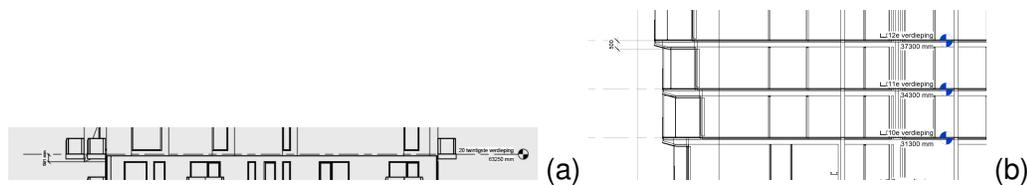


Figure 9: The two pictures show the dimensions of the slabs (from the *Elevation* value of *IfcBuildingStorey* to the exterior ceiling) in the *Peak tower* BIM (a) and in the *Terrace tower* BIM (b).

3.2.4 Overhangs

The regulation imposes that the overhang of the top part of the building with respect to the building base must not exceed 5 m in the direction of the Boompjes street, and 10 m in the direction of Heterkade street. Also these dimensions were checked in *Revit* (Figure 10).

In the *Peak tower* model, the overhang towards Boompjes is 5.08 m and the overhang towards Heterkade is 9.08 m, therefore, the regulation should result as approved.

In the *Terrace tower*, the overhang towards Boompjes is 5.00 m and the overhang towards Heterkade is 7.98 m, therefore, the regulation should result as approved.

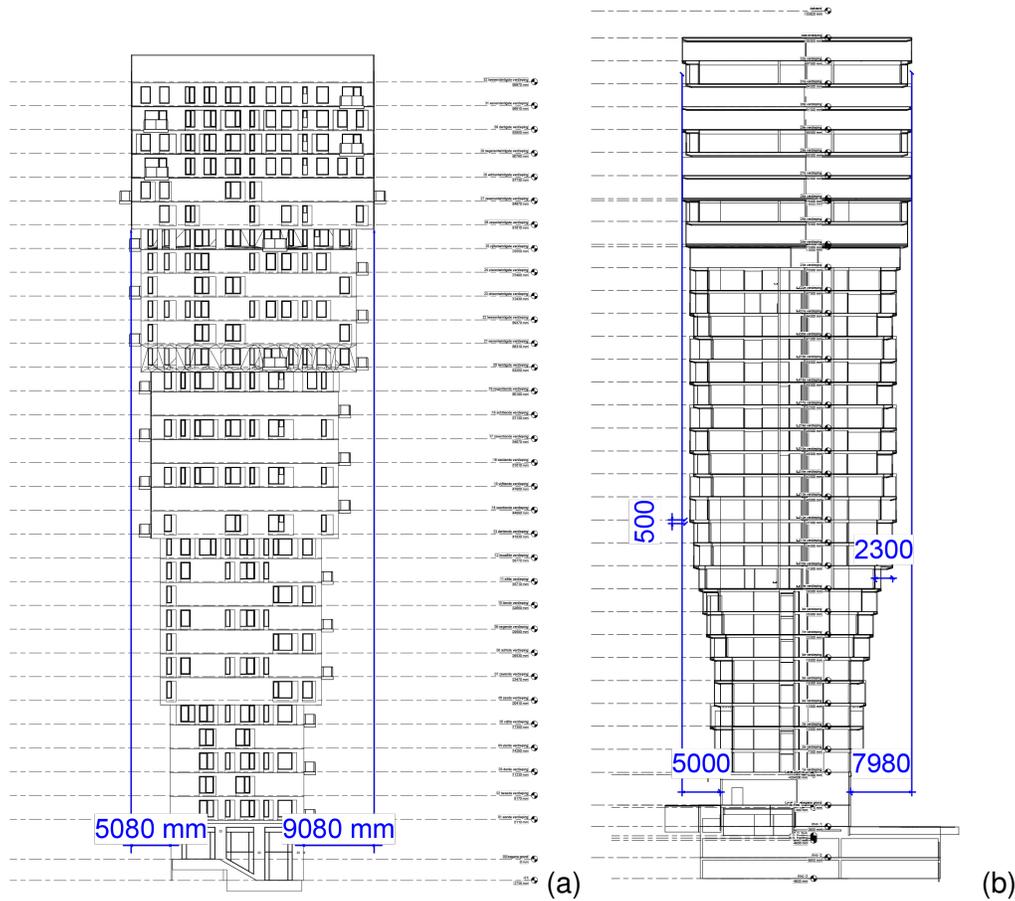


Figure 10: The two pictures show the dimensions of the overhangs of the two buildings Peak Tower (a) and Terrace tower (b) as measured in Revit.

3.3 Inspection of the IFC models provided by the stakeholders in the case studies

The practice of building design and BIM follow criteria and best practices defined both within the design discipline itself and by national guidelines, sometimes. At the same time, the buildingSMART IFC¹² data model is defined as reference standard for openBIM, which is therefore the desirable interoperable format to be used by tools. However, on the one hand designers can follow own specific unofficial rules when modelling, on the other hand, software can implement IFC in different ways, so that we cannot straightforwardly expect the same product (i.e. complying to exactly the same part of the IFC structure, used in the same way) each time and in each case.

The models were manually inspected, with several aims: outline the characteristics of the BIMs; point

¹²<https://technical.buildingsmart.org/standards/ifc/>

out the related possible issues for the use of them within automatic tools; potentially guide the implementation.

Again, mainly *Autodesk Revit* and the *Solibri model Viewer* were used to inspect the IFC models manually, in order to point out their features, both as consequence of the modelling practice and as the use and implementation of the IFC standard.

In particular, the inspected issues in the BIMs are: the containment relationships of elements and their related *IfcBuildingStorey* (Section 3.3.1); the representation of the elements in the BIM, considering the combination of several models with specific scope (structural, architectural, installations) (Section 3.3.2); the consistency of semantics (Section 3.3.3); the possible overlapping of elements (Section 3.3.4); the modelling of the ceiling-slab-floor ensemble thickness (Section 3.3.5); the use of spaces (Section 3.3.6); the representation of parking places in the BIM (Section 3.3.7).

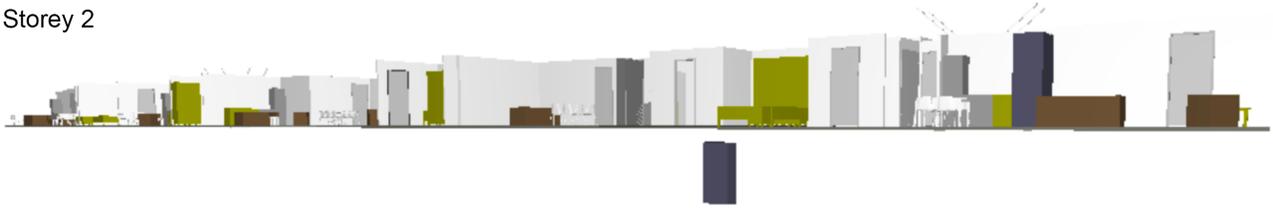
3.3.1 Grouping of building elements in storeys

The most intuitive thing to imagine when figuring out how to cut a building vertically (as it should be useful to detect the two parts of the building to be considered as the base and the top part respectively) is to rely on storey subdivision, which is a spatial structure whose representation is foreseen by the IFC data model. However, when checking the grouping consistency in the models, it is apparent how the grouping of entities in storeys is not perfect (as previously experienced also in other IFC models).

The inaccuracies can be grouped in three main cases:

1. One storey also includes elements belonging to different storeys (e.g. Figure 11, Figure 12 and Figure 13);
2. They contain solely a handful of isolated elements that should be included in an other storey at similar elevation (e.g. Figure 14).
3. The storey can contain elements that extend for two floors (e.g. Figure 15 and Figure 16). The IFC standard suggests to break up walls at storey boundaries, but for other elements it is allowed that they span multiple levels, the *IfcRelReferencedInSpatialStructure* is introduced to signify this relationship. However, in those cases, it is not a modelling error, although the measurements coming from such grouping (e.g. the bounding box) would not be accurate, anyway. The grouping in storeys (e.g. the bounding boxes containing each storey) is therefore not reliable in itself for making accurate calculations.

Storey 2



Storey 3



Storey 5

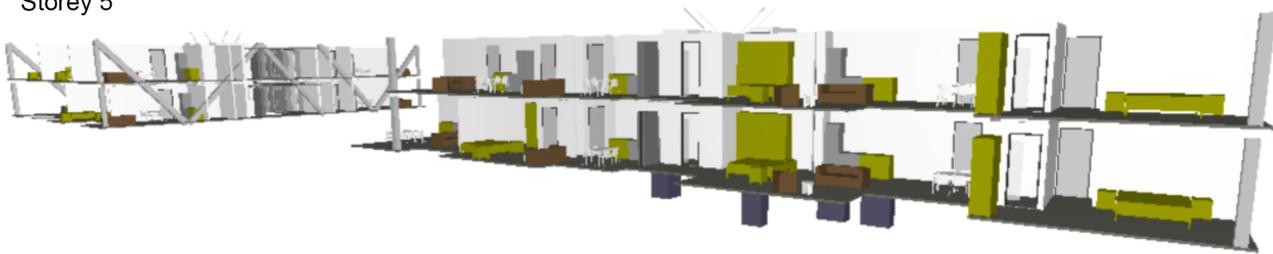


Figure 11: In Boompjes BIM, examples of elements stored as part of the following storey although belonging to the lower one. Moreover, in storey 5, two building levels are included.

- IPI nivo -3
- IPI nivo -2
- IPI nivo -1
- **IPI begane grond**
- IPI 1e verdieping
- IPI 2e verdieping
- IPI 3e verdieping
- IPI 4e verdieping
- IPI 5e verdieping
- IPI 6e verdieping

INFO

IPI begane grond

Identification	Location	Quantities	Relations	Classification
Property				Value
Model				BIM
Discipline				Architectural
Name				begane grond
Description				
Number				0
Application				Autodesk Revit 2018 (ENU)
GUID				0ac_1p1WD3OgU51Tbq36nYI
Model Categories				

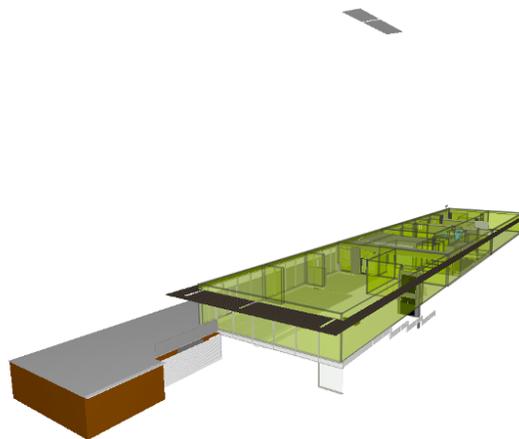


Figure 12: In Terrace tower BIM, example of element wrongly assigned to the ground floor storey.

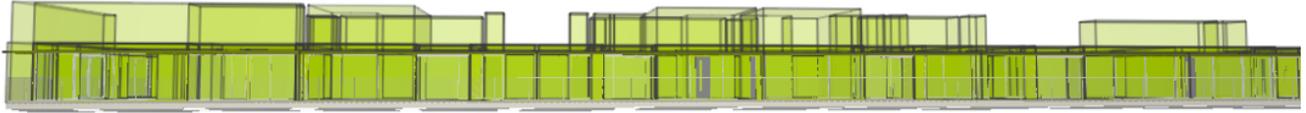


Figure 13: In *Terrace tower* BIM, example of spaces extending for two storeys.

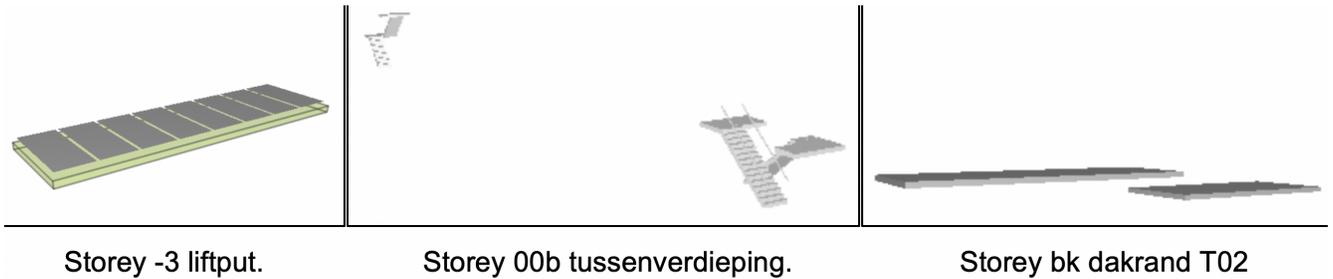


Figure 14: In *Peak tower* BIM, three examples of storeys grouping single or few elements in the architectural model.

