

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

MCDONNELL ARAÚJO MAIERON

**Integração de dados abertos na geração de
modelos 3D baseados em CityGML**

Dissertação apresentada como requisito parcial
para a obtenção do grau de Mestre em Ciência da
Computação

Orientador: Prof. Dr. José Palazzo Moreira de
Oliveira

Porto Alegre
2021

CIP — CATALOGAÇÃO NA PUBLICAÇÃO

Maieron, McDonnell Araújo

Integração de dados abertos na geração de modelos 3D baseados em CityGML / McDonnell Araújo Maieron. – Porto Alegre: PPGC da UFRGS, 2021.

89 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2021. Orientador: José Palazzo Moreira de Oliveira.

1. Cidades inteligentes. 2. Integração de dados. 3. Modelo 3D. 4. CityGML. I. Palazzo Moreira de Oliveira, José. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos André Bulhões Mendes

Vice-Reitor: Prof. Patricia Helena Lucas Pranke

Pró-Reitor de Pós-Graduação: Prof. Celso Giannetti Loureiro Chaves

Diretor do Instituto de Informática: Prof. Carla Maria Dal Sasso Freitas

Coordenador do PPGC: Prof. Luciana Salete Buriol

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“Mapear, nobre missão, é ter na mão nosso Brasil. Mapear, feito que encerra o
povo e a terra num só perfil.”*

— CEL ENG CART AYRTON DE OLIVEIRA CRUZ

AGRADECIMENTOS

Primeiramente a Deus, por iluminar a trajetória da minha vida, me dando sabedoria e confiança para alcançar meus objetivos em mais esta desafiante etapa.

Ao Prof. Dr. José Palazzo, meu orientador, por dar um voto de confiança a um iniciante na vida científico acadêmica. Sempre disponível pra passar um pouco da sua experiência e ensinamentos, que foram fundamentais para o bom andamento da pesquisa. Serei sempre grato pela oportunidade de trabalhar sob a sua tutela.

Aos colegas do grupo de pesquisa, em especial, Diego, Gabriel, Guilherme, Leonardo e Maran, pelas dicas e orientações sobre a vida acadêmica, tanto nas reuniões presenciais quanto nos encontros virtuais na sala do MConf, necessários em tempos de pandemia.

À Diretoria do Serviço Geográfico do Exército Brasileiro, pela confiança e apoio logístico no cumprimento dessa importante missão.

À minha amada esposa, companheira e amiga, Camila, por estar sempre ao meu lado, dando suporte em todos os momentos. Obrigado pela paciência nos momentos em que minha ausência foi necessária e pelo incentivo e apoio às minhas decisões.

Aos meus pais, Benhur e Cida, por terem me dado os ensinamentos e a base, não só intelectuais mas também de caráter para saber enfrentar os desafios da vida.

RESUMO

Diante da crescente complexidade dos grandes centros urbanos causada pelo aumento populacional e pela natureza dinâmica das cidades, seus administradores buscam otimizar os serviços e infraestruturas em termos de escalabilidade, ambiente e segurança para que sejam adaptados à demanda, fazendo com que suas cidades sejam, cada vez mais, inteligentes. Dessa forma, técnicas eficientes que possam auxiliar os administradores desses novos centros modernos na geração, gerência, manutenção e controle da infraestrutura física e de dados associada a todos os componentes das cidades inteligentes têm especial importância. Além disso, a adoção de modelos que estejam alinhados com a iniciativa global de dados abertos governamentais é tida como boa prática, para que os dados gerados e disponibilizados sejam amigáveis, reutilizáveis e redistribuíveis. Como ponto de interseção entre a infraestrutura física e de dados, os modelos 3D de cidades vêm tendo um papel cada vez mais importante na rotina diária das pessoas, sendo elemento fundamental para diversas aplicações. Como possível solução de modelo 3D que cubra tanto aspectos semânticos como geométricos das feições, existe o CityGML, modelo comum de informação semântica para representação de objetos urbanos em 3D. Tal modelo vem sendo adotado mundialmente e diversas cidades, em alinhamento com a iniciativa de dados abertos governamentais, disponibilizam seus modelos nesse padrão para compartilhamento. Tendo em vista que muitas cidades não possuem uma base de dados baseada em CityGML, mas possuem dados cadastrais em 2D e dados de altimetria, o presente trabalho traz uma abordagem de integração de dados abertos na geração semiautomática de modelos 3D baseados em CityGML, complementando informações semânticas sobre as instâncias com a associação com a base de dados do *OpenStreetMaps*. A abordagem é implementada em um estudo de caso utilizando dados disponibilizados pela Prefeitura Municipal de Porto Alegre-RS. O modelo gerado em CityGML passa por validações semânticas, geométricas e em nível de esquema, comprovando a exequibilidade da abordagem proposta. O *framework* disponibilizado pode ser empregado em um contexto geral, necessitando pequenas adaptações de acordo com os insumos utilizados.

Palavras-chave: Cidades inteligentes. integração de dados. modelo 3D. CityGML.

Open data integration in 3d models based on CityGML generation

ABSTRACT

Facing the increasing complexity of large urban centers caused by population growth and the dynamic nature of cities, their administrators seek to optimize services and infrastructures in terms of scalability, environment and security so that they are adapted to demand, making their cities smarter. Thus, efficient techniques that can assist the administrators of these new modern centers in the generation, management, maintenance and control of the physical and data infrastructure associated with all components of smart cities should be of particular importance. In addition, the adoption of models that are aligned with the global government open data initiative is considered good practice, so that the data generated and made available can be friendly, reusable and redistributable. As a point of intersection between physical and data infrastructure, 3D models of cities have been playing an increasingly important role in people's daily lives, being a fundamental element for several applications. A possible 3D model solution that covers both semantic and geometric aspects of the features, is CityGML, a common model of semantic information for the representation of 3D urban objects. This model has been adopted worldwide and several cities, in line with the government's open data initiative, make their models available in this standard for sharing. Bearing in mind that many cities do not have a database based on CityGML but have 2D cadastre data and altimetry data, the present work brings an approach of integrating open data in the semi-automatic generation of 3D models based on CityGML, "enriching" semantic information about the instances with the association with the OpenStreetMaps database. The approach is implemented in a case study using data provided by the Municipality of Porto Alegre-RS. The model generated in CityGML goes through semantic, geometric and schema level validations, proving the feasibility of the proposed approach. The available framework can be used in a general context, requiring minor adaptations according to the inputs datasets.

Keywords: smart cities, data integration, 3D models, CityGML.

LISTA DE FIGURAS

Figura 1.1	Principais componentes de cidades inteligentes	12
Figura 2.1	Exemplos de inconsistências entre bases heterogêneas.....	20
Figura 2.2	Diferentes níveis de detalhamento (LoD).....	22
Figura 2.3	Modulos do CityGML	24
Figura 2.4	Modelagem em UML do módulo <i>Core</i>	25
Figura 2.5	Modelagem em UML do módulo <i>Building</i>	26
Figura 2.6	Modelagem em UML do módulo <i>Transportation</i>	28
Figura 2.7	Diferença de representação do módulo <i>Transportation</i> de acordo com o nível de detalhamento	28
Figura 2.8	Modelagem em UML do módulo <i>Vegetation</i>	29
Figura 2.9	Modelagem em UML do módulo <i>WaterBody</i>	30
Figura 2.10	Modelagem em UML do módulo <i>LandUse</i>	31
Figura 2.11	Modelagem em UML do módulo <i>Relief</i>	32
Figura 3.1	Diferença entre MDS e MDT	39
Figura 3.2	Fluxo de trabalho para o módulo <i>Building</i>	41
Figura 3.3	Fluxo de trabalho para o módulo <i>CityFurniture</i>	43
Figura 3.4	Fluxo de trabalho para o módulo <i>Vegetation</i>	44
Figura 3.5	Fluxo de trabalho para o módulo <i>WaterBody</i>	46
Figura 3.6	Exemplo de malha viária em amarelo sobreposta em imagem aérea	47
Figura 3.7	Fluxo de trabalho para o módulo <i>Transportation</i>	48
Figura 3.8	Hierarquia de camadas no módulo <i>LandUse</i>	49
Figura 3.9	Fluxo de trabalho para o módulo <i>LandUse</i>	50
Figura 4.1	Área de teste em destaque na cor amarela e contorno externo do município em branco.....	51
Figura 4.2	No mosaico, os contornos amarelos representam a base de dados do OSM, os contornos vermelhos representam as feições da base de dados oficial que não foram associadas à base do OSM e o contorno azul as feições que tiveram relacionamento.....	56
Figura 4.3	Feição do tipo ponto da base do OSM fornecendo atributos de restaurante conforme tabela lateral.....	58
Figura 4.4	Detalhamento de feições com blocos construtivos consecutivos separados por alturas maiores que 6m	61
Figura 4.5	Estrutura do XML utilizando <i>Xlink:href</i> para referenciamento de geometrias.....	63
Figura 4.6	Visão das feições do módulo <i>Building</i> na área de estudo.....	64
Figura 4.7	Edição do contorno da edificação com SkechtUp	65
Figura 4.8	Detalhamento do formato do telhado com SkechtUp.....	65
Figura 4.9	Visão das feições do módulo <i>CityFurniture</i> na área de estudo.....	68
Figura 4.10	Visão das feições do módulo <i>Vegetation</i> na área de estudo	69
Figura 4.11	Visão das feições do módulo <i>WaterBody</i> na área de estudo	71
Figura 4.12	Visão das feições do módulo <i>Transportation</i> na área de estudo.....	73
Figura 4.13	Hierarquia do módulo <i>LandUse</i> para definir o atributo 'classe'	75
Figura 4.14	Amostra de feições do módulo <i>LandUse</i>	75
Figura 4.15	Visão do módulo <i>Relief</i> na área de estudo	76

Figura 4.16 Sumário da validação geométrica utilizando a ferramenta Val3dity no módulo <i>Building</i> em LoD2	78
Figura 4.17 Visualização do modelo gerado utilizando a ferramenta <i>3DCityDB Web-map-client</i>	79

LISTA DE TABELAS

Tabela 2.1 Principais aspectos da integração de dados espaciais.....	18
Tabela 2.2 Previsão de acurácia e generalização para os diferentes níveis de detalhamento.....	23
Tabela 2.3 Comparativo de abordagem, completude, detalhamento e validação dos trabalhos relacionados. Referências: [1] - Sengul (2010), [2] - Agugiaro (2014), [3] - Agugiaro (2016), [4] - Buyukdemircioglu, Kocaman and Isikdag (2018), [5] - Kolbe, Burger and Cantzler (2015), [6] - Soon and Khoo (2017), [7] - Malambo and Hahn (2010), [8] - Janečka (2019), [9] - Goetz and Zipf (2012), [10] - Goetz (2013)	36
Tabela 4.1 Mapeamento dos atributos da base oficial para <i>codelists</i> do módulo <i>Building</i>	55
Tabela 4.2 Número de feições da base oficial que tiveram atributos do OSM associados	59
Tabela 4.3 Comparativo de completude temática nos dados do OSM.....	60
Tabela 4.4 Atributos e filtros de acordo com o nível de detalhamento do módulo <i>Building</i>	61
Tabela 4.5 Mapeamento dos atributos da base oficial para <i>codelists</i> do módulo <i>CityFurniture</i>	67
Tabela 4.6 Atributos e filtros de acordo com o nível de detalhamento do módulo <i>Vegetation</i>	69
Tabela 4.7 Mapeamento dos atributos da base oficial para <i>codelists</i> do módulo <i>WaterBody</i>	70
Tabela 4.8 Mapeamento dos atributos da base oficial para <i>codelists</i> da classe <i>TransportationComplex</i> e <i>TrafficArea</i>	72
Tabela 4.9 Mapeamento dos atributos da base oficial para o atributo <i>SurfaceMaterial</i> .	73
Tabela 4.10 Mapeamento dos atributos de outros módulos e feições adicionais para <i>codelists</i> do módulo <i>LandUse</i>	74