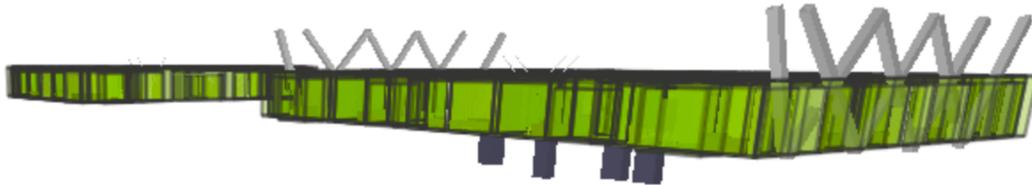


Figure 15: In *Peak tower* BIM, In this picture of storey 13 it is possible to see how some elements necessarily extend for two floors, although not being a modelling error.

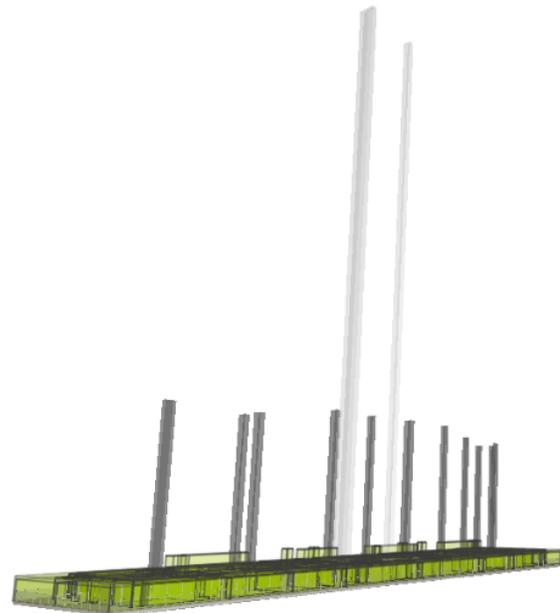
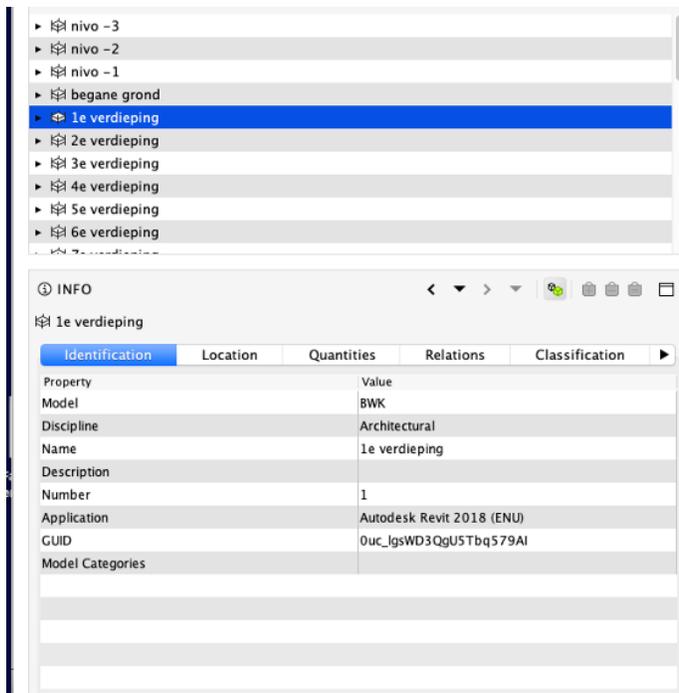


Figure 16: In *Terrace tower* BIM, many storeys contain also the representation of ducts (ventilation ducts and shafts) running for several floors.

In some of these cases, the grouping in storeys of the IFC could be fixed, for example by excluding from the grouping the elements which do not fit within a buffer with respect to some reference height, which could be given for example by the *Elevation* attribute of the *IfcBuildingStorey* or by a statistical parameter calculated among the barycentres of the *IfcBuildingElements* grouped in the storey. It could be a solution for the cases 1 and 3 listed above. For fixing the case n.2 (isolated elements grouped as one storey) some other processing would be necessary, such as putting a threshold to the number of entities included or related to the dimension of the bounding box with respect to the one of the other storeys.

The implementation of such a tool could be one of the future developments.

Additional note, to support the automation of the use of the BIM, both to check the building regulations and also to use its element for the 3D city model update, through integration, it would be necessary to group consistently and carefully the elements belonging to the building itself and the ones which are part of the site around it, by grouping them accordingly either in *IfcBuilding* (and sub-grouped in *IfcBuildingStoreys*) or *IfcSite*.

3.3.2 Storage of information in the IFC models: split models

It is important to consider how the information is stored within the BIM. First, one BIM is usually composed by different models, stored in different IFC files.

To use a consistent model, the recommendation is to consider all the IFC models belonging to the BIM for the calculations. For example, in Figure 17 it is possible to see how it is not even possible to rely on

the maximum extent of slabs, or of one only model composing the BIM. In this case, for example, to the protrusion of the slab overhanging the lower part of the building is measured as 1.94 m, if considering only the architectural model. However, additional 0.47 m are occupied by the space of the facade, here stored in a different model, and just represented here by means of spaces (which is not an official best practice at the moment, though).

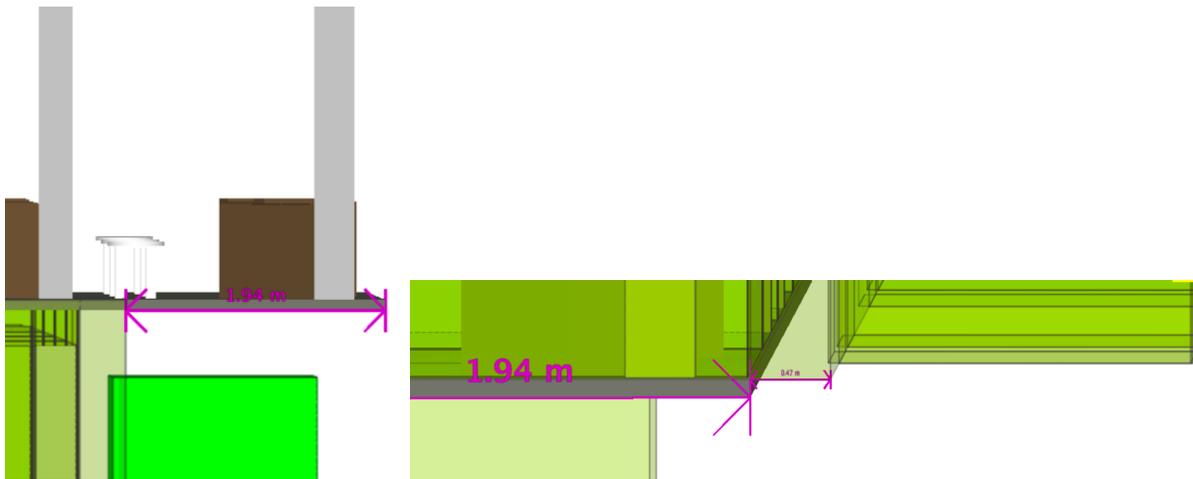


Figure 17: In *Peak tower* BIM, Dimensions of the slab protruding from the lower building part.

Another reason to consider all the models together is that elements that can be important for measurements or other kinds of computations (such as the calculation of paths) can be contained in different models. The building is finally shaped by their whole. For example, structural walls are missing in the architectural model as well as the structural part of the slabs, e.g. Figure 18 and Figure 19.

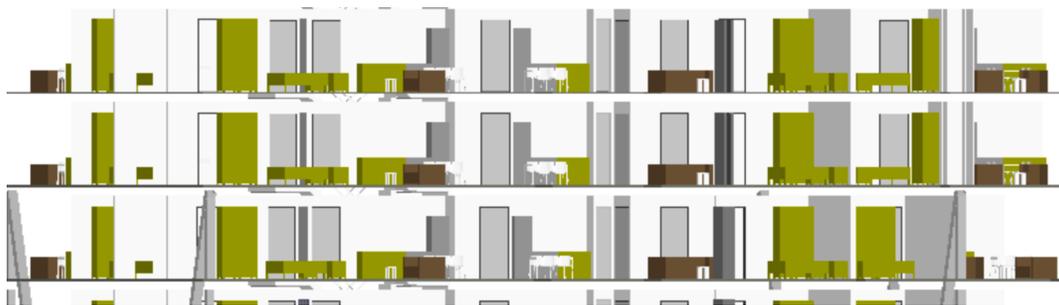


Figure 18: In *Peak tower* BIM, side view of the architectural model, from where it is possible to see that there is no connection among floors, since the structural slabs are missing.

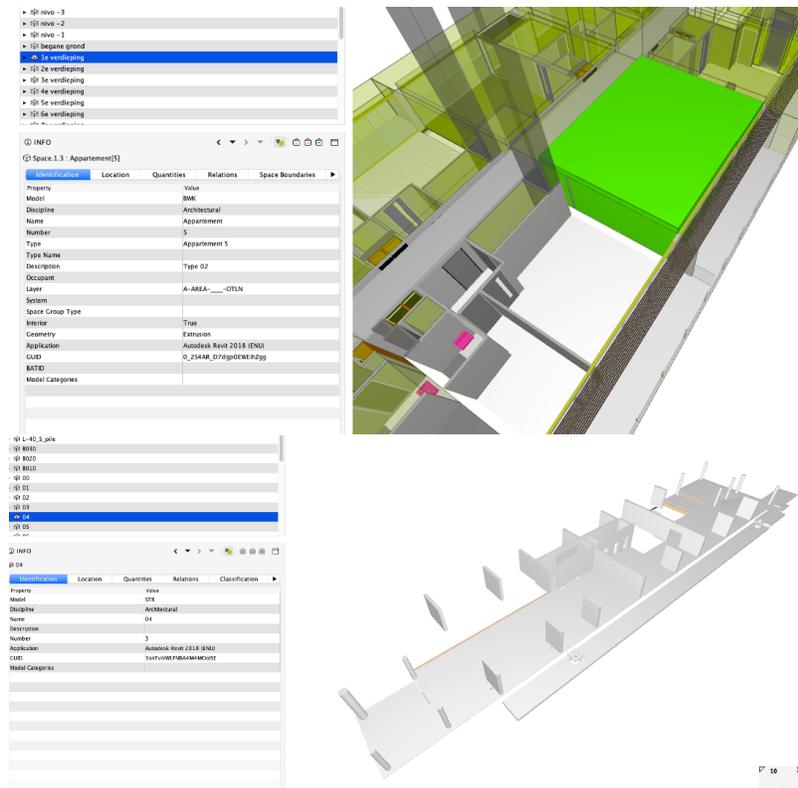


Figure 19: In *Terrace tower*, example of apartments where the dividing (structural) walls are absent in the architectural model, since included within the structural one (see the example of a storey in of the structural IFC model in the second picture).

3.3.3 Storage of information in the IFC models: semantics accuracy

The employed semantics is the one defined by the IFC standard. However, it is possible that such semantics could be wrongly assigned, either by the user during the design (which could be due both to human error and to the lack of constraints in the IFC model) or as a result of implementation inaccuracy in software or mapping tables.

First of all, human care by the designer is necessary, but probably, a tool allowing the check and validity of assigned semantics following clearer constraints would be useful.

This issue is strongly connected with the distribution of information in several models (Section 3.3.2) because both have to be considered when developing tools automatically handling them.

For example, for some application, only walls and slabs could be sufficient, and their extraction from the whole BIM could allow the composition of a lighter but complete model.

This issue is for sure complex and would deserve a more extensive investigation. However, what is possible to state by the plain inspection of these models is that it is not wise to blindly trust the semantics as stored in the IFC model (just a small example in Figure 20, but many cases could be listed). For this reason, unfortunately, the use of specific entities as reference for the measurements, at present, would not work.

For example, when considering if excluding some of the IFC entities for the calculation of dimension, it is

possible to notice that the elements to be counted in the building dimensions could be various (including installations, chimneys, or *IfcBuildingElementProxy*, which is a general IFC entity to represent whatever is not alternatively defined). In this case, this fuzziness makes the elimination of any entity risky (with the only possible exception of *IfcOpenings* and *IfcFurniture*).

Another example is the way the balconies are represented: in the *Peak tower* BIM, the slabs of the balconies are represented by means of *IfcWalls* (of course inconsistent representation), with an *IfcRailing* in its correct place.

In other cases, some structural elements are labelled as beams but probably have a different function, since their main extent is vertical.

A combination of guidelines and tools to fix the semantics after the export to IFC would be necessary as a next step.

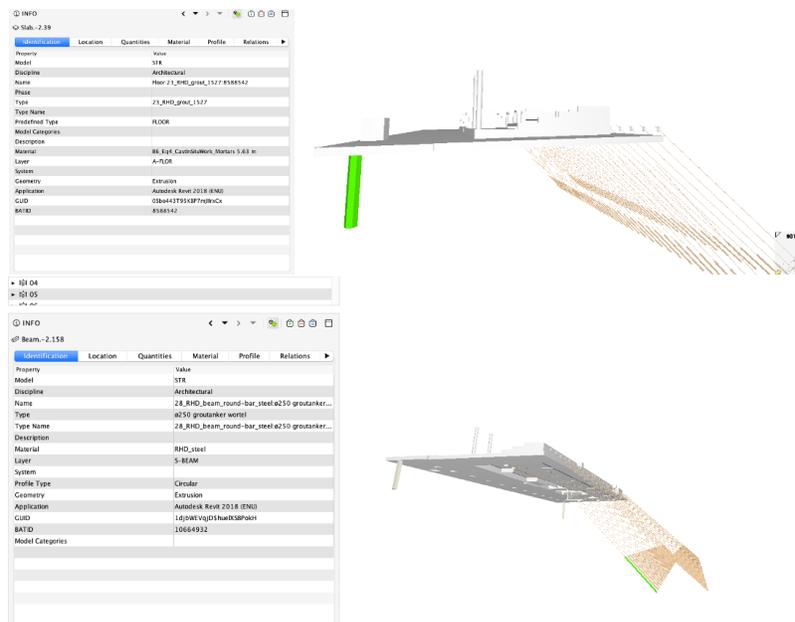


Figure 20: In *Peak tower* BIM, some examples of elements overlap.

3.3.4 Overlapping elements in the BIM

Moreover, it is quite frequent to find elements that intersect each other (walls and beams, walls and slabs, walls and doors, as few examples) (e.g. Figure 21 and Figure 22). This can be due not only to inaccuracy in the modelling phase, but also to the implementation of export algorithms in software. They represent an issue preventing the possibility of fully trusting the geometry present in the IFC model.

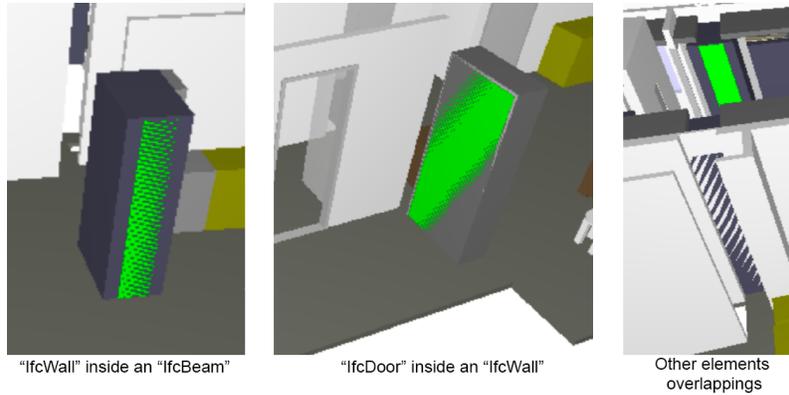


Figure 21: In *Peak tower* BIM, some examples of elements overlap. The bright green is the selected element, that is represented as superimposed to another one, so that two faces corresponds (the stripes and blurry colours are shown in that case)

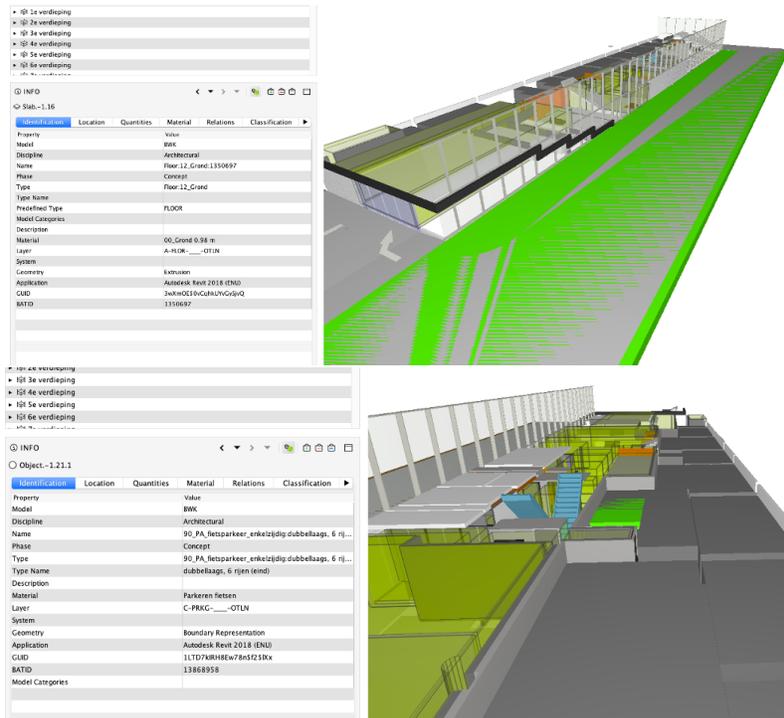


Figure 22: In *Terrace tower* BIM, some examples of elements overlaps. In the first case two slabs overlap. In the second case, many elements, modelling the bikes parkings and represented by means of *IfcBuildingElementProxy* overlap. As in the previous Figure, the bright green is the selected element, that is represented as superimposed to another one, so that two faces corresponds (the stripes and blurry colours are shown in that case)

3.3.5 Ceiling-slab-floor ensemble thickness in the facade

One more issue deserving a check regards the height of the lower surface of protruding parts of the building, seen as a ceiling from outside the building, as already pointed out in Section 3.2.2.

It is hard to assume that the distance between this and the floor height as stored in the *Elevation* attribute of the *IfcBuildingStorey* is easily calculated.

For example, considering the *Peak tower* BIM, 26th floor, the elevation of the *IfcSlab* is 81.61 m, the lower surface adjoining from the building (in green in Figure 23) is 81.02 m, that means that the total thickness is 0.59 m. The same slab global bottom elevation is instead 81.53 m, therefore the slab itself is designed as only 8 cm thick. Many other design choices can influence these dimensions (e.g. kind of construction and structural system, kind and amount of isolating materials, false ceilings, installation ducts or halls, specific facade systems etc.).

In the *Terrace tower* BIM such thickness is 0.50 m (see Figure 9).

Therefore, we can't just assume some dimensions but it is safer and easier to measure such height from the ground reference height, rather than rely on assumptions. However, a very rough approximation between 0.6 and 0.5 meters below the *Elevation* attribute value of the upper *IfcBuildingStorey* (without considering the terrain height in the site either) could provisionally work.



Figure 23: In *Peak tower* BIM, External lower part of a slab.

3.3.6 Use and modelling of Spaces and *IfcSpaces* in the BIM

A relevant spatial structure defined in the IFC data model, that can be very useful in a number of occasions is the *IfcSpace*. Both the *Peak* and the *Terrace tower* models present an interesting modelling of spaces, that can be really helpful in the checking of regulations, both for the potential extraction of the building envelope geometry and for the check of parking regulation.

In this specific cases, a redundant but comprehensive hierarchical modelling of spaces is provided, representing the whole storey (Figure 24 and Figure 25), the single apartments (Figure 26), the single rooms (Figure 27). Moreover, the same spaces are represented with different volumes (considered with ceiling or until the higher slab, balconies and so on).

Space1 One space per floor includes the whole storey area (Figure 24), excluding only the boundary slabs and walls with the exterior of the building (Figure 24);

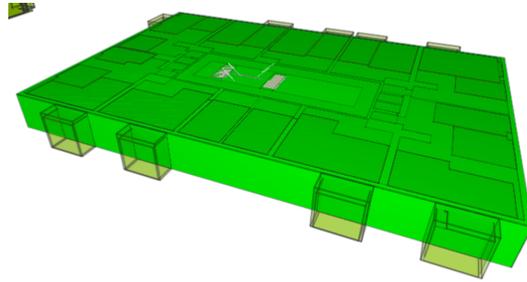


Figure 24: Example of one floor space in *Peak tower* BIM. It is a box including the whole storey, gross space inside the exterior walls and slabs. *Type BVO* (=GFA, gross floor area).

Space2 One includes the same, but with the exclusion of the vertical connection spaces and external walls (Figure 25);

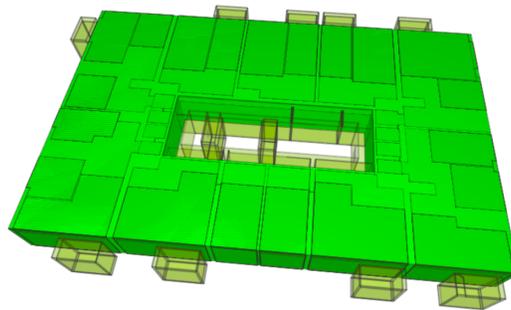


Figure 25: Example of one floor spaces in *Peak tower* BIM. It is a box including the whole storey, except for the vertical connection spaces and installations at the centre of the building and the external walls.

Space3 Subspaces include single apartments (e.g. Figure 26);

Space4 Next level is the room level, including both the single rooms and balconies. For the same room there are two identical spaces arriving at the top of the ceiling plus one finishing where the walls finish. In some cases, it's possible to observe that there are many overlapping spaces, all starting from the slab, sometimes with different heights but sometimes just repeated, which is weird, and would deserve to be investigated further with the help of designers (e.g. Figure 27).

Some remaining issues can be noticed, for example in the modelling of apartments (Figure 26), some of the internal walls are included and some other aren't. Maybe this is due to the generation of spaces starting from one of the models (e.g. the structural one, or the architectural one), which make the modelling software refer to the information there present.

Notwithstanding, such detailed modelling of spaces is a great opportunity for many computations.

For example, in the *Peak tower* BIM IFC models, the spaces describing the apartments are clustered as different types, which are stored under the "*Name*" attribute (e.g. "Type B 50–70 m²"; "Type C 70–90 m²"; "type E 70–140 m² F2"). This can be helpful for us in order to detect the *IfcSpaces* describing a whole apartment and isolate them to calculate the reference areas for parking places calculation.

Similarly, in the *Terrace tower* BIM, they are even more explicitly labelled with the *Name* and *Type* "Apartment".

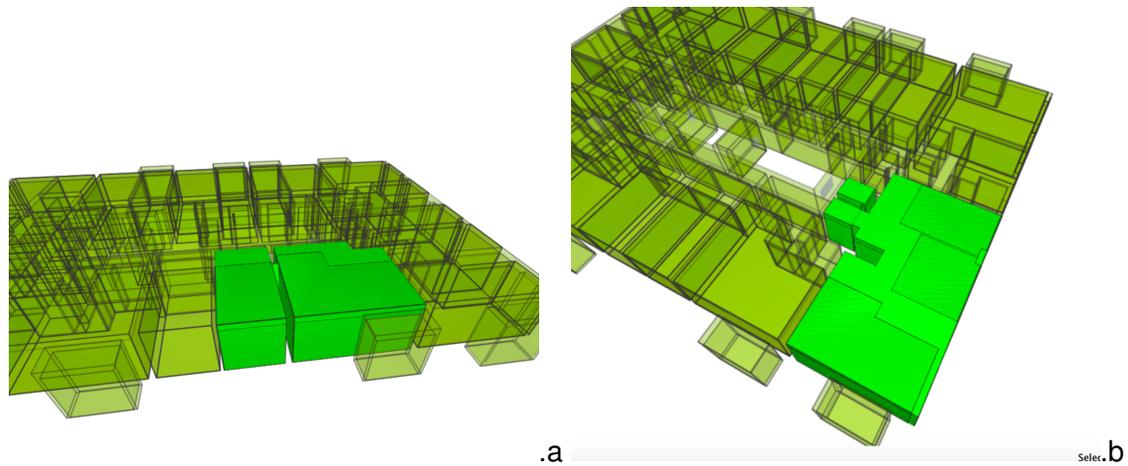


Figure 26: Example of apartment-level spaces. (a) IfcSpace type B50 – 70m² (area = 65.57 m²) interior: True. (b) Type e70 - 140m² F2 n. T1.17.6 (Area=82.68m²).