LISTA DE ABREVIATURAS E SIGLAS

ADE	Application Domain Extension
CAD	Computer Aided Desing
ESRI	Environmental Systems Research Institute
FME	Feature Manipulation Engine
GML3	Geography Markup Language 3
KML	Keyhole Markup Language
LiDaR	Light Detection and Ranging
LoD	Level of Detail
MDS	Modelo Digital de Superfície
MDT	Modelo Digital do Terreno
OGC	Open Geospatial Consortium
OSM	OpenStreetMaps
PMPA	Prefeitura Municipal de Porto Alegre
SMMAS	Secretaria de Municipal do Meio Ambiente e da Sustentabilidade
SIG	Sistemas de Informação Geográfica
TIC	TerrainIntersectionCurve
TIN	Triangular Irregular Network
UML	Unified Modeling Language
xAL	eXtensible Address Language
XML	eXtensible Markup Language

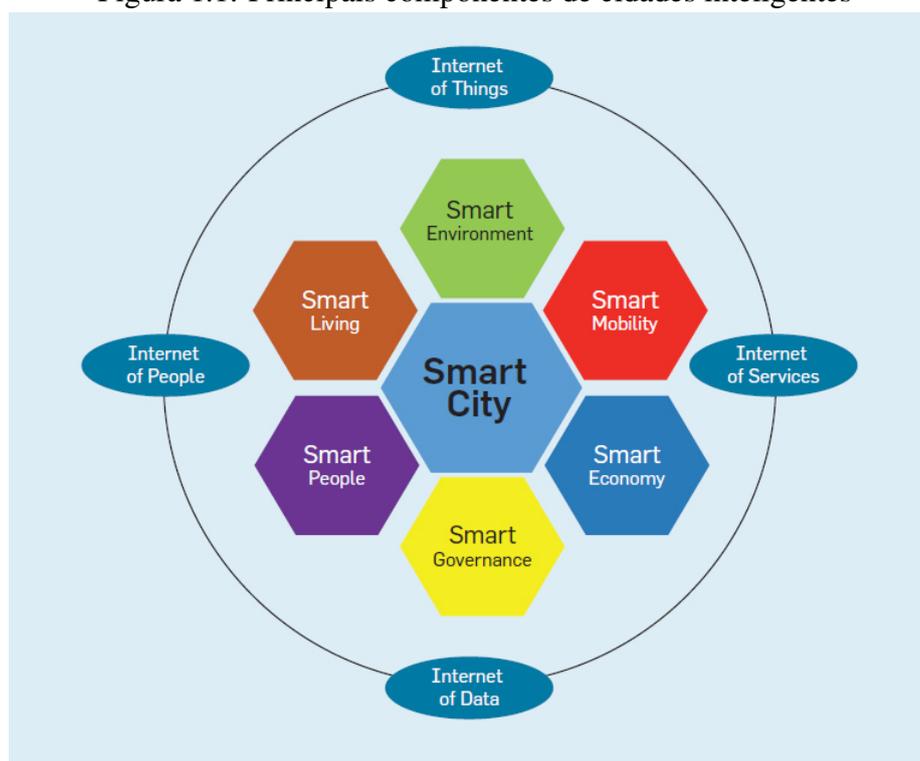
SUMÁRIO

1 INTRODUÇÃO	12
1.1 Objetivos e Contribuições	15
1.2 Organização do Texto	15
2 FUNDAMENTAÇÃO	17
2.1 Integração de dados espaciais	17
2.2 CityGML - Visão Geral	21
2.3 CityGML - Níveis de detalhamento	22
2.4 CityGML - Módulos	24
2.4.1 Módulo <i>Core</i>	25
2.4.2 Módulo <i>Building</i>	25
2.4.3 Módulo <i>CityFurniture</i>	27
2.4.4 Módulo <i>Transportation</i>	27
2.4.5 Módulo <i>Vegetation</i>	29
2.4.6 Módulo <i>WaterBody</i>	29
2.4.7 Módulo <i>LandUse</i>	30
2.4.8 Módulo <i>Relief</i>	31
2.5 Trabalhos Relacionados	32
2.6 Considerações Finais	36
3 ABORDAGEM PARA INTEGRAÇÃO DE DADOS	38
3.1 Fluxograma para o módulo <i>Building</i>	38
3.2 Fluxograma para o módulo <i>CityFurniture</i>	42
3.3 Fluxograma para o módulo <i>Vegetation</i>	43
3.4 Fluxograma para o módulo <i>WaterBody</i>	45
3.5 Fluxograma para o módulo <i>Transportation</i>	46
3.6 Fluxograma para o módulo <i>LandUse</i>	48
3.7 Considerações Finais	50
4 ESTUDO DE CASO	51
4.1 Área de teste	51
4.2 Dados de entrada	52
4.3 Modelagem 3D	54
4.3.1 Módulo <i>Building</i>	54
4.3.2 Módulo <i>CityFurniture</i>	66
4.3.3 Módulo <i>Vegetation</i>	68
4.3.4 Módulo <i>WaterBody</i>	69
4.3.5 Módulo <i>Transportation</i>	71
4.3.6 Módulo <i>LandUse</i>	73
4.3.7 Validação	76
4.4 Considerações Finais	79
5 CONCLUSÃO	81
5.1 Limitações	82
5.2 Trabalhos Futuros	82
REFERÊNCIAS	84

1 INTRODUÇÃO

Em 2018, aproximadamente 55,3% da população mundial vivia em centros urbanos, com projeção do aumento dessa estatística para 60% em 2030 (UN, 2018). Devido a esse crescente aumento populacional nas cidades, seus administradores buscam otimizar os serviços e infraestruturas em termos de escalabilidade, ambiente e segurança para que sejam adaptados à demanda, fazendo com que suas cidades sejam, cada vez mais, inteligentes. Segundo Khatoun and Zeadally (2016), o modelo amplamente adotado para descrever cidades inteligentes foi o proposto pelo Instituto Americano de Padrões e Tecnologia, sendo caracterizado como sistemas complexos consistindo em seis componentes principais: Governo, economia, mobilidade, meio ambiente, qualidade de vida e sociedade. Todos eles envoltos pela conectividade da Internet das coisas, das pessoas, dos serviços e dos dados, conforme visto na Figura 1.1.

Figura 1.1: Principais componentes de cidades inteligentes



Fonte: Khatoun and Zeadally (2016).

Qualidade de vida inteligente envolve conceitos como cultura e bem-estar, saúde e segurança. Já o conceito de meio ambiente inteligente está relacionado com prédios sustentáveis, gerenciamento de recursos e urbanismo sustentável. Mobilidade inteligente é mobilidade multimodal, transportes eficientes e tecnologia e infraestrutura de transporte.

Transparência, acessibilidade e infraestrutura de dados abertos e serviços *on-line*, são conceitos buscados pela governança inteligente. Na economia inteligente são utilizadas conexões comerciais locais e globais para atingir uma melhor produtividade com cultura de inovação e, por fim, a sociedade inteligente tem inclusão, criatividade e educação e cultura como principais conceitos.

Todos os componentes podem ser associados de alguma forma à infraestrutura física ou de dados da cidade. Pode-se dizer que a mobilidade inteligente está relacionada com a infraestrutura dos modais de transporte disponibilizados e a governança inteligente necessita de plataformas e bases de dados interoperáveis para prover serviços *on-line*, também chamados de *e-services* e dados compartilháveis alinhados com padrões de estruturação adotados em nível global. Em 2012, foi realizada uma iniciativa envolvendo grandes empresas privadas como Microsoft e órgãos governamentais como o Governo dos Estados Unidos para adoção global da governança inteligente, principalmente na questão dos dados abertos governamentais, para que os dados disponibilizados pelos governos sejam amigáveis, reutilizáveis e redistribuíveis (KHATOUN; ZEADALLY, 2016).

Dessa forma, é necessário a busca constante por técnicas eficientes que possam auxiliar os administradores desses novos centros modernos na geração, gerência, manutenção e controle da infraestrutura associada a todos os componentes das cidades inteligentes, adotando modelos alinhados com a iniciativa global de dados abertos governamentais. Como ponto de interseção entre a infraestrutura física e de dados, os modelos 3D de cidades vêm tendo um papel cada vez mais importante na rotina diária das pessoas. Em sua revisão, Biljecki et al. (2015) cita diversos estudos científicos relacionados às aplicações de modelos 3D de cidades, entre eles os mais tradicionais como planejamento urbano, cadastro 3D, roteamento e também estudos mais específicos como propagação de ondas de rádio, estimativa de irradiação solar e aplicações militares de comando e controle. Dessa forma, é notável que os modelos 3D devem ser considerados parte essencial da base de dados de cidades inteligentes (PRANDI et al., 2013).

Para algumas aplicações, os dados geométricos e gráficos dos modelos 3D são suficientes para que se alcancem os objetivos previstos, porém, para a maioria delas, os aspectos semânticos são essenciais ou potencializam a capacidade dos modelos. Em estudos relacionados à gestão de desastres e resposta a emergências, por exemplo, ter o conhecimento da função da edificação é necessário, pois estruturas estratégicas e hospitais requerem especial atenção nesse contexto (GRÖGER; PLÜMER, 2012). Além disso, existe a heterogeneidade de dados, onde os usuários finais reúnem dados de diversas fon-

tes para uma determinada aplicação, espaciais e não espaciais, com diferentes resoluções e baseados em diferentes modelos, sendo necessária uma, por vezes custosa, etapa de integração. Sendo, dessa forma, um senso comum de boa prática, a adoção de padrões internacionais de modelagem por parte dos órgãos fornecedores de dados.

Como possível solução de modelo 3D que cubra tanto a questão dos aspectos semânticos como a da heterogeneidade, existe o CityGML¹. Ele é um modelo comum de informação semântica para representação de objetos urbanos em 3D gerenciado desde 2008 pelo OGC (*Open Geospatial Consortium*) (GRÖGER et al., 2012). O CityGML vem sendo adotado mundialmente e diversas cidades, em alinhamento com a iniciativa de dados abertos governamentais, disponibilizam seus modelos nesse padrão para compartilhamento. Alguns desses conjuntos de dados podem ser encontrados em TU Delft (2020). Outro formato existente para descrição geométrica e semântica de cidades é o CityJSON², porém ainda não é reconhecido como padrão comunitário pelo OGC.

Nota-se que a maioria das cidades que detêm uma base de dados com modelo 3D e a disponibiliza estão localizadas na Europa ou na América do Norte, não sendo encontrado nenhum conjunto de dados de cidades localizadas na América do Sul.

Apesar da ausência de dados em 3D, existem cidades que disponibilizam diversos dados geográficos em 2D que podem ser utilizados como insumos para geração de modelos 3D quando agregados com dados de altimetria. Além disso, fontes de outros níveis da administração como estadual e federal, também disponibilizam dados que podem ser úteis nesse contexto. Alguns dos dados que podem ser encontrados de forma aberta são: dados de cadastro em 2D com a descrição geométrica de objetos urbanos, outros dados vetoriais como malha viária e planos diretores contendo informações diversas como estabelecimentos de saúde, estruturas sanitárias e parques, e dados de altimetria como nuvem de pontos.

De uma forma geral, esses dados governamentais são gerados a partir de métodos de produção cartográfica tradicionais e bem estabelecidos, seguindo uma série de critérios e normas no processo produtivo, conseqüentemente sua precisão geométrica atende aos padrões de qualidade estipulados para esse tipo de produto. Por outro lado, via de regra esses dados somente descrevem geometricamente as instâncias, possuindo pouca ou nenhuma informação semântica como nome, endereço, classe e função, conhecimento esse de grande relevância para determinadas aplicações.

¹<<https://www.ogc.org/standards/citygml>>

²<<https://www.cityjson.org/>>

1.1 Objetivos e Contribuições

O **objetivo geral** desta dissertação consiste em desenvolver uma abordagem para integração de dados abertos heterogêneos na geração semiautomática de modelos 3D baseados em CityGML, realizando o "enriquecimento" semântico das instâncias com o relacionamento dos dados oficiais com dados de mapeamento colaborativo aberto. Para atingir tal objetivo, os seguintes **objetivos específicos** foram definidos:

- Desenvolver uma abordagem geral de integração de dados abertos heterogêneos para geração semiautomática de modelos 3D baseados em CityGML;
- Implementar tal abordagem tendo como base dados disponibilizados pela Prefeitura Municipal de Porto Alegre-RS e dados extraídos da base de dados de mapeamento colaborativo aberto;
- Avaliar a abordagem e seu desempenho realizando a validação em diferentes níveis do modelo final gerado.