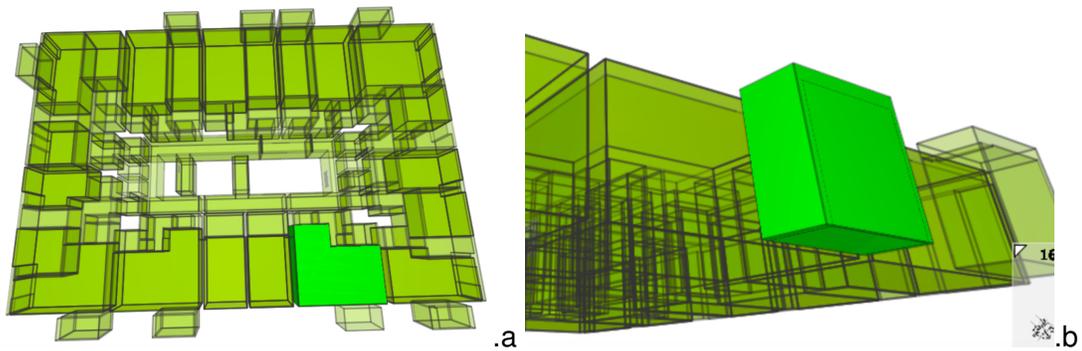


Figure 27: Example of room-level spaces. (a) Name balkon n.315 type balkon T01 315 / type Name balkon t1.17.7.90:8210271 Interior: True. (b). Woonfunctie t01 n.t1.17.7a.

The alternative to the use of provided spaces for measuring the dwellings area would be their inference starting from the architectural model, according to the paths allowed by the building elements dispositions.

The challenges involved in this can be mainly summarised (Figure 28) as:

Issue1 Some parts are missing (e.g. the structural elements are not in the architectural model, therefore it is again necessary to join the different models);

Issue2 Some parts are wrongly defined (e.g. walls instead of doors prevent a possible algorithm to understand path/non path).

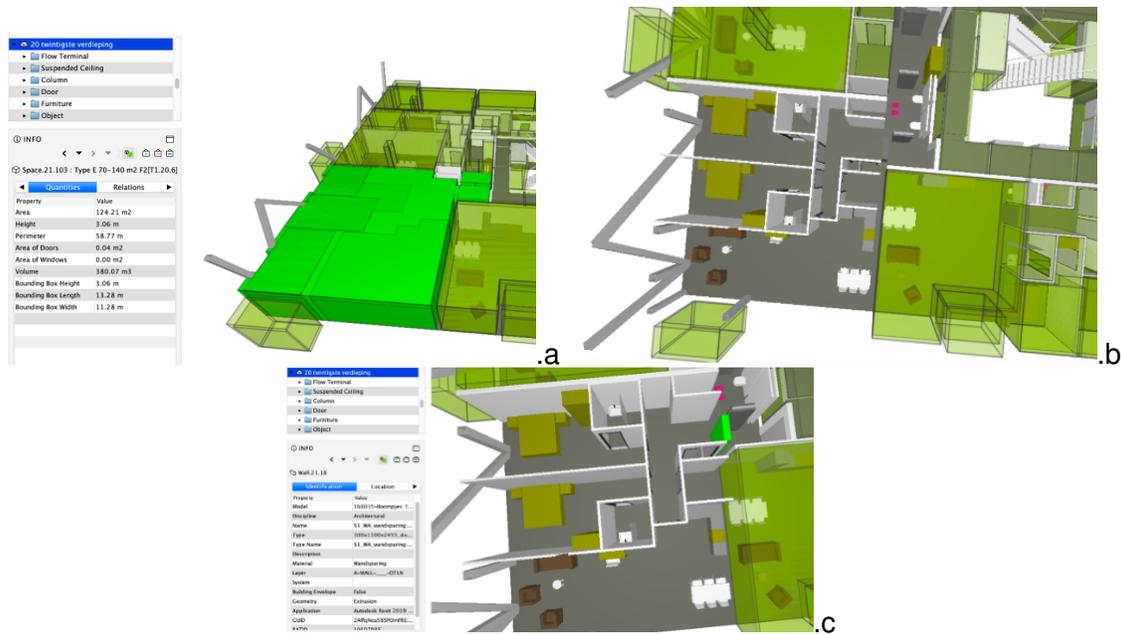


Figure 28: Example of one apartment represented in the *Peak tower* BIM: (a) The *IfcSpace* enclosing it; (b) How it appears in the architectural model, it is possible to notice the absence of (structural) walls; (c) The bright green element is an *IfcWall*, but it should be a door.

If effectively and consistently labelled with agreed types, they could even be used to extract the envelope useful for building dimensions checks.

However, looking at the IFC models there are few inconsistencies. *IfcSpaces* with the same label (stored as name and/or as type) can represent different spaces (Figure 29). On the contrary, redundant *IfcSpaces* representing the same volume could be labelled differently, but without explicit rules (Figure 30). In other cases, different spaces intersect (e.g. Figure 31) but it is not possible to understand the different meanings of them and consequently if and why are they allowed to overlap.

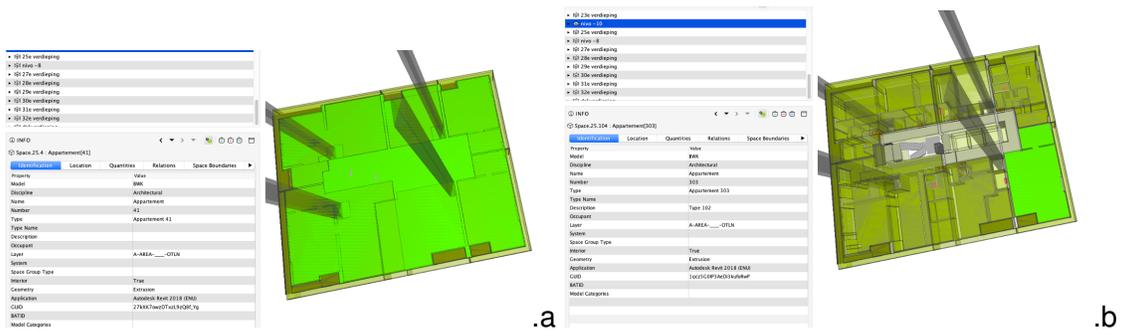


Figure 29: In the *Terrace tower* BIM, in the same floor, two *IfcSpace* entities (a) and (b) with the same name “apartment” represent two different things. In the type, a different number is added after the word “apartment”: 41 and 303; but this could refer to an internal code of the design team, whose meaning is unknown outside.

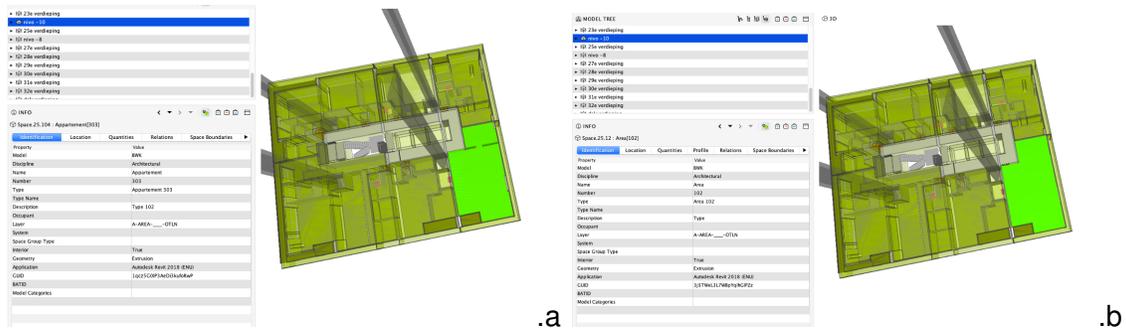


Figure 30: Example in the Terrace tower BIM, in the same floor, two *IfcSpace* entities describing almost the same volume are labelled differently: as “apartment” (a) and just generally as “area” (b).

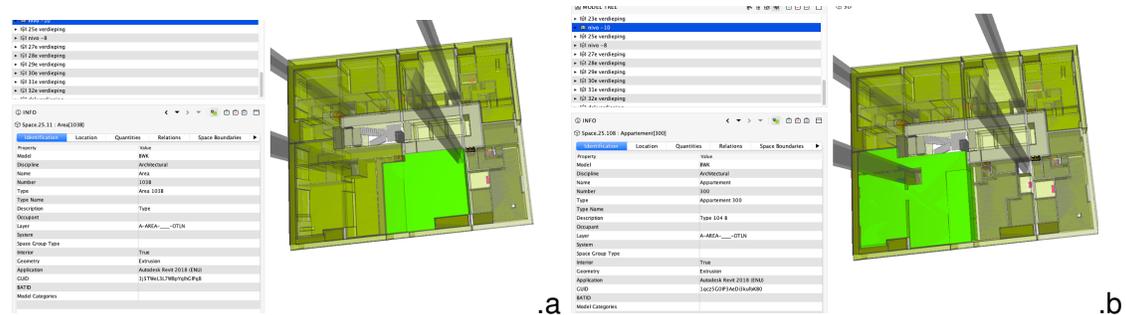


Figure 31: Example in the Terrace tower BIM, in the same floor, two *IfcSpace* entities describing almost the same volume are labelled differently: as “apartment” (a) and just generally as “area” (b).

A final remark is that such use of *IfcSpaces* is extremely useful to use the BIM more easily for automatic processing.

The gap which is still to be filled is the provision of an official agreed and useful values to describe spaces. Such enumeration is not provided yet by IFC. For the demonstrator developed within this project, the spaces describing apartments were searched and selected manually to check the parking regulation.

The definition of specific codelist values to be used for automating the system are proposed in this document (see Section 4.2).

3.3.7 Representation of parking places

In order to check the parking regulation, it is necessary to measure the gross floor area of each apartment and building unit, in association to its function, and to count the designed number of parking places for cars and for bikes.

The first part of the check could be effectively supported by the representation of apartments as *IfcSpaces*. However, they do not include the thickest walls and the facade (Figure 32), therefore they are not exactly the representation required, although the area value should be close.



Figure 32: *IfcSpaces* representing apartments. It is possible to notice how the gross floor area is not corresponding to the enclosed volume.

For the count of the parking places, we are supposed to consider both the *IfcBuilding* and the *IfcSite* (that should enclose the objects external to the building consistently).

In the *Peak tower* BIM, the parking places are represented by means of *IfcBuildingElementProxy*, with *Pset_ProductRequirements* → *Category*= Parking (Figure 33). It is therefore possible to count them: they are 57, distributed on the two underground floors.

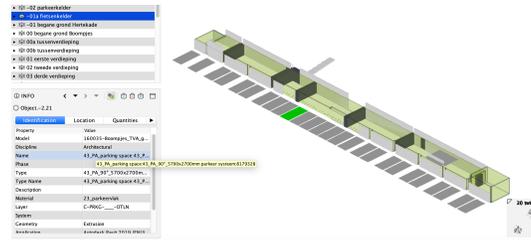


Figure 33: Parking places for cars in the Peak Tower BIM.

buildingSMART foresees the use of *IfcSpaces*, in association to specific properties, to represent parking places¹³. The use of such structures should be proposed as a guideline, and possibly their effectiveness could be discussed with architects and developers of the tools supposed to process the resulting BIM

The bike parking places are represented as *IfcSpace* with Name “fietsenstalling”. In addition, text is added reporting the number of bike places hosted in an area, but it is a very undefined representation even for a human check, on the one hand (and the designer should be trusted, since the area is not drawn specifically) and on the other hand, no specific entities or attributes are used to be automatically processed (Figure 34).

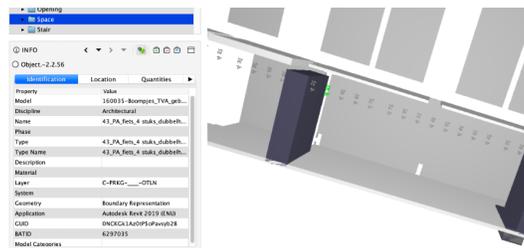


Figure 34: Parking places in the Peak Tower BIM for bikes.

In the Terrace tower BIM, one space is including the whole storey, labelled as “parkeren” and another overlapping one labelled as “stallingruimte (garage)”, inclusive of both the area of parking places and the path for cars movement (Figure 35). However, in the architectural BIM, neither the slab nor possible walls dividing the space are represented. These are probably included in the structural model, in which, however, no element can link the part of the model to the parking function.

¹³e.g. see <https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/FINAL/HTML/ifcproductextension/lexical/ifcspace.htm> with *Pset.SpaceCommon*: common property set for all types of spaces; *Pset.SpaceParking*: specific property set for only those spaces that are used to define parking spaces by *ObjectType* = 'Parking'; *Pset.SpaceParkingAisle*: specific property set for only those spaces that are used to define parking aisle by *ObjectType* = 'ParkingAisle'

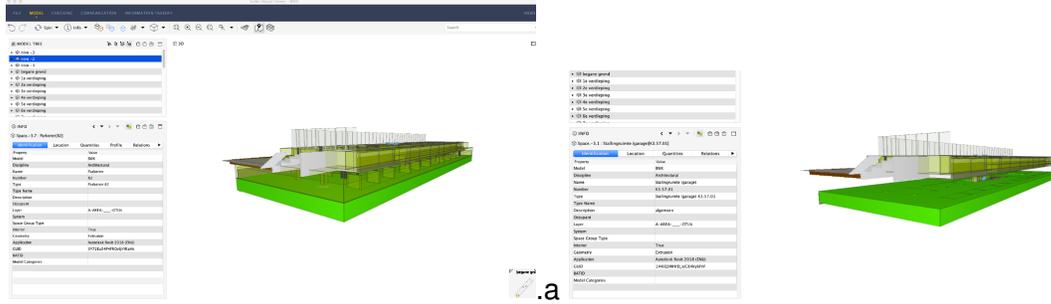


Figure 35: Parking places in the *Terrace Tower* BIM.

Another space in the *Terrace tower* BIM is representing “Fietsstalling” according to the name and type (Figure 36). In addition, there are boxes, represented as *IbcBuildingElementProxy* equally intended to represent parking spaces for bikes (Figure 37). However, there are a lot of overlapping similar geometry which make it not reliable.

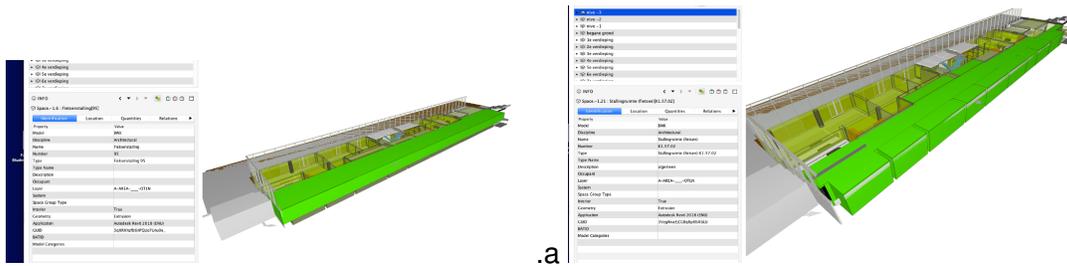


Figure 36: Parking spaces for bikes in the *Terrace tower* BIM. For some reasons, they are labelled differently: as fietstalling (a) and stallingruimte (fietsen) (b).

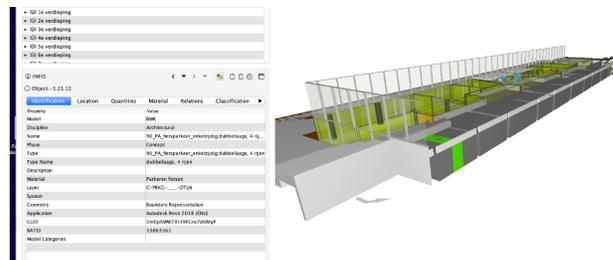


Figure 37: *IbcBuildingElementProxy* representing bike parking places in the *Terrace tower* BIM.

3.4 The implemented demonstrator

In this section, the algorithms of the tool implemented are described. Starting from the grouping of *IbcBuildingElements* in *IbcBuildingStoreys* as a base, we describe the storey overlap calculation algorithm (Section 3.4.1) and the overhang distance checking algorithm (Section 3.4.2).

It is possible to download the source code of the tool and the applications at https://github.com/twut/GEOBIM_Tool

3.4.1 Storey overlap calculation algorithm

The storey overlap calculation algorithm computes the overlap between the towers and the base. It contains the following steps.

First is the base selection. In the Rotterdam GeoBIM project, the ground floor of the BIM is selected as the base. Future development could make this selection automatically. The base selection is also able to be a user-defined option.

Second, each building storey is cut by a horizontal plane, as shown in Figure 38. In this step, the footprint of each building storey can be obtained by means of plane intersection with building element geometries. These footprints are stored as 2D polygons.

Third, the overlap percentage between the polygon of each floor and the polygon of the base is calculated by using polygon intersection.

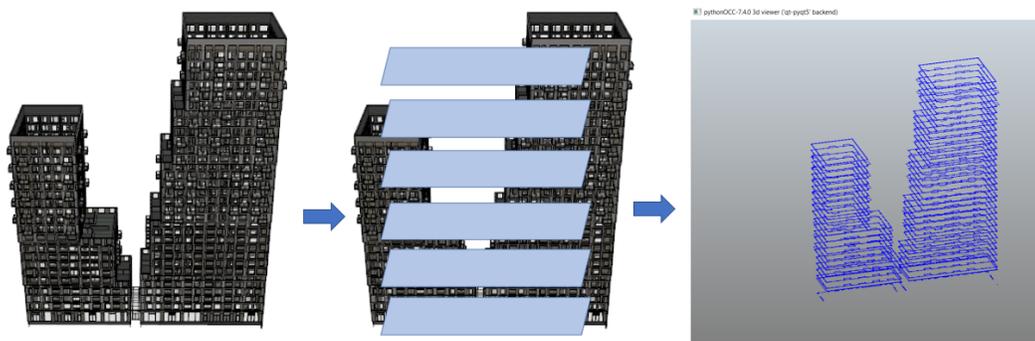


Figure 38: Section cut of each building storey with a horizontal plane, cutting result shown in the right

Details of balconies during cutting are illustrated here. The cutting height of one building storey has impact on the balconies of that floor. Figure 39 shows the cutting results of *Peak tower* BIM floor 6th. When the cutting height is equal to $lfcBuildingStorey.Elevation + 1.5$ m, the balconies are contained in the result polygon (Figure 39 in blue). When the cutting height equals $lfcBuildingStorey.Elevation + 2.5$ m, balconies can be excluded in the resulting polygon. Users can define the cutting height with respect to $lfcBuildingStorey.Elevation$ by themselves for their own requirements.

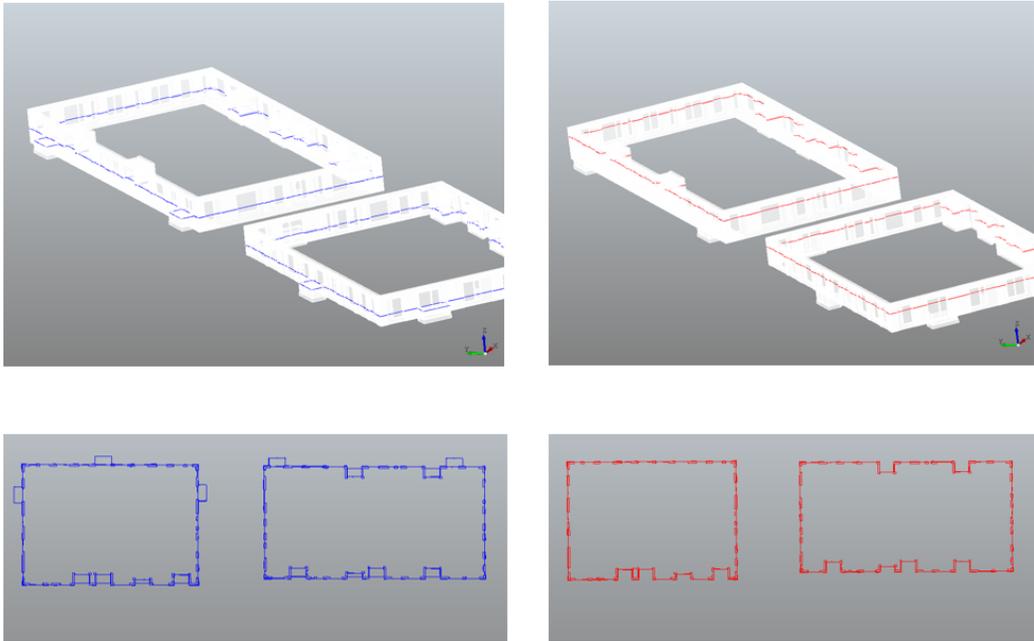


Figure 39: Result of the extraction of the storeys profile, using two different heights as input parameters (blue: balcony included, red: balcony excluded)

The number of polygons resulting from the cutting are automatically detected in this algorithm. Figure 40 gives an overview of the automatic polygon detection from the cutting result. The input of this process is the cutting shape obtained from the building storey cutting. The output are the detected polygons.

The first step is to sample points every 20 cm in each edge of the shape that was obtained from the building storey cutting. Second, clustering those points into different groups by using the Density-based spatial clustering of applications with noise (DBSCAN) algorithm [Ester, 1996]. The final step is to calculate the concave hull of each point cluster. The concave hull generation is based on the concave hull algorithm stated by Moreira and Santos [2007]. Each concave hull generated from this process will be considered as a polygon of the cutting result. In Figure 40, there are two point clusters generated by DBSCAN. Therefore, in the end two polygons are detected for the floor.

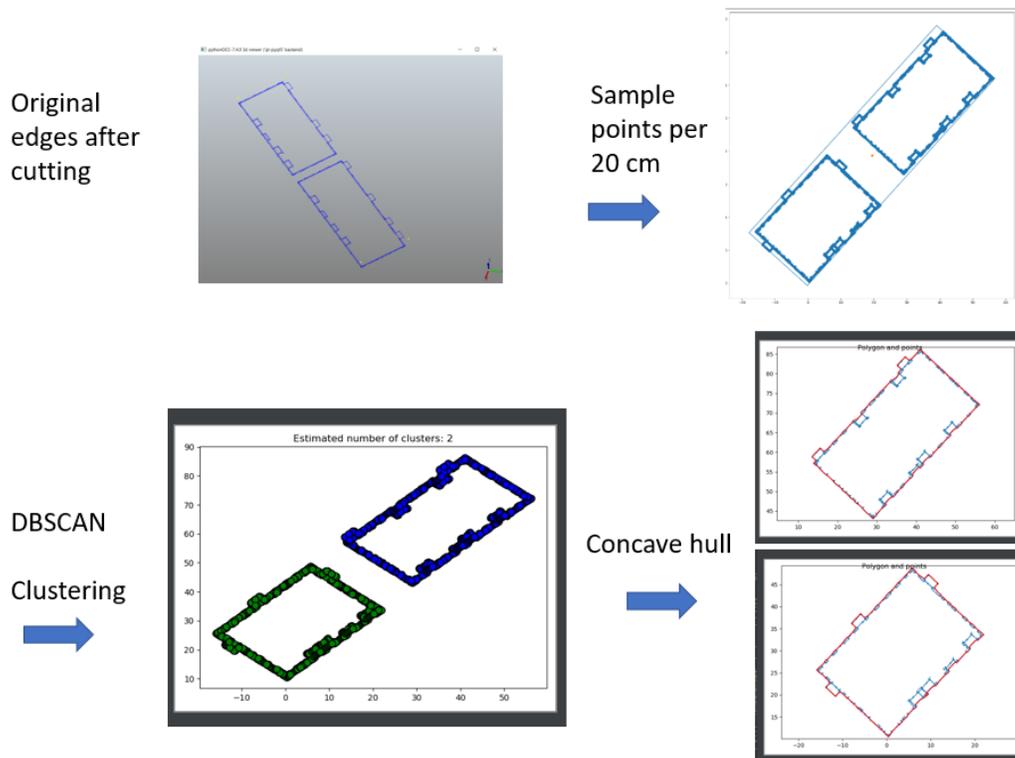


Figure 40: Overview of the automatic polygon detection from the cutting result

The concave hull implementation is based on the k-nearest neighbours algorithm [Moreira and Santos, 2007]. The input of the algorithm is a set of points and the output is a group of points which represents the concave hull of the input. In the case study, both the input and output points are 2D. The first step is to select a starting point, such as the point with the minimum Y value. In the second step, the starting point is considered as the current point. The k nearest neighbours of the current point are collected. The angles between the current point and neighbours are calculated. The one with the largest right-hand turn angle measured from the horizontal line is regarded as the point of concave hull. In the third step, the point with the largest right-hand turn angle is regarded as the current point. Then the calculation in the second step is repeated. Another concave hull point will be selected. The process is repeated until the start point is selected again.