A expectativa é que tal abordagem possa auxiliar os administradores das cidades que buscam o alinhamento com as diretrizes de cidades inteligentes a gerarem seus modelos semânticos 3D válidos e íntegros com reuso e integração de dados existentes, aproveitando os aspectos positivos de cada base de dados, reduzindo custos nesse processo. Além disso, a abordagem é formalizada na forma de fluxos de trabalho, facilitando sua reimplantação em diferentes contextos.

1.2 Organização do Texto

Este trabalho está estruturado, a contar desta introdução, em cinco capítulos:

- o capítulo 2 apresenta a fundamentação teórica da pesquisa, inicialmente apresentando o panorama geral dos aspectos a serem observados durante o processo de integração de dados espaciais oriundos de bases heterogêneas, seguido de uma visão geral do modelo CityGML, detalhando um pouco mais os módulos a serem implementados no presente trabalho, tendo como principal referência a especificação técnica encontrada em Gröger et al. (2012). Também nesse capítulo são apresentados os trabalhos relacionados ao tema;
- o capítulo 3 apresenta a abordagem geral de forma conceitual, mostrando os fluxos

de trabalhos específicos para implementação dos diferentes módulos do CityGML.

- o capítulo 4 entra nos detalhes da implementação do estudo de caso, especificando a área de estudo, os dados de entrada, etapas da modelagem 3D e o "enriquecimento" semântico com dados do OSM. Neste capítulo também são apresentados os resultados obtidos dentro de cada módulo, incluindo a validação em diferentes níveis;
- por fim, o capítulo 5 apresenta as conclusões, contribuições, limitações e trabalhos futuros resultantes desta pesquisa.

2 FUNDAMENTAÇÃO

Este capítulo apresenta a base conceitual utilizada neste trabalho, sendo separado em seis seções. A seção 2.1 traz um panorama geral dos aspectos a serem observados durante o processo de integração de dados espaciais oriundos de bases heterogêneas. As seções seguintes abordam as especificações técnicas do CityGML, sendo que a seção 2.2 descreve as características gerais do modelo, a seção 2.3 apresenta os diferentes níveis de detalhamento previstos, na seção 2.4, são apresentadas as especificações técnicas dos 8 módulos do CityGML implementados na pesquisa e a seção 2.5, apresenta os trabalhos relacionados considerados mais importantes, fazendo um comparativo da abrangência e completude de cada estudo. A última seção apresenta as considerações finais sobre o capítulo.

2.1 Integração de dados espaciais

Integração de dados consiste na combinação de conjuntos de dados obtidos de diferentes fontes e fornecimento de uma versão unificada dos mesmos. É uma questão antiga e pervasiva, sendo encontrada em diversos setores tanto científicos como governamentais (LENZERINI, 2001).

Por se tratar de um problema de grande amplitude, nessa seção são abordadas apenas questões relativas aos dados espaciais.

Existem diversos estudos na literatura que tratam da integração de dados espaciais. Alguns têm uma visão objetiva, tratando de casos específicos como User, Finn and Starbuck (2009), que define integração espacial como a correspondência geométrica, topológica e de atributos entre feições. Outros abordam a temática de forma mais ampla como Mohammadi et al. (2006), que lida com a questão no contexto de implementação de infraestruturas de dados espaciais e cita os cinco principais aspectos a serem considerados na integração: técnico, legal, institucional, político e social.

Na Tabela 2.1 é mostrado um sumário de todas as questões atinentes à integração de dados espaciais.

Sobre os aspectos técnicos, heterogeneidade computacional remete à adoção de diferentes padrões de dados pelos gerentes de bases distintas, dificultando a interoperabilidade e compartilhamento. Já topologia trata do relacionamento espacial entre as instâncias, incluindo associação por adjacência, conectividade e contenção (THEOBALD,

Tabela 2.1: Principais aspectos da integração de dados espaciais

Técnicos	Institucionais	Políticos	Legais	Sociais
<ul style="list-style-type: none"> Heterogeneidade computacional (padrões e interoperabilidade) Topologia 	<ul style="list-style-type: none"> Modelos de colaboração Modelo de financiamento 	<ul style="list-style-type: none"> Questões de legislação Prioridades (desenvolvimento sustentável) e objetivos Precificação 	<ul style="list-style-type: none"> Direitos, restrições e responsabilidades Direitos de propriedade e direitos de propriedade intelectual Acesso e privacidade de dados Licenciamento 	<ul style="list-style-type: none"> Questões culturais Desenvolvimento de capacidade Fundamentos das partes interessadas
<ul style="list-style-type: none"> Semântica Sistema de referência e escala Qualidade de dados Modelo de dados Metadados Formato 	<ul style="list-style-type: none"> Ligação entre unidades de gerenciamento de dados Consciência da existência de dados 			

Fonte: Adaptado de Mohammadi et al. (2006).

2001). A topologia faz parte de um dos elementos que descrevem a qualidade de dados espaciais, a consistência lógica, que é definida como o grau de aderência das feições às regras lógicas de estrutura de dados (ISO, 2013 apud FONTE et al., 2017, p. 146). Além da consistência lógica, a norma prevê outros 5 elementos da qualidade de dados espaciais abaixo descritos:

- **Acurácia posicional:** Comparação das coordenadas geográficas de uma feição com as coordenadas geográficas do objeto por ela representado no mundo real dentro de um mesmo sistema de referência;
- **Acurácia temática:** Precisão das classificações e temas associados a localizações ou objetos específicos, ou seja, a classe de um *pixel* em uma imagem de cobertura terrestre ou rótulo de uma feição vetorial;
- **Completude:** Presença ou ausência de feições, dos seus atributos ou de suas relações tendo como referência as especificações do produto final;
- **Qualidade temporal:** Qualidade dos atributos temporais como data de coleta, data de publicação, frequência de atualização ou validade temporal;
- **Usabilidade:** Alinhamento dos dados com as necessidades e requisitos do usuário final.

A semântica, nesse contexto, lida com o significado ou definição formal das entidades que compõem as ontologias representativas de conceitos do mundo real ou, mais especificamente, do mundo geográfico. Uma descrição mais detalhada das inconsistências semânticas geradas por granularidade pode ser encontrada em Fonseca et al. (2002).

Assim como a topologia, os sistemas de referência em conjunto com a escala estão

diretamente associados a um elemento da qualidade de dados, a acurácia posicional. Sistemas de referência são um conjunto de regras matemáticas que indicam como associar valores de coordenadas às feições espaciais (LOTT, 2010). Existem transformações matemáticas capazes de converter as coordenadas de um sistema para o outro visando passar as feições para um sistema homogêneo porém, tais transformações podem inserir erros nas coordenadas (MONTECINO; SANTIBANEZ; FREITAS, 2013) (ESRI, 2020), afetando a acurácia posicional. Já a escala, no contexto de dados espaciais, é a relação de proporção das dimensões da feição no mundo real e a sua representação gráfica. Mapas em grande escala cobrem uma área menor, mas geralmente com mais detalhes, e um mapa em pequena escala cobre uma área maior com menos detalhes (LAM; QUATTROCHI, 1992).

Por modelo de dados entende-se uma abstração da realidade que incorpora somente os atributos relevantes das feições para uma determinada aplicação. Geralmente os modelos de dados são formalizados e documentados através de uma estruturação gráfica de diagramas e listas visando sua posterior implementação (KUFONIYI, 1995).

Outro aspecto técnico relacionado à integração de dados espaciais é o metadado. Segundo BRASIL (2008) metadados são:

"Conjuntos de informações descritivas sobre os dados, incluindo as características do seu levantamento, produção, qualidade e estrutura de armazenamento, essenciais para promover a sua documentação, integração e disponibilização, bem como possibilitar a sua busca e exploração"

Dessa forma, os metadados permitem que um produtor descreva totalmente um conjunto de dados para que os usuários possam entender as possibilidades e limitações e avaliar a sua aplicabilidade para determinado fim (KIM, 1999).

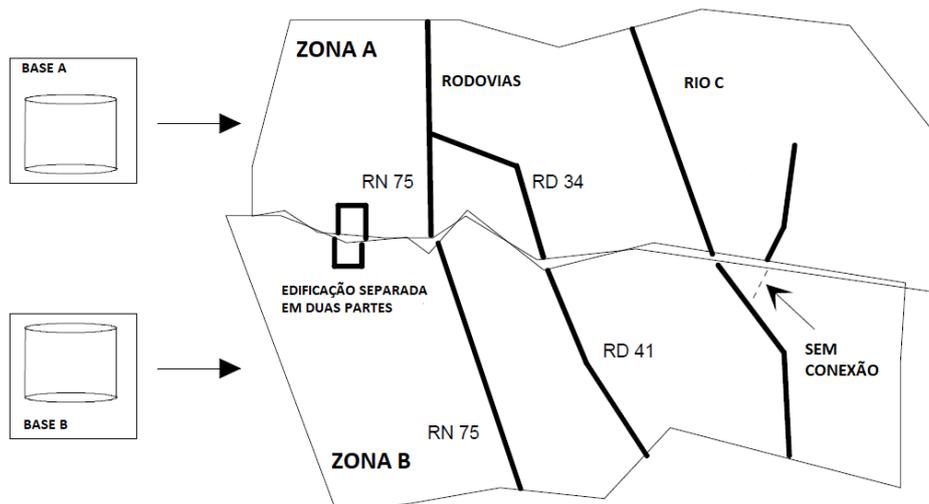
Por fim, formato está diretamente ligado ao primeiro aspecto técnico, heterogeneidade computacional. Existem uma variedade de formatos de dados espaciais. Eles se dividem em duas grandes classes: vetoriais e matriciais, também chamados de *raster*. O formato *ShapeFile*, pertencente ao *Environmental Systems Research Institute* - ESRI, é um formato abertamente documentado e considerado um formato estável para compartilhamento de dados espaciais vetoriais. DWG e DFX são formatos proprietários da empresa Autodesk, também abertamente documentados mas com restrições de informações semânticas Cote (2020). Atualmente o OGC adota como formato padrão o *GeoPackage*, um formato aberto, baseado em padrões, independente de plataforma, portátil, autoexplicativo e compacto para transferência de informações geoespaciais (OGC, 2020). Da mesma forma, existem diversos formatos de dados *raster* utilizados para representa-

ção de informações geográficas, sendo alguns deles GIF, PNG, TIFF, GeoTIFF e JPEG, tendo como principais diferenças a capacidade de representação de cores e compressão de dados. O formato GeoTIFF possui georreferenciamento interno nativo porém, todos os outros podem ser georreferenciados utilizando plataformas específicas (COTE, 2020).

Posto que o escopo desta pesquisa engloba somente dados abertos, será dada ênfase na parte técnica da integração de dados, uma vez que os outros aspectos não influenciam de maneira significativa nesse contexto.

Pôde-se verificar a existência de diversos cenários de heterogeneidade de dados encontrados em *datasets* do mundo real, uma vez que existem um grande número de variáveis que podem interferir no processo de integração de dados espaciais. A Figura 2.1 mostra um exemplo simples de aspectos técnicos a serem considerados na integração de duas bases distintas.

Figura 2.1: Exemplos de inconsistências entre bases heterogêneas



Fonte: Adaptado de Laurini (1998).

Verifica-se a falta de conectividade geométrica entre feições, no caso da edificação, feições com falta de acurácia temática, ou seja, atributos da mesma feição do mundo real divergindo nas duas bases, no caso da rodovia e ausência de feições, no caso da porção do afluente do rio C. Dessa forma os aspectos relacionados à qualidade de dados divergem nas duas bases, necessitando de homogeneização para serem integradas.

2.2 CityGML - Visão Geral

O CityGML foi projetado como um modelo de dados abertos baseados em XML (*eXtensible Markup Language*), sendo um esquema de aplicação da GML3 (*Geography Markup Language 3*), norma internacional extensível para compartilhamento e codificação de dados espaciais emitida pelo OGC e pela ISO TC 211 (GRÖGER et al., 2012), possuindo uma estrutura adequada para apresentar modelos de cidades com informações semânticas. Esses recursos semânticos permitem que os usuários executem funções que não são possíveis sem os metadados fornecidos pelo CityGML. Os módulos no CityGML refletem a aparência, as características espaciais e de tema de um objeto (BUYUKDEMIRCIUGLU; KOCAMAN; ISIKDAG, 2018). Com o modelo foram alcançadas definições em comum dos atributos e relações entre entidades básicas de um modelo 3D de cidade, levando em consideração aspectos semânticos e geométricos/topológicos. Dessa forma, o relacionamento entre as instâncias pode ser derivado tanto do nível semântico, através dos atributos e relações de hierarquia e agregação, como do nível geométrico/topológico através do seu relacionamento espacial. Deve haver coerência entre os dois níveis de informação para que o modelo seja válido e íntegro (STADLER et al., 2009).

Uma das implementações para manter a integridade semântica é a inclusão de listas, chamadas *codelists*, com valores discretos para preenchimento dos atributos das feições, evitando erros comuns como digitação incorreta ou criação de valores fora do domínio do atributo. Tais listas são do tipo *gml:CodeType* e são especificadas em um esquema externo ao CityGML, sendo somente referenciadas no modelo. Como forma de atender a situações particulares de modelagem que possam ocorrer de acordo com o contexto da região mapeada, as listas geralmente não são rígidas, podendo ser adaptadas, caso necessário, devendo haver uma entidade responsável pela governança das listas adaptadas. As listas presentes nas especificações do modelo são sugestões, e foram utilizadas no escopo da pesquisa.

Considerando que os modelos 3D são geralmente oriundos de outras bases de dados contendo dados em 2D, outra forma de ser manter a integridade do modelo é a manutenção de uma relação com essas bases de dados herdadas, através de um conceito chamado de *ExternalReference*, onde cada elemento instanciado no modelo 3D pode ter um *link* com o identificador único do elemento da base de dados herdada que o originou.

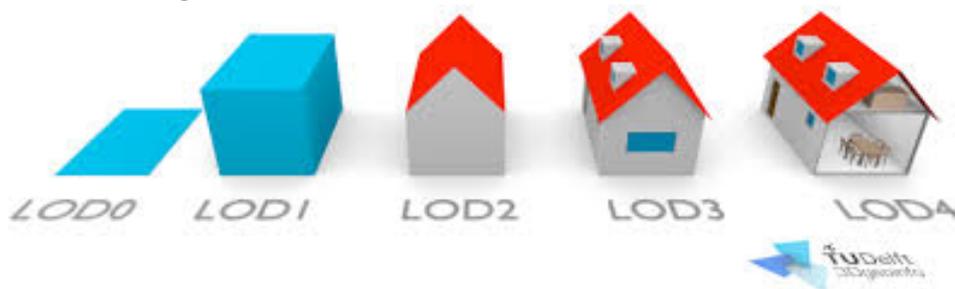
Mantendo a conformidade com padrões internacionais e aplicando as diretrizes de reúso, o CityGML adotou a *eXtensible Address Language* (xAL) (OASIS, 2002), como

linguagem padrão para agregação de endereços às edificações. Além disso, CityGML também é extensível, possibilitando o enriquecimento de dados com propriedades específicas de determinados domínios. Tais extensões são chamadas de *Application Domain Extension* (ADE) e mantém a interoperabilidade semântica com o restante do modelo.

2.3 CityGML - Níveis de detalhamento

O CityGML possibilita a modelagem das instâncias em cinco diferentes níveis de detalhamento (*Level of Detail* - LoD). Os níveis de detalhamento proporcionam a adequação da quantidade e refinamento de informações disponibilizadas com os distintos níveis de exigência de cada aplicação. A Figura 2.2 mostra a diferença entre os níveis de detalhamento do módulo *Building*.

Figura 2.2: Diferentes níveis de detalhamento (LoD)



Fonte: Stoter et al. (2016).

O nível mais simples (LoD0) representa a edificação somente como polígono em 2.5D com o contorno da edificação no nível do solo ou no nível do telhado, sendo destinado para projetos de mapeamento de grandes regiões, o segundo nível (LoD1) representa como um sólido simples, sem o detalhamento do formato do telhado e se destina ao mapeamento de cidades ou pequenas regiões, o terceiro nível (LoD2) já apresenta detalhes como formato do telhado, devendo o sólido ser decomposto em superfícies que o compõem (telhado, paredes e solo), sendo destinados ao mapeamento de bairros ou distritos de uma cidade. O grande diferencial entre o LoD1 e o LoD2 no módulo *Building* é a possibilidade de evitar redundâncias na implementação de geometrias. Isso é feito realizando o referenciamento das geometrias que compõem um edifício, ao invés de declarar a geometria da edificação explicitamente. Assim, uma parede que é compartilhada por duas edificações vizinhas é declarada somente uma vez no arquivo XML, e referenciada pelas duas edificações utilizando o conceito de *Xlink* do GML 3.1.1. Caso haja neces-

Tabela 2.2: Previsão de acurácia e generalização para os diferentes níveis de detalhamento

	<i>LOD0</i>	<i>LOD1</i>	<i>LOD2</i>	<i>LOD3</i>	<i>LOD4</i>
Descrição da escala do modelo	regional, paisagística	cidade, pequena região	idades, bairros, grandes projetos	bairros, modelos de arquitetura externos, pontos de referência	modelos de arquitetura internos, pontos de referência
Classe de acurácia	menor	baixo	médio	alto	muito alto
Acurácia absoluta de pontos 3D (posição/altura)	menor que LOD1	5/5m	2/2m	0.5/0.5m	0.2/0.2m
Generalização	generalização máxima	blocos de objetos como feições generalizadas; >6*6/3m	objetos como feições generalizadas; >4*4/1m	objetos como feições reais; 2*2/1m	elementos construtivos e aberturas são representadas
Representação da estrutura do telhado	não	não	sim	feições externas representativas	formato original
Módulo CityFurniture	não	objetos importantes	protótipos, objetos generalizados	objetos em formato real	objetos em formato original
Classe SolitaryVegetation	não	objetos importantes	protótipos maiores que 6m	protótipos maiores que 2m	protótipos no formato original
Classe PlantCover	não	>50*50m	>5*5m	<LOD2	<LOD2

Fonte: Adaptado de Gröger et al. (2012).

sidade de edição de edificações, devido, por exemplo, a uma demolição, a referência às geometrias serão apagadas, mas as geometrias em si podem permanecer caso existam outras instâncias de edifícios referenciando-as. O quarto nível (LoD3) descreve com mais detalhamento o contorno externo das edificações, podendo conter as informações de localização das entradas e janelas, sendo destinado ao mapeamento de pontos de referência ou um conjunto pequeno de edificações e, por fim, o quinto nível (LoD4) descreve, também, a estrutura interna das edificações, contendo as separações dos cômodos, escadas, passagens interiores e a mobília interna, sendo utilizado para mapeamento de edificações específicas.

Para cada módulo existe uma descrição das informações disponibilizadas em cada nível de detalhamento. Buscando também uma normatização de generalização para o modelo, foi proposta uma tabela contendo os dados de acurácia, generalização e outras descrições necessárias para adequação dos dados a cada um dos níveis de detalhamento, conforme Tabela 2.2.

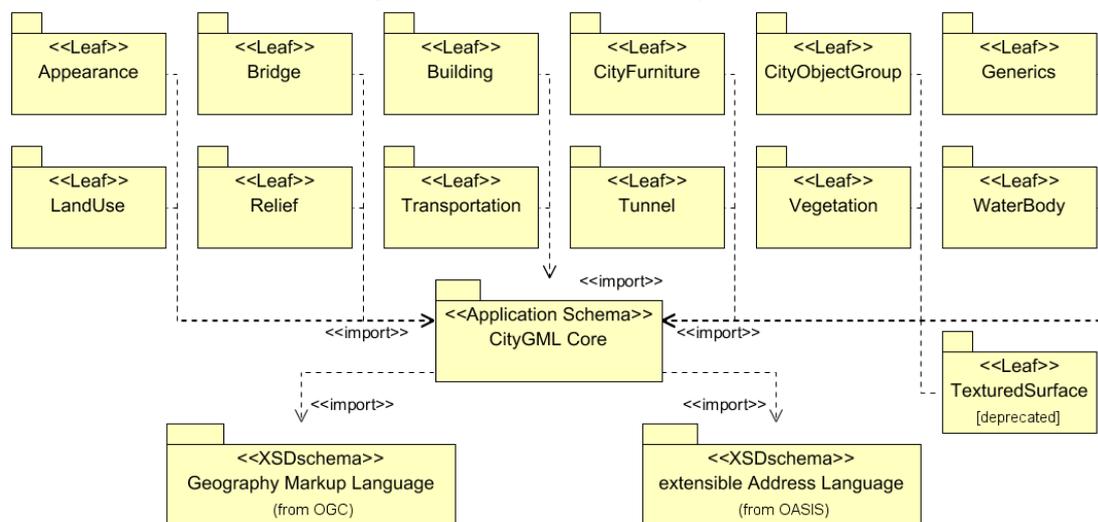
Como se pode ver a tabela é uma proposta, com estudos de mudança ainda em andamento para consolidação em versão posterior do CityGML.

O CityGML também possibilita a representação de uma mesma instância em diferentes níveis de detalhamento em um mesmo documento, proporcionando a análise e visualização de um mesmo objeto em diferentes refinamentos simultaneamente.

2.4 CityGML - Módulos

Outra característica do CityGML é a sua separação em módulos. Atualmente, conforme Figura 2.3, o modelo é decomposto em um módulo central chamado *Core*, que abrange os conceitos e componentes gerais de qualquer modelo de dado, e outros 13 módulos temáticos, destinados à modelagem de classes específicas de instâncias, sendo eles: *Appearance*, *Bridge*, *Building*, *CityFurniture*, *CityObjectGroup*, *Generics*, *LandUse*, *Relief*, *Transportation*, *Tunnel*, *Vegetation*, *WaterBody*, e *TexturedSurface [deprecated]*.

Figura 2.3: Módulos do CityGML



Fonte:Gröger et al. (2012).

Uma das vantagens dessa modularização é a possibilidade de implementação parcial do modelo com qualquer combinação de módulos temáticos em associação com o módulo *Core* para atender a aplicações específicas. Por exemplo, se o modelo a ser criado é destinado para estudo de roteamento, módulos como *LandUse*, *Vegetation* e *WaterBody* podem não ser necessários.

A modelagem formal do CityGML, apresentando as relações de associação, cardinalidade, agregação e herança, é feita utilizando os diagramas da linguagem *Unified Modeling Language (UML)*. Para uma descrição mais detalhada da linguagem verificar Booch, Rumbaugh and Jacobson (1999).

Além disso, todos os módulos são especificados individualmente em esquemas em XML, com *namespaces* globais únicos.

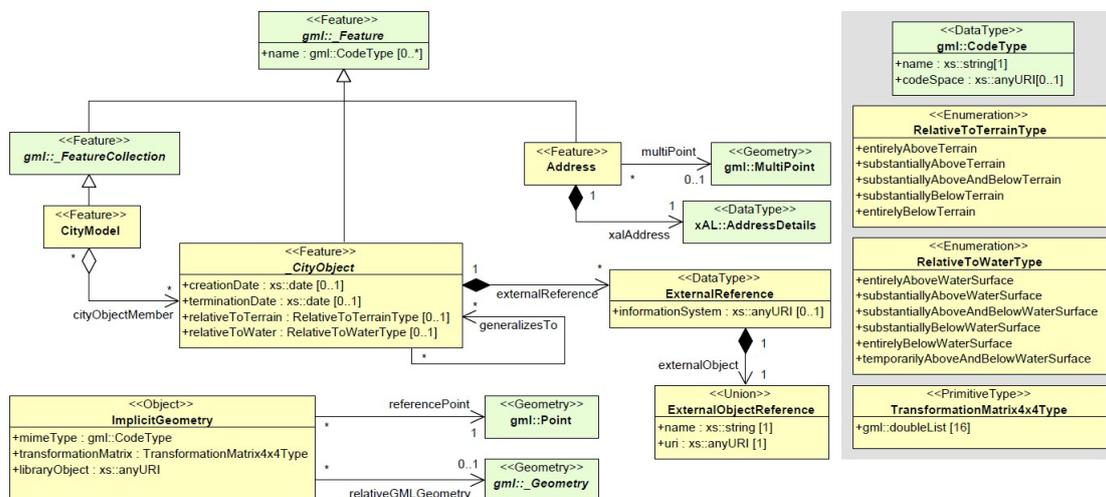
A seguir serão apresentadas as características específicas dos módulos presentes no estudo de caso e incluídos no escopo da presente pesquisa. Grande parte das informa-

ções deste capítulo foram extraídas de Gröger et al. (2012) e Gröger and Plümer (2012).