3.4.2 Overhang distance checking algorithm

The overhang distance checking algorithm contains four steps. This algorithm is implemented by using *IfcOpenShell* [Thomas, 2020a] and the *pythonocc* library [Thomas, 2020b]. Since the model has been manually georeferenced, this process is based on the assumption that the georeferencing of the model is correct. Therefore, in the case study, the side A is Heterkade and the side B is Boompjes in Figure 41. The first step is to select the base. For this project, the ground floor is selected as the base to calculate the overhang distance. The second step is to extract the vertices from all the *IfcObjects* in the target floor using *pythonocc*. In the third step, shown in the right of Figure 41, the two highlighted lines of the base box are manually selected as the overhang calculation origins. The upper line is used to calculate the overhang distance towards Boompjes. And the lower line is corresponding to the Heterkade direction.

The final step is the calculation of the distance between each vertices and the selected lines. Then maximum distance to the upper line and lower line are the overhang distances in the North and South directions respectively.

Figure 41 illustrates the steps of the overhang calculations. In the left of Figure 41, the blue polygon is the base floor footprint and the red one represents target floor. In the middle of Figure 41, the red points are the vertices extracted from the geometries of the target floor `IfcObjects`. In the right, the distances between each red vertex and the highlighted lines are calculated.

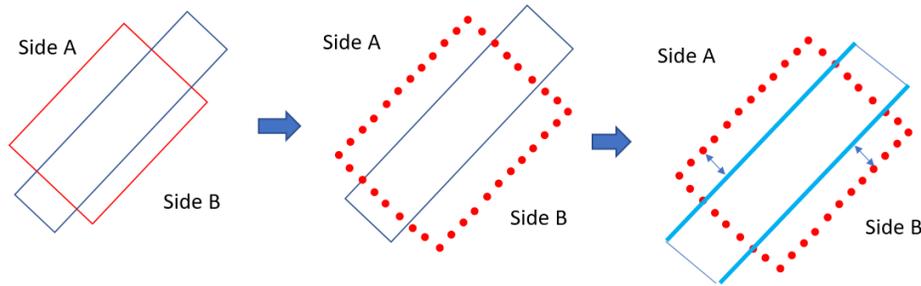


Figure 41: Overview of overhang distance calculation: red points are the vertices extracted from the geometries of the target floor `IfcObjects`. In the right, the distances between each red vertex and the highlighted lines are calculated

3.4.3 Tool performance

Table 1 gives an overview of the overlap information of the *Peak tower* BIM. The ground floor is chosen as the base during the overlap calculation. The ground floor has 100 percentage overlap with itself.

The building in this case is apparently not compliant with the regulation, since an overlap lower than 50% is only reached from the 23rd floor upwards, which is clearly above the lower part considered as “the base” of the building.

Table 1: Overlap percentage of *Peak tower* BIM, base: ground floor

Floor name	Overlap percentage(%)	Floor name	Overlap percentage(%)
ground floor	100	17th	67.9
1st floor	94.1	18th	67.9
2nd floor	93.8	19th	67.9
3rd floor	95.3	20th	64.0
4th	93.9	21st	63.6
5th	88.9	22nd	63.7
6th	85.9	23rd	39.2
7th	85.9	24th	39.2
8th	86.6	25th	39.3
9th	86.2	26th	35.0
10th	85.9	27th	35.0
11th	72.3	28th	35.1
12th	72.0	29th	35.0
13th	67.9	30th	35.2
14th	67.9	31st	35.0
15th	67.9	32nd	35.0
16th	67.9		

Table 3.4.3 shows the overhang result of the *Peak tower* BIM. The overhang distance towards the Hertekade side is 10.5 meters. For *Peak tower* side the overhang distance is 6.4 meters. However, the balconies are included in this attempt. The change of parameters could allow excluding them from the computation.

Table 2: Overhang result of the *Peak tower* BIM

Floor name	Overhang distance North, Hertekade side	Overhang distance South, Boompjes side
27th floor	10.5 meter	
12th floor		6.4 meter

Finally, the footprints extracted from each floor can be saved as a Well-known text files and, after georeferencing, they could be easily added to the city map (Figure 42).



Figure 42: WKT geometry representation in QGIS of ground floor (in orange), 7th floor (in green) and 29th floor (in pink).

3.4.4 Using the tool

The tool provided works on Windows. Once downloaded from https://github.com/twut/GEObim_Tool, it is possible to launch the .exe file directly, stored in the folder where the references files and folders are hosted (.py files, parameters and so on).

An interface will be opened, from where it is possible to open, load and visualize the IFC file and to use the provided tools. In the folder hosting the tool, it is possible to find the .yaml files specifying the useful parameters and the folders where the results will be stored.

In the github repository¹⁴, it is possible to find some documentation about the tool. A video attached to this document show some more detail about how to set parameters and where to read the results.

3.5 Guidelines

According to the experience we got both from the investigation of the topic and the data (Sections 3.2.2–3.3.7) and the implementation of the demonstrator (Section 3.4), we propose here some guidelines to modellers and planners.

3.5.1 Guidelines to BIM modellers

Semantic information also take a significant part of the BIM. For instance, the label of the *IfcSpace* mentioned in Section 3.3.6. Without the label, apartment recognition can be complex and challenging.

The guidelines useful to produce suitable IFC models for automatic processing them (especially, but not limited to, within the developed tool) are listed in Table 3.

¹⁴https://github.com/twut/GEObim_Tool

N.	Guideline	Aim
1	Group the objects external to the building in the <i>IfcSite</i> , separating them clearly from the elements which are part of the building (in <i>IfcBuilding</i>)	Avoid disturbance in the computation of the geometry of the building such as exterior envelope or bounding boxes
2	Care the grouping of <i>IfcBuildingStoreys</i> avoiding and fixing misassignments (i.e. inaccuracies pointed out in Section 3.3.1)	Avoid disturbance in the computation of envelope and bounding box of the building, as well as the measurements of dimensions such as maximum and minimum heights
3	Use correct semantics and avoid using <i>IfcBuildingElementProxy</i>	Support any automatic processing
4	Care the modelling of elements in order to avoid intersections	Support any automatic processing
5	Georeference the IFC model preferably with minimum LoGeoRef 30 (see Section 4.1)	Support integration with GIS and geoinformation
6	Model <i>IfcSpaces</i> for each room	Help conversions to boundary representation
7	Model an <i>IfcSpace</i> for each building unit, by assign the functions as proposed in Section 4.2 (at the suitable level of hierarchy)	Support the detection and interpretation of spaces and related useful calculations
8	Model the car parking places explicitly (geometry and attributes)	Support automation of car parking checks
9	Model the bike parking places explicitly (geometry and attributes)	Support automation of bike parking checks

Table 3: Initial guidelines to produce suitable IFC files

The guidelines n. 3, 8 and 9 are quite straightforward, although little helpful at the moment. However, more specific instructions and constraints should be provided, after considering the IFC model (and an eventual extension) in connection with the modelling practice.

3.5.2 Guidelines to geoinformation modellers

The 3D city model data of Rotterdam consists of the BGT dataset, flowerbed position, tree location and charging station place. However, the height information does not stored in the dataset. They are still 2D dataset. The height information could be saved as an attribute in the dataset, but still there is a gap between 2.5D and 3D.

In order to support the integration with the geoinformation for automating the checks, some information (accurately georeferenced in the national projected CRS EPSG:28992 and EPSG:7415) should be provided to designers as a preliminary step:

1. The parcels corresponding to the zoning, stored with explicit attributes and identifiers clearly referring to the regulation text (unambiguously).¹⁵
2. The city elements, in particular the streets and terrain, which are useful to assess the heights of

¹⁵For example, the regulation should be written as "in parcel with code. . . a rule is valid", instead of "in Boompjes 50-59" as it is written now, in the dimension regulation that we considered.

the building. In addition, the streets are also useful in this case to detect the direction “towards” them, on which the regulation depends.

3. An accurately measured network of reference 3D points (with approximated reciprocal distance of 50 m), to be used as control points in the georeferencing (and possibly as check points to assess its quality in the most difficult cases).

3.5.3 Guidelines to planners

First suggestion is to leave as little ambiguity and interpretation possibility into the regulation text, and quantify the involved parameters as much as possible. Similarly, the reference to available information should be done by means of formal identifiers (e.g. the code of the parcel instead of a general indication of the name).

A further work could provide more rules about how to support a more strict and systematic formalization of regulations.

4 From the detail to the GeoBIM: general topics arisen

Besides the specific issues related to the use case, some more general topics should be tackled. The ideal would be them to be proposed as extension to the standards or official and shared white papers. In the following sections, some examples related to the issue of georeferencing IFC (Section 4.1) and the values to identify and differentiate the *IfcSpaces* (Section 4.2) are described.

4.1 Georeferencing the IFC models

An important premise for the GeoBIM integration is that we have some rather accurate information about the georeferencing of the model available and explicit.

Georeferencing, which is straightforward and mandatory for geoinformation, is not usually applied to BIM, since the added value with respect to the building design workflows and construction management, which were the original aims of BIM, was not worth the effort of an accurate georeferencing calculation. This is especially true considering the technically complex nature of accurate georeferencing, which means that earlier BIM standards used simpler methods (e.g. lat-long), which have limitations in precision and when doing computations. Moreover, many 3D modelling and BIM software have accuracy issues when managing coordinates with very high values, such as those georeferenced using a large offset. However, recent applications (such as infrastructure design and facility management, besides GeoBIM integration) pointed out how georeferencing the BIM can enable further powerful functions. For this reason, in recent efforts, the software and standards intended to manage BIMs were extended to allow a more comprehensive storage of georeferencing information.

However, such implementations and the theory behind them are still immature or, at least, not well-aligned yet, so that it is still difficult to find perfect off-the-shelf tools allowing good control and accuracy when georeferencing. In addition, such information is not always stored consistently and documented explicitly, so that interoperability suffers. Furthermore, a human factor plays its role: georeferencing is not traditionally part of the background of building designers, who often cannot master it properly.

In this section, a general introduction to georeferencing and to the alternatives to store georeferencing in the IFC models is given, followed by a proposal of methodology supporting designers in the IFC

georeferencing, possibly assisted by a tool to be developed in future work.

4.1.1 Georeferencing introduction

Georeferencing are the methods used to define the location of a point on the surface of the Earth. Cartographic disciplines study it, together with the method of drawing such locations on a plane managing the deformations which are involved when reporting a curved and irregular surface on a plane one.

A document¹⁶ by a buildingSMART working group was recently published to begin filling such a gap in the BIM field. There, it is possible to find some very basic essential theoretical information about the georeferencing concepts.

Moreover, the difference between “small” and “large” sites is defined, with “small” sites measuring approximately less than one square kilometre. Therefore, most of the buildings considered for the regulations in this project fall within this category.

The Coordinate and Reference System (CRS) defines what is the reference grid, which can be based on angular distances (e.g. latitude and longitude) or in a Cartesian system (x, y) (in projected reference systems), with a point identified as origin. Historically, projected CRS were the standard choice in all geoinformation-related domains, since they are more accurate, and countries define systems allowing the representation of the national territory reducing deformations as much as possible. However, recent developments have resulted in an increased use of lat-long coordinates, such as the increased use of web maps and internationally standardised systems.

A database codifying a large number of existing CRS is the EPSG, within which the several parameters describing each system are stored and specified, together with the conversion parameters allowing the transformation of geometries from one representation in a certain CRS to another one. The same database defines identifiers (the EPSG code) that are largely implemented in software and adopted and well-known around the world. For this reason, it is wise to refer to these in order to uniquely identify each CRS (as opposed to setting parameters individually).

Some relevant CRS for this project are summarised in Table 4.