2.4.1 Módulo *Core*

O módulo *Core* provê as definições abstratas de base para todos os outros módulos presentes no CityGML, devendo ser necessariamente implementado em qualquer modelo. Na Figura 2.4 pode-se verificar a modelagem em UML do módulo.

Figura 2.4: Modelagem em UML do módulo *Core*

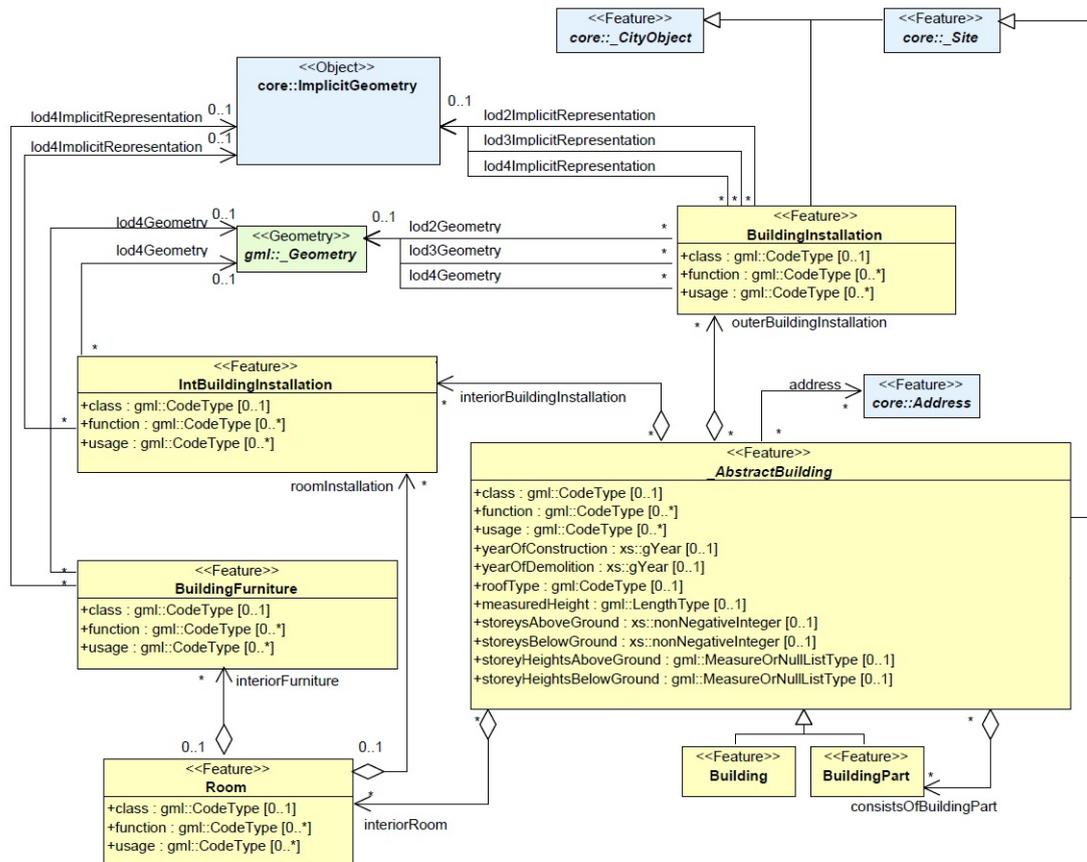


Fonte: Adaptado de Gröger et al. (2012).

Classes hierarquicamente abaixo da classe *_CityObject* herdarão 4 atributos: *creationDate*, *terminationDate*, *relativeToTerrain* e *relativeToWater*, além de poderem possuir uma referência externa, tendo nenhum ou no máximo 1 valor possível. Os atributos *creationDate* e *terminationDate* determinam as datas de criação e desuso, respectivamente, das instâncias no modelo. Os possíveis valores discretizados dos atributos *relativeToTerrain* e *relativeToWater* também são especificados na modelagem.

2.4.2 Módulo *Building*

É o mais importante dos módulos, uma vez que as edificações são as principais instâncias e objetos de pesquisa em um ambiente urbano. A principal classe do módulo é a *AbstractBuilding*, que é herdeira, por transitividade da classe *_CityObject*. A Figura 2.5 mostra a modelagem do módulo *Building* em UML.

Figura 2.5: Modelagem em UML do módulo *Building*

Fonte: Adaptado de Gröger et al. (2012).

Os atributos estipulados para a classe *AbstractBuilding* podem ser verificados na modelagem, bem como a definição do tipo de dados de entrada. Verifica-se que os atributos *class*, *function*, *usage* e *roofType* são do tipo *CodeType*, visando manter a integridade semântica dos dados. Outros atributos que o usuário julgar necessário para melhor descrever as entidades podem ser modelados na forma de *generic attributes*, presente no módulo *Generics*. A classe *AbstractBuilding* se especializa em *Building* e *BuildingPart*. Dado que também existe a relação de agregação entre a classe *AbstractBuilding* e a classe *BuildingPart*, a modelagem permite uma hierarquização em profundidade. Comumente encontrados em um ambiente urbano, os complexos de prédios como fábricas, escolas e hospitais, são modelados em separado, no módulo *CityObjectGroups*.

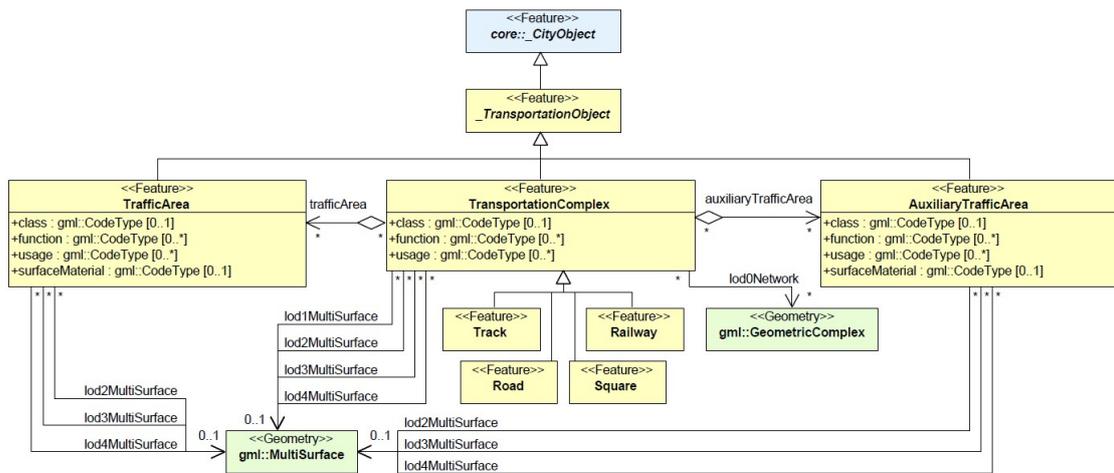
2.4.3 Módulo *CityFurniture*

O módulo *CityFurniture* é composto por elementos como postes, sinais de trânsito, bancos, paradas de ônibus, entre outros objetos fixos que fazem parte de um ambiente urbano. Existem duas possíveis formas de modelagem desses objetos urbanos, uma é através da geometria explícita, ou seja, a geometria de cada elemento é armazenada individualmente no modelo, a outra é através da geometria implícita, onde a geometria de objetos similares é armazenada uma única vez em um sistema de referência local, sendo referenciada pelas instâncias propriamente ditas. A segunda alternativa é eficaz no sentido de representação visual pois, em uma mesma cidade, feições como postes, bancos e sinais de trânsito possuem geometrias idênticas, modificando somente sua orientação e localização no espaço, além disso, esse método possibilita a otimização do modelo final em termos de espaço. Alguns aspectos negativos apontados por Soon and Khoo (2017) podem ocorrer como não reconhecimento de geometrias implícitas por algumas plataformas de visualização de modelos 3D e quebra do *link* de referenciamento entre as instâncias e as geometrias.

2.4.4 Módulo *Transportation*

O módulo *Transportation* inclui as feições do sistema de transporte de uma cidade. Além de incluir as vias terrestres para veículos, o modelo também inclui outras feições como via de pedestres, ciclovias, vias aquáticas, heliporto, entre outras relacionadas a transportes em geral. O diagrama em UML apresentado na Figura 2.6 mostra a classe principal *TransportationComplex* sendo composta por agregação por duas classes, *TrafficArea* e *AuxiliaryTrafficArea* e se especializa em 4 subclasses: *Track*, *Road*, *Railway* e *Square*. *TrafficAreas* são aquelas feições essenciais para caracterização de determinado espaço como área de transporte. Alguns exemplos são faixas de rodagem, calçadas e ciclovias. Já as *AuxiliaryTrafficAreas* fazem parte da feição de transporte, mas não são essenciais para essa caracterização como canteiros divisórios, pequenos jardins, entre outras.

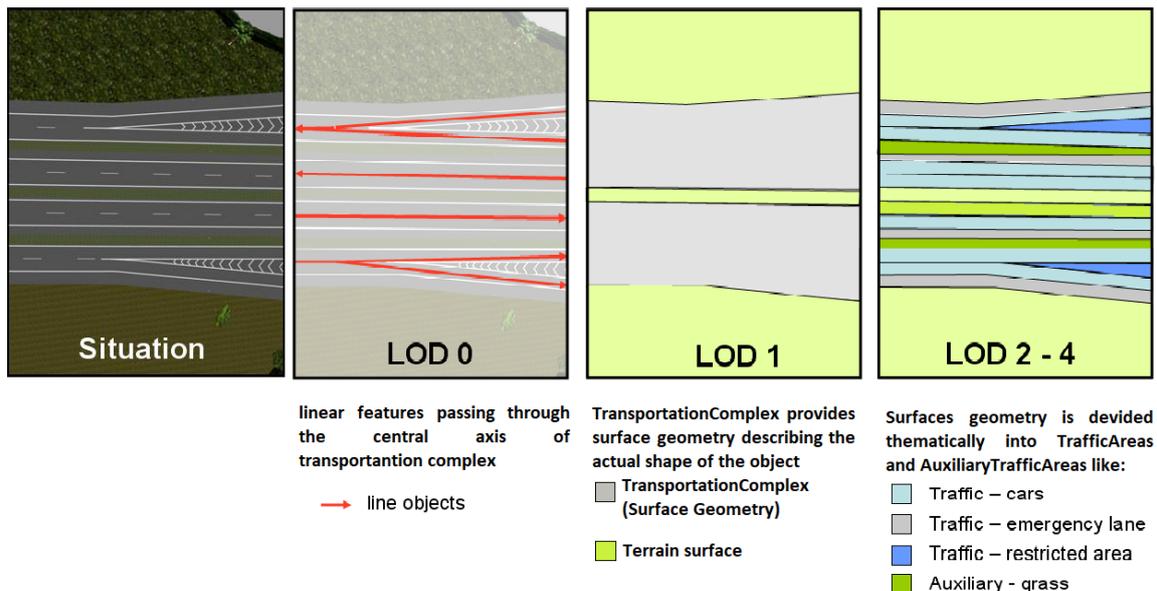
Figura 2.6: Modelagem em UML do módulo *Transportation*



Fonte: Adaptado de Gröger et al. (2012).