CRS	EPSG code	Short description
<i>World Geodetic System 1984 - WGS84</i>	4326	The name refers to the ellipsoid currently used by Global Positioning Systems. For this reason, as well as its use throughout the web, it is also the base for most world-wide and globe-based representations.
<i>Amersfoort / RD New - Netherlands — Holland — Dutch</i>	28992	National Dutch projected reference system for 2D representation
<i>Amersfoort / RD New + NAP height</i>	7415	Association of the Amersfoort CRS (EPSG:28992) and the national height reference system. Dutch reference for 3D coordinates.

Table 4: Synthesis of useful CRS for this project.

¹⁶<https://www.buildingsmart.org/wp-content/uploads/2020/02/User-Guide-for-Geo-referencing-in-IFC-v2.0.pdf>

4.1.2 3D Georeferencing parameters calculation

Proper georeferencing of an IFC file allows the link between the model of a single building or construction within its context and environment.

In order to georeference a BIM properly, it is necessary to apply suitable rotations and translations (and eventually a scale). [Ugglä and Horemuz \[2018\]](#) present an interesting and accurate study of three different approaches to georeference a BIM and can be taken as a reference for more specific cases.

The document published by buildingSMART¹⁷ proposes to apply a Helmert transformation¹⁸ with two points (foreseeing the use of multiple points in the final suggestions) and to treat the height separately, by applying a simple translation.

However, georeferencing with only two points can introduce large errors, even when the two points are chosen to be as far away as possible. Instead, in order to minimize the discrepancies between coordinates in the georeferenced (projected) system and the BIM (Cartesian) system of coordinates, a redundant system considering the third dimension (heights) directly and several points homogeneously distributed in the building site should be considered. Their coordinates are supposed to be known, measured in a reference network by specific survey campaign by the municipality, and the reciprocal transformations should be calculated starting from the coordinates of the corresponding points in the two models¹⁹.

The last observation regards the storage of the North direction. In the buildingSMART document, it is pointed out that the direction obtained by the Helmert Transformation calculation represents the grid North. Therefore, the calculation of such result should be “customized” based on the used projected CRS and the *TrueNorth* direction could be stored within the *IfcGeometricRepresentationContext* attribute. However, other options are possible based on the scheme described in the following section.

4.1.3 Georeferencing storage within the IFC model

There are several techniques to store georeferencing information in IFC, which involve trade-offs between difficulty/information required and the accuracy that can be achieved. A comprehensive such scheme is proposed by [Clemen and Goerne \[2019\]](#) (Table 5). The options range from basic address information over the specification of the geographic coordinates of a reference point to the definition of an offset between the project coordinate system and the global origin of a coordinate reference system and the corresponding rotation of the XY-Plane. The coordinates of a reference point (which is usually the origin of the local Cartesian system where the model is designed) are stored and sometimes complemented with a direction of the axis in the local system. This classification scheme is not officially defined in the IFC standard but can help practitioners to quickly assess which georeferencing information is available in the IFC file, and could be a suitable guide for developers to orient the related implementations.

¹⁷<https://www.buildingsmart.org/wp-content/uploads/2020/02/User-Guide-for-Geo-referencing-in-IFC-v2.0.pdf>

¹⁸https://en.wikipedia.org/wiki/Helmert_transformation

¹⁹https://en.wikipedia.org/wiki/Procrustes_analysis

LoGeoRef	Supported CRS	Storing entities
<i>LoGeoRef10</i>	No CRS, approximate location by means of the address.	<i>IfcPostalAddress</i> referenced by either <i>IfcSite</i> or <i>IfcBuilding</i> .
<i>LoGeoRef20</i>	WGS84 EPSG:4326	Attributes <i>RefLatitude</i> , <i>RefLongitude</i> , <i>RefElevation</i> within <i>IfcSite</i>
<i>LoGeoRef30</i>	Any cartesian CRS, including projected coordinates (CRS not specified in the file)	<i>IfcCartesianPoint</i> referenced within <i>IfcSite</i> (defining the projected coordinates of the model reference point); <i>IfcDirection</i> attribute of <i>IfcSite</i> (stores rotations regarding project or global north). ²⁰
<i>LoGeoRef40</i>	Any cartesian CRS, including projected coordinates (CRS not specified in the file)	Attribute <i>WorldCoordinateSystem</i> storing the coordinates of the reference point in any cartesian CRS (including the projected ones) and direction <i>TrueNorth</i> . Both are stored within <i>IfcGeometricRepresentationContext</i> . ²¹
<i>LoGeoRef50</i>	Specific projected CRS, specified by means of the EPSG code	IFC v.4 only. Coordinates of the reference point stored in <i>IfcMapConversion</i> using the attributes <i>Eastings</i> , <i>Northings</i> and <i>OrthogonalHeight</i> for global elevation. Rotation for the XY-plane stored using the attributes <i>XAxisAbscissa</i> and <i>XAxisOrdinate</i> . The coordinate reference system (CRS) used is specified by <i>IfcProjectedCRS</i> in the attribute <i>Name</i> by means of the proper EPSG code.

Table 5: Synthesis of *LoGeoRefs* as defined by Clemen and Goerne [2019].

4.1.4 Proposals of concrete solutions for BIM models georeferencing

In this section, a workflow and a method are proposed to help designers to apply georeferencing to IFC files properly.

First of all, considering the Levels of georeferencing (LoGeoRef) of IFC defined by Clemen and Goerne [2019], the worst solutions are the cases in which either only the address of the building is known (LoGeoRef 10), or nothing. From the point of view of GeoBIM integration there is very little difference between those two cases. Even the use of LoGeoRef 20 implies errors in the results that are too high for many applications²².

Therefore, the minimum desired levels of georeferencing should be either LoGeoRef 30 or LoGeoRef 40. That means that, at least, a projected national coordinate reference system (CRS) has to be used to define the coordinates of the reference point, and national geoidic heights for its elevation. Moreover, the orientation of the model must be known and the rotation towards the True North of the y axis must be accurately stored in the file. Such information should likely come from suitable survey data in the area, or from similarly accurate reference points.

An alternative to the direct inclusion of LoGeoRef 30 or higher in the IFC files, is the inclusion in the BIM of one or more objects making explicit reference to existing immovable objects (materialising reference to coordinates and directions). These could be most likely imported in the BIM software from an existing provided dataset (e.g. the 3D city model, the zoning polygons or the city reference point network). In this case, the georeferencing data allowing to rotate and move the model to the correct reference system are provided and calculated from those known elements.

For example, reliable reference points could be provided by the city (e.g. within the 3D city model or among the geoinformation provided to support the designers), to be imported in the BIM software by

²²https://wiki.openstreetmap.org/wiki/Precision_of_coordinates, <https://confluence.qps.nl/qinsy/latest/en/world-geodetic-system-1984-wgs84-182618391.html>

the designers and exported together with the BIM itself. In this way, the necessary transformations to georeference the BIM would be much more reliable. Moreover, in case of huge BIMs, it would be possible to calculate how to segment the model in order to get the lowest residuals, if considering some of those points as check points. To support this, the city of Rotterdam provided us an available network of points (Figure 43), although some remaining issues (e.g. they have an uncertain source — GPS, network densification, etc.; they are in the Dutch national CRS, but mostly created in local networks which are referred to it; it is not certain that every point is existing; the database has not been updated for a few years).

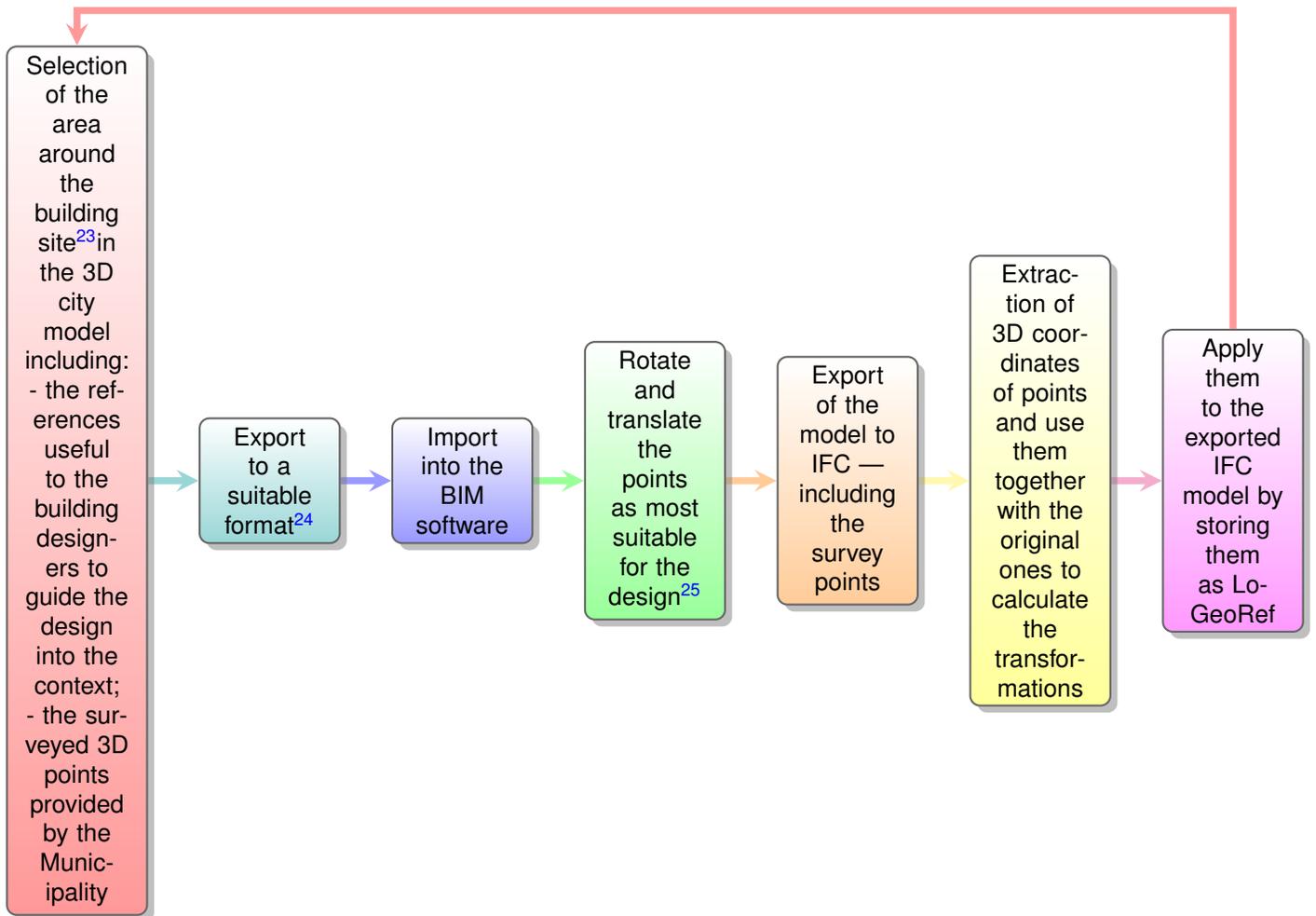


Figure 43: The orange points are part of the point network available in Rotterdam; the yellow ones in the image on the right could be the ones useful for georeferencing.

In both cases, the information about what CRS are used needs to be provided, for example as a note or in some attribute in the case it is not foreseen by the data model (i.e. in the models having previous IFC version than IFC4).

Such georeferencing information must be added to all the models composing the total BIM. However, it is easy to transpose the results of the calculations to all of them, assuming that they are correctly and consistently reciprocally registered in a common Cartesian system. This is actually an important essential premise, to be translated to a requirement for BIM designers, but which is current practice already.

A useful reference workflow could be:



4.2 IfcSpaceType enumerations extension

The use of IfcSpaces can be useful to support automation in some tasks, by referring directly to the volume modelled by the designer instead inferring or measuring it on the model. Moreover, the addition of useful attributes is enabled, for example specifying the function of the considered space, which conditions the rule applications (parameters often depend on the spaces functions).

The most important example for this project is the parking regulation: the number of parking places to be provided depends on the number, area and function of the planned building units.

In addition, the modelling of spaces in both the BIMs considered as case studies (Section 3.3.6) show that an effort of designers in this direction is already being made.

Starting from these two facts, we list here the values that can be considered for filling the attribute Type

²²around 100–200 m buffer around the site?

²³IFC could be sensible as IfcProxy/IfcSite

²⁵they could keep track of such transformations directly, but it is likely better not to ask designers to manage the georeferencing: possible inaccuracies; changes and moves during the modelling; collaboration with many people and exchanges of models

of the modelled IfcSpaces.

For this reason, in this document that part of the problem is considered, with reference to the values foreseen by the Rotterdam parking regulation.

However, the classification used in the case studies BIM will also be explicated, since probably used by designers and useful to implement other kinds of checks (for example the restrictions on specific rooms dimensions or ventilation, noise values and so on).

The space classification topic is not easily solved. A high number of classifications of spaces exist:

- CityGML v.2.0 codelist for Room “class”;
- CityGML v.2.0 codelist for Room “function and usage”;
- Landinfra “CondominiumUseType” codelist;
- Landinfra “BuildingPartType” codelist;
- INSPIRE data model for Buildings²⁶ “CurrentUseValue” codelist;
- INSPIRE data model for Buildings²⁷ “BuildingNatureValue” codelist;
- in the Getty Vocabulary Art and Architecture Thesaurus²⁸, as subclasses of “singlebuilt works by functions”²⁹;
- others related to real estate fields³⁰
- others related to facilities representation and management³¹

The enumeration IfcSpaceTypeEnum is foreseen by the IFC model (v.4), but at the moment it’s quite elementary (see Figure 44).

²⁶<https://inspire.ec.europa.eu/id/document/tg/bu>

²⁷<https://inspire.ec.europa.eu/id/document/tg/bu>

²⁸<https://www.getty.edu/research/tools/vocabularies/aat/about.html>

²⁹<https://www.getty.edu/vow/AATHierarchy?find=church&logic=AND¬e=&subjectid=300004894>

³⁰https://www.archibus.net/ai/abizfiles/v21.2_help/archibus_help/Subsystems/webc/Content/web_user/space/inventory/bldg_perform/room_category_import_standards.htm

³¹<https://nces.ed.gov/pubs2006/ficm/content.asp?ContentType=Section&chapter=4§ion=1&skip=chapter&skiptitle=4%2E+Space+Use+Codes>

IfcSpaceTypeEnum

Definition from buildingSMART: This enumeration defines the available generic types for *IfcSpaceType*.

HISTORY New enumeration in Release IFC2x Edition 3.

Enumeration

- USERDEFINED
- NOTDEFINED

EXPRESS specification:

```
TYPE IfcSpaceTypeEnum = ENUMERATION OF
( USERDEFINED,
  NOTDEFINED );
END_TYPE;
```

5.4.2.13 IfcSpaceTypeEnum



▼ Natural language names

EN	Space Type Enum
FR	Énumération des types de locaux

▼ Change log

Item	SPF	XML	Change	Description
IFC2x3 to IFC4				
IfcSpaceTypeEnum				
SPACE			ADDED	
PARKING			ADDED	
GFA			ADDED	
INTERNAL			ADDED	
EXTERNAL			ADDED	

5.4.2.13.1 Semantic definitions at the type

▼ Type definition

This enumeration defines the available generic types for *IfcSpace* and *IfcSpaceType*.

HISTORY New enumeration in IFC2x3.

IFC4 CHANGE The enumerators *INTERNAL* and *EXTERNAL* have been added for upward compatibility to replace *InteriorOrExteriorSpace* usage.

▼ Enumeration definition

Constant	Description
SPACE	Any space not falling into another category.
PARKING	A space dedication for use as a parking spot for vehicles, including access, such as a parking aisle.
GFA	Gross Floor Area - a specific kind of space for each building story that includes all net area and construction area (also the external envelop). Provision of such a specific space is often required by regulations.
INTERNAL	
EXTERNAL	
USERDEFINED	
NOTDEFINED	

Figure 44: Side view of the architectural model of Peak tower, from where it is possible to see that there is no connection among floors, since the structural slabs are missing.

Alternative reasonable works could be carried out. For example a mapping among all those classifications could be an option. Another case could be the selection of some parts of them based on identified criteria, most likely based on the useful ones for practice or supporting specific tasks. This reasoned and comprehensive study could be one of the future developments of this work.

At the moment, we limit our contribution at listing the functions of building units present in the parking regulations (Table 6), based on which the calculation of parking places is performed.

On the other hand, we also report the classification used to label the spaces in the two datasets used as case studies (the Boompjes BIM and the Terrace Tower BIM) (Table 7).

The need of a more complex study to propose a suitable classification also comes from the necessity of structuring it according to several criteria.

One of these is the hierarchy in functions (e.g. under the category “shop” many kinds of shops can be included, such as small showrooms or commercial centres). This is the case of the classification needed for the parking calculation, which is related to building units (apartment, shop, office, etc.).

This one can be in turn related to different levels of a meronymic (part-of) structure of spaces and aggregations possibly enclosed within spaces, as it happens in the two examples of BIMs for this project. Therefore, it is possible to find, for example (from the small to the large): part of the room - the single room - the building unit - the building storey - possibly larger aggregations.

Third point of complexity, each of these parts could be defined for specific use cases, and they would have different configurations based on specific needs. Some examples are the different zones for energy analysis, the volumes enclosing the walls for gross floor area calculation, the spaces including the spaces above the compound ceiling or not which is possible to find now in the inspected BIM ans so on.

Function	Sub-function
home	
to work	
	office with counter function (banks, post offices) of- fice
	labor intensive / visitor intensive company (industry, laboratory, workshop etc.)
	labor-intensive / visitor-extensive company (ware- house, storage, transport company, etc.)
	Multi-company building
shop	
	store
	convenience store
	large-scale retail trade
	hardware store / garden center / thrift shop
	showroom
Sports and recreation	
	indoor gym / sports hall
	outdoor sports ground (per ha net)
	dance studio / gym
	squash court (per court)
	tennis court (per court)
	golf course (per hole)
	bowling center / billiard room (per court / table)
	swimming pool (per 100 m2 surface basin)
	allotments / utility gardens (per garden)
	riding school (per box)
	marina (per berth)
	event hall / exhibition building / conference building
	indoor playground / hall
Culture	
	museum / library
	cinema / theater / theater (per seat)
	social cultural center / community center / funeral home religious building (church, mosque, etc.) per se. / visitor pl.)
	cemetery (by simultaneous funeral)
catering industry	
	cafe / bar
	restaurant
	hotel (per room)
	disco / party room
	cafeteria / snack bar
Education	
	crèche / playgroup / nursery / after-school care
	primary education (per classroom of 30 ll)
	Preparatory daytime education (vmbo, havo, vwo, per 30 ll classroom)
	vocational education day (MBO, ROC, HBO, WO)
healthcare	
	hospital (per bed)
	nursing home (per residential unit)
	pharmacy
	1 e health care line (general practitioner, dentist, therapist) (per treatment room)

Table 6: Functions listed in the parking regulation.

Level of specification 1	Level of specification 2	Level of specification 3
area	Apartment / woonfunctie Commerciele ruimte Bijeenkomstruimte – meeting room Parkeren stallingruimte (garage) stallingruimte (fietsen) diversen	gang - corridor leftenhall - lift hall Leidingschacht (toegankelijk)- Pipe shaft (accessible) Toilet Brandweerlift – fire lift Liftkool – elevator cabbage Hoofdentree (tochtsluis) - Main entrance (draft lock) Flatrek Hoogspanningsruimte - High voltage room Inkoopruimte - Purchasing space Laagspanningruimte - Low voltage space Containerruimte In- en uitrit PG Pompruimte sprinkler Stadsverwarming - district heating sprinklerruimte fietsenstalling balcony terras woonkamer slaapkamer loggia badkamer berging hal garderobe

Table 7: Labels assigned to IfcSpaces in the two BIM models of the case study.

5 Discussion

The experiment done in this project allowed first of all the explication of the issues involved in the task of using digital models and information to automate the regulation checks for building permit issuing.

Although standards are available for storing BIM models and geoinformation in shared formats, the way they are found in practice is still preventing their best potential.

Starting from specific and concrete data, this study could point out the most frequent issues in the models, to be translated to guidelines and aspects to care during the data production phase.

Once these were clarified and fixed, a demonstrator was implemented to automatically check the chosen regulations. Unless the information in the models was insufficient or not fixable, the two IFC models were used as base for the checks. The handling of IFC files was way more tricky than expected, and a lot of effort was necessary to extract the proper geometry of the building. For this reason, some of the planned steps were not performed, such as a suitable connection with the 3D city model to automate the checks further, besides increasing their accuracy (e.g. by considering the correct height as reference).

However, the issues tackled and underlined within this project are common to most part of the BIMs and solving them would be functional to many regulations. Therefore, they constitute the necessary starting

point for the development of more powerful and scalable methodologies, guidelines and tools. Moreover, some of the difficult points regarding modelling, standardization and representation together, which emerged in the project, would need wider collaborations to be tackled. For example, the involvement of designers to agree on guidelines would be one of the most urgent actions.

The tool developed within the project, although still in an initial demonstrator phase and improvable with respect to performance, attractiveness of visualization and user-friendliness (e.g. the transformation in a web-based tool could be one of the next interesting steps for the sake of usability), is built on solid theoretical, open standard-based and data-based bases. This makes it effective and able to support the checks of the considered regulation.

In particular, according to the experience of measuring the ground truth manually within spread and high-performance BIM software (Autodesk Revit) and viewers (Solibri Model Viewer), it must be observed that the measurements are not very fast. In fact, even if the person making the checks is an expert user, time is necessary, for example, for the loading of the model (or likely more than one) within the software, for the first time; for reloading of the model while changing view, measuring or query the model elements to check their semantics and specific features, plus the time needed to make the measurements and do the checks (possibly after the needed calculations). The specific times can vary a lot based on the used computer and on the dimension of the BIM.

However, considering a rough approximation, about one hour is necessary to assess the building from the BIM by humans, by means of available tools. Another example is the height of the streets to be considered for the measurement of the building heights.

In order to assess and discuss the obtained results and progress of the project in higher detail, the table in Annex I is filled.

6 Conclusions

The paper presents the work developed for implementing a tool and a methodology able to support the municipality of Rotterdam in building permits regulation checks by means of digital solutions, starting from digital standardised datasets (mainly IFC BIM and CityGML 3D city model). The approach was completely bottom-up, starting from a very specific case study (one regulation and few datasets per type).

This allowed the explication of several sub-issues which need to be overcome before solving the most apparent steps in the building permit workflow. Addressing them could allow the implementation of a demonstrator, working with initial solutions and promising to be possibly further improved and scaled for actually supporting the municipality in the building permit task on a wider scale.

This approach began slow, since it was necessary to start from small basic points and experiment with the available data in order to achieve the necessary knowledge for formulating a suitable methodology, to be implemented in a tool. From now on, the continuation of such a project could be easier, since several basic issues, which are common to many processes useful for many regulations, are already tackled and the problems are already well-defined, which make the achievement of a solution more feasible.

Future directions could effectively regard the improvement of the tool in order to test alternative algorithms detecting the most useful exterior facade geometry to support the checks and the extension of the research and tool to make it able to check dimension regulation for the whole city, by improving customizability.

References

- Christian Clemen and Hendrik Goerne. Level of Georeferencing (LoGeoRef) using IFC for BIM. *Journal of Geodesy*, (10):15–20, 2019.
- M Ester. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, No. 34, pp. 226-231)., 1996.
- Adriano Moreira and Maribel Yasmina Santos. Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points. 2007.
- F. Noardo, C. Ellul, L. Harrie, I. Overland, M. Shariat, K. Arroyo Ogori, and J. Stoter. Opportunities and challenges for geobim in europe: developing a building permits use-case to raise awareness and examine technical interoperability challenges. *Journal of Spatial Science*, 2019.
- Francesca Noardo, Teng Wu, Ken Arroyo Ogori, Thomas Krijnen, Hasim Tezerdi, and Jantien Stoter. GeoBIM for digital building permit process: learning from a case study in Rotterdam. In *15th 3D GeoInfo Conference 2020*, ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. ISPRS, 2020a.
- Francesca Noardo, Teng Wu, Ken Arroyo Ogori, Thomas Krijnen, Hasim Tezerdi, and Jantien Stoter, 2020b.
- Krijnen Thomas. IfcOpenShell, 2020a. URL <https://github.com/IfcOpenShell/IfcOpenShell>.
- P Thomas. Pythonocc library, 2020b. URL <https://github.com/tpaviot/pythonocc>.
- Gustaf Ugglar and Milan Horemuz. Geographic capabilities and limitations of industry foundation classes. *Automation in Construction*, 96:554–566, 2018.