Do ponto de vista geométrico/topológico, as feições podem ser representadas em LoD0 por uma rede linear descrevendo o eixo central da malha viária ou por *MultiSurfaces* nos demais níveis de detalhamento, sendo os polígonos que representam a área ocupada pela feição. A Figura 2.7 mostra a diferença de representação das feições nos LoD previstos.

Figura 2.7: Diferença de representação do módulo *Transportation* de acordo com o nível de detalhamento

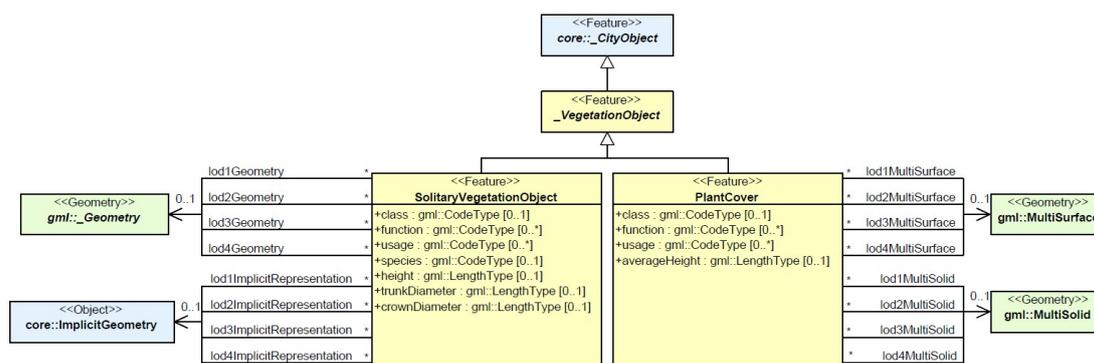


Fonte: *Rheinmetall Defence Electronics*, extraída de Gröger et al. (2012).

2.4.5 Módulo *Vegetation*

A modelagem das feições de vegetação prevê a separação em duas grandes classes, *SolitaryVegetationObject* e *PlantCover*. A primeira classe modela árvores isoladas que se destacam e podem ser facilmente identificadas visualmente no ambiente urbano. Já na segunda se encontram áreas ocupadas por um determinado tipo de vegetação, não sendo possível fazer a distinção entre feições individuais. A Figura 2.8 mostra o diagrama em UML definindo essa distinção além de prever, assim como no módulo *CityFurniture*, a utilização da geometria implícita para instâncias das classes *SolitaryVegetationObject*, uma vez que árvores da mesma espécie terão formato e dimensões similares, modificando somente sua localização no espaço geográfico.

Figura 2.8: Modelagem em UML do módulo *Vegetation*



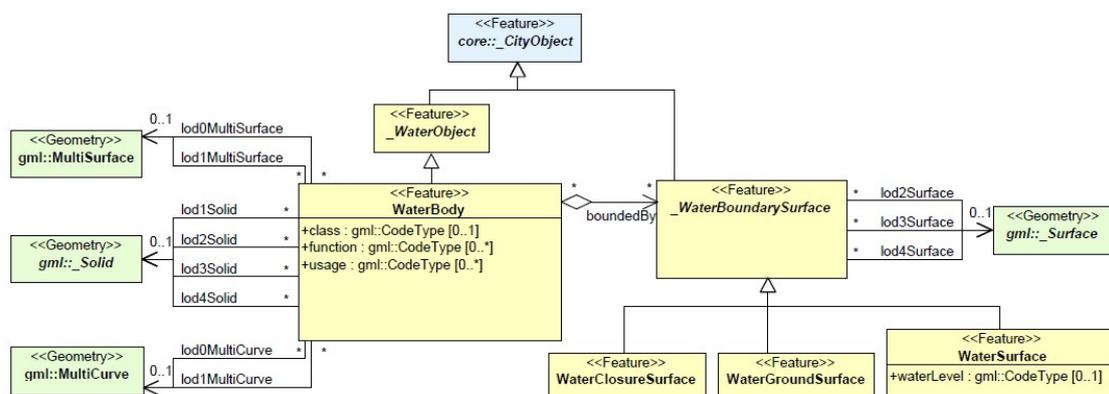
Fonte: Adaptado de Gröger et al. (2012).

2.4.6 Módulo *WaterBody*

O módulo *WaterBody* inclui feições que desempenham papel importante em aplicações urbanísticas como rios, lagos, fontes e reservatórios. O caráter dinâmico das feições, por sofrerem influência de eventos sazonais, as tornam ainda mais sensível para simulações de desastres naturais como enchentes (GRÖGER et al., 2012). Apesar de algumas feições terem função de transporte, não é previsto neste módulo a modelagem das feições na forma de rede linear conectada, ficando essa abordagem prevista somente no módulo *Transportation*. Como o módulo não foi incluído na Tabela 2.2, não há uma definição clara das feições que devem ser incluídas em cada nível de detalhamento de acordo com suas dimensões. Apenas algumas menções de generalização são feitas. Em LoD0, rios e lagos com extensões significantes são modelados como superfícies (*gml:MultiSurface*) e

as demais feições com geometria do tipo linha (*gml:MultiCurve*). Níveis de detalhamento LoD1 os corpos d'água podem ser representados como sólidos envoltos por superfícies topologicamente conectadas, chamadas *WaterSurfaces*, *WaterClosureSurfaces* e *WaterGroundSurfaces*. Essa implementação é realizada caso o relevo seja obtido também por levantamento batimétrico, onde são feitas medições do relevo abaixo do nível da água. Uma abordagem prevista no modelo para lidar com o dinamismo das feições é a inclusão do atributo *WaterLevel*, consistindo em uma *codeList* descrevendo diversos níveis de altura que um corpo d'água possa atingir. A Figura 2.9 mostra a modelagem em UML para o módulo *WaterBody*.

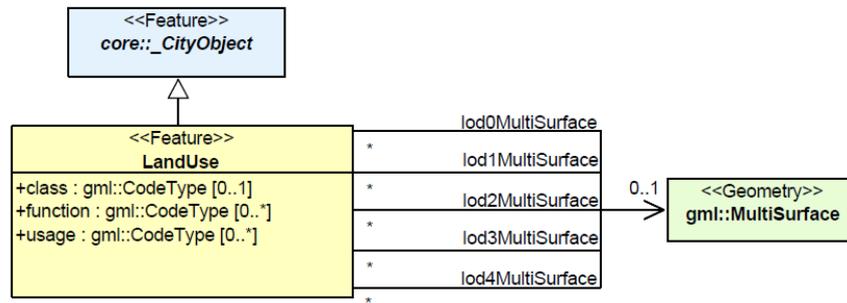
Figura 2.9: Modelagem em UML do módulo *WaterBody*



Fonte: Adaptado de Gröger et al. (2012).

2.4.7 Módulo *LandUse*

O módulo *LandUse* têm a finalidade de classificar e descrever tanto a cobertura quanto o uso do solo. A classificação da cobertura do solo remete a sua aparência física natural, como areia, terreno exposto, gramado, etc. Já no uso do solo é especificada a finalidade do solo para uso humano como, parque, área industrial, residencial, etc. A Figura 2.10 mostra a modelagem em UML para o módulo *LandUse*.

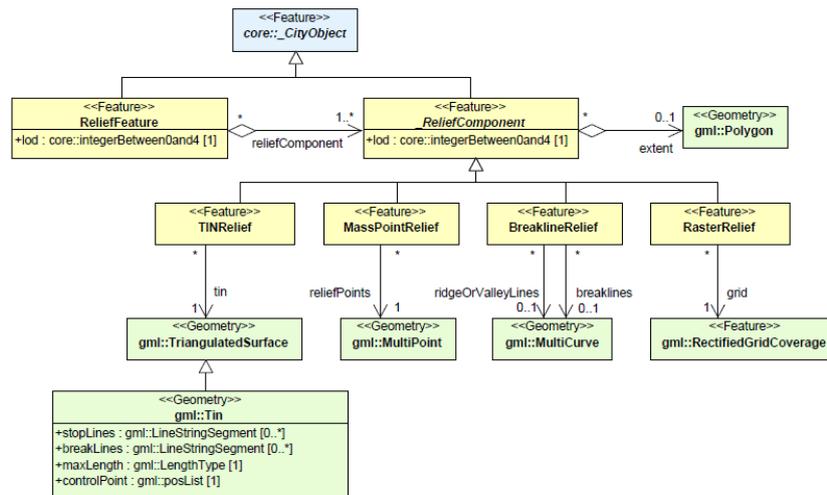
Figura 2.10: Modelagem em UML do módulo *LandUse*

Fonte: Adaptado de Gröger et al. (2012).

No atributo *citygml_class*, o solo é classificado em cinco categorias gerais: Área edificada, Área não desenvolvida, Tráfego, Vegetação e Corpos D'água, tendo uma única ocorrência. Já nos atributos *citygml_function* e *citygml_usage*, são especificados o uso e a cobertura do solo, podendo receber múltiplos valores, possibilitando, por exemplo, que uma porção do solo seja classificada como parque (uso) e gramado (cobertura) ao mesmo tempo.

2.4.8 Módulo *Relief*

O modelo digital do terreno (MDT) representa graficamente o contorno da superfície terrestre Yan et al. (2019). No CityGML o módulo dedicado à especificação do MDT é o módulo *Relief*, prevendo quatro alternativas geométricas de modelagem do terreno, conforme diagrama UML da Figura 2.11.

Figura 2.11: Modelagem em UML do módulo *Relief*

Fonte: Adaptado de Gröger et al. (2012).

No modelo *TINRelief*, o terreno é representado por uma rede triangular irregular (*Triangular Irregular Network* - TIN), onde os vértices são compostos por pontos em 3D e a triangulação é reconstruída por métodos de tesselação como triangulação de *Delau-nay*. O modelo *MassPointRelief* consiste basicamente por um conjunto de pontos em 3D. Já no modelo *BreakLineRelief* o terreno é representado por curvas em 3D que acompanham o seu contorno, chamadas isolinhas. Por fim, no modelo *RasterRelief*, o terreno é representado graficamente por uma malha regular, onde cada célula armazena o valor de altimetria do espaço geográfico por ela representado. Para garantir a integração com outros módulos, são previstas as chamadas *TerrainIntersectionCurves* (TIC). São curvas contendo a exata posição onde as feições tocam o terreno, garantindo, dessa forma, a consistência topológica entre os módulos. As TIC são previstas explicitamente nos módulos *Bridge*, *Building*, *CityFurniture* e *Tunnel*. No módulo *WaterBody* essa integração é feita de modo implícito, através das *WaterGroundSurfaces*. Não há previsão de TIC nos módulos *Transportation* e *Vegetation*.

2.5 Trabalhos Relacionados

A geração de modelos 3D tem diferentes abordagens, podendo ser manual, semi-automática ou automática, sendo escolhidas de acordo com os insumos e suporte técnico disponível. Métodos manuais envolvem a geração sistemática dos objetos urbanos em 3D a partir do contorno em 2D utilizando *Computer Aided Desing* (CAD) ou Sistemas de In-

formação Geográfica (SIG). Técnicas semiautomáticas e automáticas utilizam imagens de satélite ou ortofotos para extrusão das edificações através de métodos fotogramétricos ou utilizam dados de altimetria, como nuvem de pontos para identificar os modelos digitais do terreno e das edificações (MALAMBO; HAHN, 2010).

Como a abordagem manual é altamente custosa, especialmente em grandes centros, demandando recursos financeiros, humanos e tempo, a comunidade acadêmica se voltou para o desenvolvimento de pesquisas direcionadas ao avanço de técnicas semiautomáticas e automáticas. Uma revisão de metodologias para geração de modelos essencialmente geométricos em 3D pode ser encontrada em Moreia et al. (2013).

Outra categorização que pode ser aplicada em trabalhos nesse segmento é a de completude de módulos do CityGML implementados. Grande parte dos estudos são voltados especificamente à implementação do módulo *Building*, de fato o módulo mais importante na modelagem de cidades inteligentes, enquanto apenas alguns trabalhos são mais abrangentes, englobando os demais módulos. Além disso, a questão da validação do modelo gerado também não é mencionada em alguns estudos.

Sengul (2010) utilizou métodos fotogramétricos para identificar a geometria dos edifícios do centro histórico de Istambul. Algumas das etapas realizadas neste processo foram feitas de forma manual, como a obtenção do contorno e formato dos telhados das edificações. A implementação do módulo *Building* do CityGML foi feita com a integração das geometrias obtidas no processo fotogramétrico com outras duas bases de dados oficiais através do software FME (*Feature Manipulation Engine*).

Em Agugiaro (2014) e Agugiaro (2016) foram realizados estudos de caso de implementação de modelos baseados em CityGML nas cidades de Trento-Itália e Viena-Áustria, respectivamente, dando ênfase nas etapas de pré-processamento necessárias para garantir a integridade dos dados de entrada, uma vez que os dados existentes no mundo real, via de regra, são heterogêneos, "subótimos" e, originalmente, não eram destinados para geração de modelos 3D. No primeiro estudo as etapas de pré-processamento e integração são realizadas com o FME e a geração do modelo CityGML em si é feita com o software *BuildingReconstruction*. No segundo, são aproveitados dados em 3D pré-existentes, sendo realizadas somente etapas de integração de dados atualizados e agregação ao módulo *Building*.

Buyukdemircioglu, Kocaman and Isikdag (2018) também utilizaram técnicas semiautomáticas para geração de modelo semântico da cidade de Izmir-Turquia, incluindo no mesmo fluxo de trabalho a reconstrução das edificações e a texturização, utilizando

métodos fotogramétricos. A utilização de software *BuildingReconstruction* possibilitou a extração automática dos formatos dos telhados das edificações, proporcionando um aspecto visual mais similar à realidade. Uma das propostas de estudos futuros foi a implementação de controle de qualidade e avaliação do produto final gerado, para que seja comprovada a integridade e validade do modelo.

Um estudo mais abrangente em relação à implementação de módulos do CityGML é apresentado por Kolbe, Burger and Cantzler (2015). No artigo são apresentados os resultados do projeto de criação do modelo semântico em 3D da cidade de Nova Iorque-Estados Unidos da América, tendo como dados de entrada *datasets* disponibilizados pelo portal de dados abertos da cidade. Dado que não havia uma homogeneidade nos dados disponibilizados pelos diversos departamentos da prefeitura, foi feita uma etapa de transformação e integração, visando o alinhamento dos dados com a a estrutura prevista no CityGML. Além disso, foram adotadas estratégias de armazenamento para lidar com o grande volume do conjunto de dados. O módulo *Building* foi implementado em LoD1. O Modelo Digital do Terreno (MDT) disponibilizado como *raster* foi transformado em uma TIN visando facilitar a integração com os demais módulos. A transformação dos dados de eixo central das vias para LoD0Network do módulo *Transportation* foi feita de forma direta. Para obter o LoD1 foi feita uma etapa automática de integração dos dados de eixo central com dados de uso da terra disponibilizados por outro departamento. Também foram integrados dados para implementação dos módulos *LandUse*, *Vegetation* (composto somente por feições do tipo *SolitaryVegetation*), e *WaterBody*. Todos módulos foram agregados e armazenados em uma base de dados 3D e disponibilizados para visualização.

Soon and Khoo (2017), apresentam as etapas de implementação do projeto nacional de mapeamento 3D da cidade-estado de Singapura. Assim como o da cidade de Nova Iorque, um modelo abrangente em relação ao número de módulos foi gerado. Ao todo foram implementados 8 módulos no projeto em questão. O módulo *Relief* foi gerado por nuvem de pontos, sendo representado por uma TIN (LoD0). O módulo *Building* foi gerado em LoD1 e 2, a partir de dados cadastrais em 2D e nuvem de pontos. Uma alternativa utilizada visando otimizar a visualização dos dados foi a de salvar cada instância de edificação em um único arquivo CityGML. Os módulos *Transportation*, *Bridge* e *Tunnel* foram gerados em LoD0, 1 e 2. Já, os módulos *WaterBody* e *Vegetation* foram gerados em LoD0 e 1. Por fim, o módulo *CityFurniture* foi implementado em LoD2. Também foram geradas *CodeLists* específicas da área de estudo, possibilitando a criação de valo-

res de atributos visando a adequação com a cultura local. O modelo final passou por uma fase de validação, apresentando inconsistências relativas à integridade geométrica, sendo justificadas pela existência de etapas manuais durante a fase de aquisição dos dados de entrada. O autor não apresentou análises quantitativas das inconsistências identificadas.

Outros estudos nessa vertente podem ser encontrados na literatura como Malambo and Hahn (2010), onde foi feita a implementação do módulo *Building* em LoD 1 e 2 com auxílio de dados LiDAR (*Light Detection And Ranging*) do campus da *Stuttgart University of Applied Sciences*-Alemanha e Janečka (2019)[8] que apresenta a conversão de dados geográficos disponibilizados de forma aberta no geoportal da cidade de Praga-República Checa para CityGML. No último, uma área de teste foi utilizada e foram implementados e validados os módulos *Building*, *Bridge* e *Relief*.

Já Goetz and Zipf (2012) e Goetz (2013) trazem sugestões de *frameworks* de derivação automática do módulo *Building* em todos níveis de detalhamento com base em dados abertos de mapeamento colaborativo do *OpenStreetMaps* (OSM)¹. São previstas duas etapas para a conversão. A primeira semântica, consistindo na realização do mapeamento de atributos de um modelo para o outro, e a segunda trata da conversão geométrica da base de dados para o CityGML. Não são realizados estudos de caso em larga escala. São apresentadas algumas limitações como a necessidade de obtenção da altura das edificações a partir do completamento de atributos por parte dos colaboradores, etapa essa muitas vezes negligenciada; falta de precisão geométrica nos dados de entrada, gerando inconsistências no modelo final; e a limitada disponibilidade de edificações com mapeamento interno, essenciais para conversão das edificações em LoD3 e 4. Além disso, não é apresentada uma solução para obtenção da altimetria da base das edificações, dado necessário para geração do módulo *Building* em qualquer nível de detalhamento.

Os estudos que não mencionam validações formais finais, abordam algumas limitações dos modelos gerados em termos de geometria e semântica, sendo apresentadas possíveis soluções ou indicando extensões para trabalhos futuros. Além disso, as etapas de pré-processamento para harmonização dos dados de entrada citadas tendem a reduzir ou eliminar as inconsistências no modelo final.

A Tabela 2.3 mostra um comparativo dos estudos citados levando em consideração a abordagem utilizada, nível de detalhamento dos módulos implementados, expresso pela numeração na tabela, e validações aplicadas nos modelos gerados. O presente trabalho é indicado como Item 11.

¹<https://www.openstreetmap.org/>

Tabela 2.3: Comparativo de abordagem, completude, detalhamento e validação dos trabalhos relacionados. Referências: [1] - Sengul (2010), [2] - Agugiaro (2014), [3] - Agugiaro (2016), [4] - Buyukdemircioglu, Kocaman and Isikdag (2018), [5] - Kolbe, Burger and Cantzler (2015), [6] - Soon and Khoo (2017), [7] - Malambo and Hahn (2010), [8] - Janečka (2019), [9] - Goetz and Zipf (2012), [10] - Goetz (2013)

Estudo		[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	
Abordagem	Auto					✓			✓	✓	✓		
	Semi-auto	✓	✓	✓	✓		✓	✓				✓	
Módulos implementados	Bldg	2	2	2	2	1	2	2	2	2	4	2	
	Trans					1	2	1				1	
	Veg					1	1	1				2	
	Tun						2						
	Brid						2		2				
	Frn						2					2	
	Wtr						1	1				1	
	Luse						1					1	
	Dem	0					0	0	0	2			1
	Níveis de validação	Esquema		✓	✓	✓	✓	✓		✓			✓
Geométrica							✓		✓			✓	
Semântica				✓			✓		✓			✓	
Dados de entrada	Oficiais	✓	✓	✓	✓	✓	✓	✓	✓			✓	
	Mapeamento colaborativo									✓	✓	✓	

Fonte: O autor.

Esta pesquisa possui pontos similares com as pesquisas citadas, se diferenciado por propor uma abordagem híbrida de utilização de dados abertos oficiais e de dados de mapeamento colaborativo visando atenuar a falta de completude de atributos dos *datasets* disponibilizados pelos órgãos governamentais. Também se equipara aos estudos de maior abrangência se tratando da quantidade de módulos do CityGML implementados e de validações realizadas. Além disso, nos modelos gerados disponibilizados de forma aberta, não é verificada a implementação da identificação automática de superfícies compartilhadas no módulo *Building*. Tal implementação é realizada, com algumas limitações, no presente estudo.

2.6 Considerações Finais

Neste capítulo foi apresentada uma visão geral dos aspectos técnicos relacionados à integração de dados espaciais, além de um breve extrato das especificações técnicas relativas ao CityGML, necessário para o entendimento da abordagem proposta e das implementações presentes no estudo de caso. Através dos trabalhos relacionados, pôde-se verificar uma grande variedade de estudos realizados apresentando diferentes abordagens

nas implementações de modelos baseados em CityGML, notando-se a ausência de uma abordagem que agregue os pontos positivos dos dois grupos de dados, a qualidade geométrica dos dados oficiais e a completude geométrica e de atributos e acurácia temporal dos dados de mapeamento colaborativo.

No próximo capítulo é apresentada a definição formal da abordagem proposta.

3 ABORDAGEM PARA INTEGRAÇÃO DE DADOS

Este capítulo descreve o desenvolvimento da abordagem que visa aplicar transformações e integrações nos dados de entrada para gerar, semi-automaticamente, o modelo 3D semântico baseado em CityGML.

Devido à particularidade de cada módulo do modelo CityGML, diferentes fluxos de trabalho são definidos de acordo com a necessidade de processos para se alcançar a integração desejada.

Com um visão realística, onde dificilmente serão encontradas fontes de dados oficiais que disponibilizem informações detalhadas das feições, como posicionamento de janelas, portas, divisões dos espaços internos e outros detalhes que fazem parte das feições e necessários para implementação dos módulos em LoD3 e 4, tais níveis de detalhamento não foram inseridos na abordagem. Apesar dessa ausência, a grande maioria das aplicações dos modelos 3D não necessitam desses níveis de detalhamento mais altos para serem executadas.

3.1 Fluxograma para o módulo *Building*

Para o módulo *Building* são inseridos os dados cadastrais 2D obtidos de base de dados oficiais. É importante salientar que os dados de entrada devem ter nível de acurácia posicional compatível com a geração do modelo em CityGML no nível de detalhamento desejado, conforme Tabela 2.2. Por exemplo, para gerar o módulo *Building* em LoD2, os dados de entrada devem ter acurácia posicional de, pelo menos, 2 metros.

Uma etapa de pré-processamento é feita com o objetivo de validar a geometria dos dados de entrada do ponto de vista topológico, verificando possíveis inconsistências como pontos duplicados, colineares e auto-interseções. Feições que não possam ser corrigidas de forma automática são descartadas ou, caso haja disponibilidade de recursos para correção manual, tais feições podem ser recolocadas no fluxo. Nessa etapa também é feito o mapeamento de atributos inicial de acordo com a classificação inicial das feições. Na hipótese de existir mais de uma base de dados de entrada, todos os aspectos técnicos de integração de dados citados na seção 2.1 devem ser levados em consideração para homogeneizar as bases.

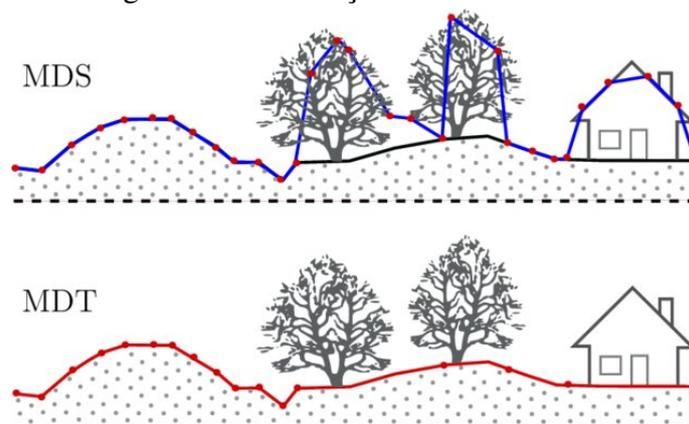
Caso os dados de entrada não possuam em seus atributos os endereços das feições, é possível obtê-los de forma automática através do processo de extração de informações

textuais a partir das suas coordenadas geográficas, chamado de geocodificação reversa (KOUNADI et al., 2013). Algumas plataformas *on-line* disponibilizam o serviço de geocodificação reversa de forma gratuita, com restrições de número de requisições variáveis de acordo com a plataforma. Uma análise da qualidade dos serviços de geocodificação reversa disponibilizados pode ser encontrada em Kilic and Gülgen (2020).

Na etapa seguinte é realizada a integração dos dados oficiais com a base de dados de mapeamento colaborativo, onde inicialmente é feita a comparação geométrica entre as feições visando encontrar elementos análogos nas duas bases, seguindo do mapeamento de atributos, caso haja correlação geométrica entre duas feições.

Os dados de altimetria são obtidos realizando a interseção dos dados em 2D com dados 3D disponibilizados. Nesta abordagem são previstos dois conjuntos dados de altimetria distintos, um com dados do MDT e outro com dados do MDS (Modelo Digital de Superfície), este último representa a superfície terrestre considerando o terreno e todos elementos que estão acima do nível do solo como vegetação, vias elevadas e edificações (CARRILHO, 2016). A Figura 3.1 mostra graficamente a diferença entre os dois modelos. A altimetria da base das edificações é obtida da interseção com o MDT e a altimetria do topo das edificações da interseção com o MDS. Caso a altimetria de topo esteja presente em algum atributo nativo, conforme descrito em Kolbe, Burger and Cantzler (2015), o mesmo pode ser reaproveitado.

Figura 3.1: Diferença entre MDS e MDT



Fonte: Carrilho (2016).

São feitas 3 filtragens de área e altura distintas de acordo com o especificado na Tabela 2.2 para cada nível de detalhamento. Feições a serem implementadas em LoD2 são desagregadas em superfícies. As superfícies que possuem compartilhamento entre duas edificações são identificadas, assim, a geometria da superfície compartilhada

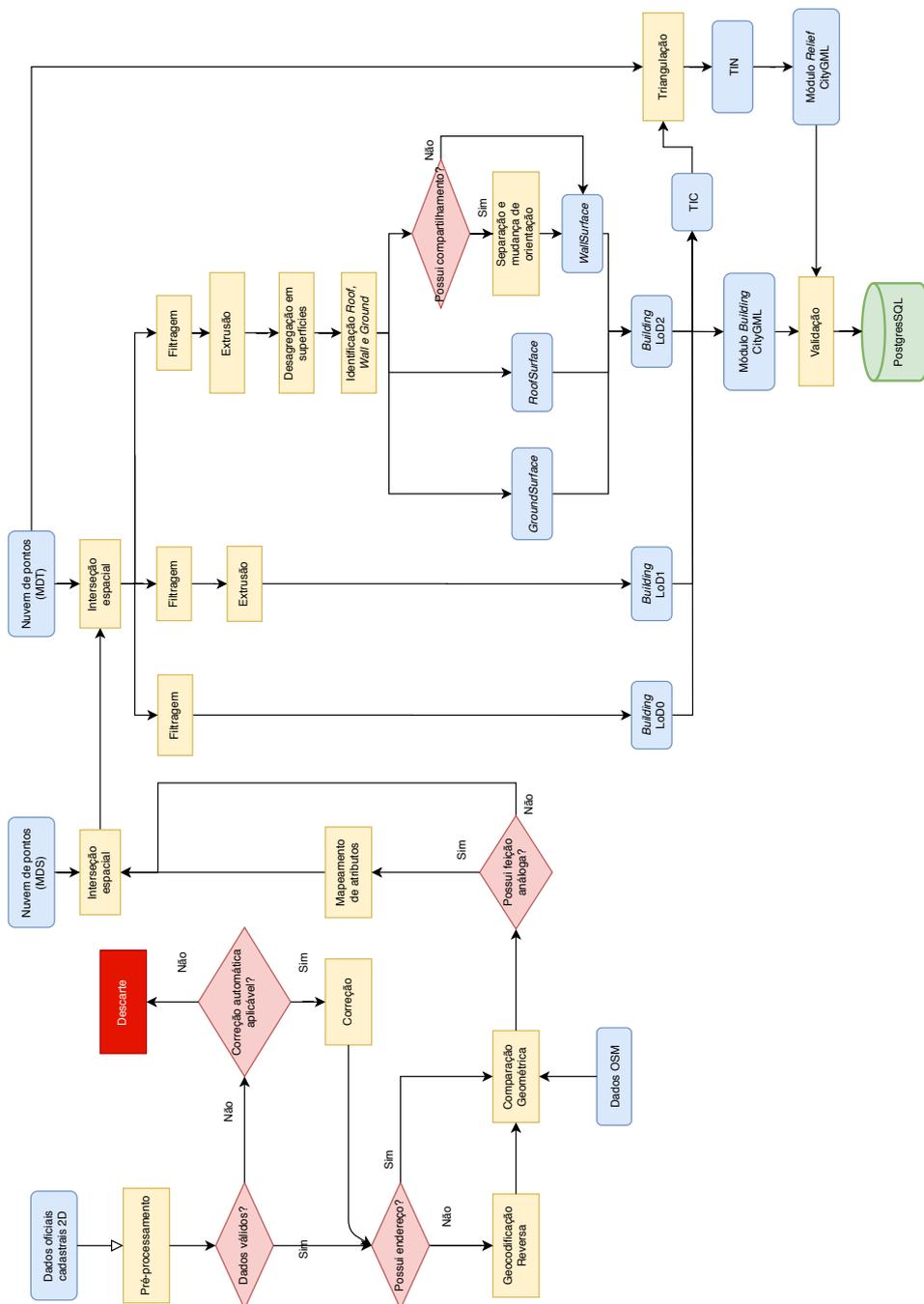
é implementada uma única vez e referenciada com o mecanismo *Xlink* pelas duas edificações que possui relacionamento hierárquico, evitando redundância na implementação de geometrias. Nestes casos, em umas das edificações, essa superfície compartilhada deve ser referenciada com mudança na orientação dos vértices, utilizando a ferramenta *gml:OrientableSurface*. Isso é necessário para que as plataformas de visualização possam interpretar corretamente as geometrias e exibi-las em ambas edificações

No processo seguinte é feita a integração com o módulo de relevo, gerando a chamada TIC, que garante a integridade entre as feições evitando inconsistências como feições “enterradas” ou “flutuando” sobre o terreno. Cada feições podem ter diferentes TIC de acordo com os níveis de detalhamento implementados. Tal etapa é indicada como trabalho futuro por Janečka (2019) mas é realizada em outros estudos.

A exemplo de Soon and Khoo (2017) e Janečka (2019) é prevista uma etapa de validação do modelo gerado que verifica aspectos geométricos, semânticos e de esquema.

Por fim, para facilitar a gestão, exploração e análise do modelo gerado, é feita a exportação para base de dados geográficos. O fluxo de trabalho do módulo *Building* é mostrado na Figura 3.2.

Figura 3.2: Fluxo de trabalho para o módulo Building

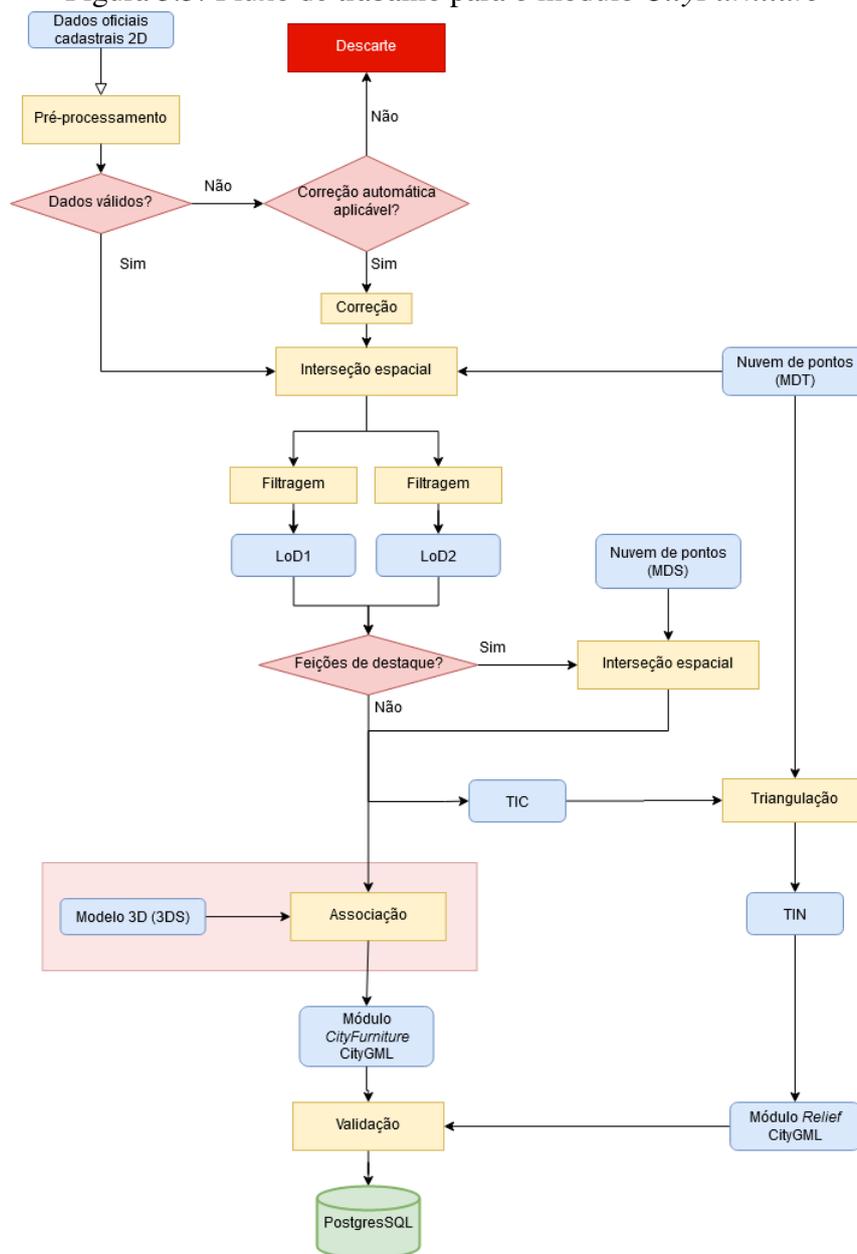


Fonte: O autor.

3.2 Fluxograma para o módulo *CityFurniture*

O módulo *CityFurniture* tem uma versão simplificada de fluxo de trabalho, uma vez que a geometria e a modelagem das feições são menos complexas nesse módulo. A fase inicial de pré-processamento é similar à do módulo *Builing*, tendo o mesmo objetivo de verificar inconsistências geométricas nos dados de entrada e realizar o mapeamento de atributos nativos dos dados oficiais. Os dados de altimetria da base das feições são obtidos com a interseção com o MDT de entrada e são feitas duas filtragens para os níveis de detalhamento LoD1 e 2, uma vez que a Tabela 2.2 não prevê feições deste módulo em LoD0. Em seguida é feita uma seleção de feições que tenham destaque em altimetria, como antenas de rádio de grande porte e torres de transmissão de energia, por exemplo. Tais feições passam por uma fase adicional de interseção com o MDS, para obtenção da altimetria do topo da feição, de forma que a sua representação no modelo seja mais próxima da realidade. Posteriormente são geradas as TIC para posterior integração com o módulo *Relief*. Feições com possuam forma similar, como postes, pontos de ônibus e antenas de transmissão de energia podem, opcionalmente, ter sua geometria representada de forma implícita com a associação com modelos 3D existentes para representação das feições de modo mais realístico. Por fim, o módulo *CityFurniture* é gerado, validado e armazenado na base de dados 3D. O fluxo de trabalho do módulo *CityFurniture* é mostrado na Figura 3.3.

Figura 3.3: Fluxo de trabalho para o módulo *CityFurniture*



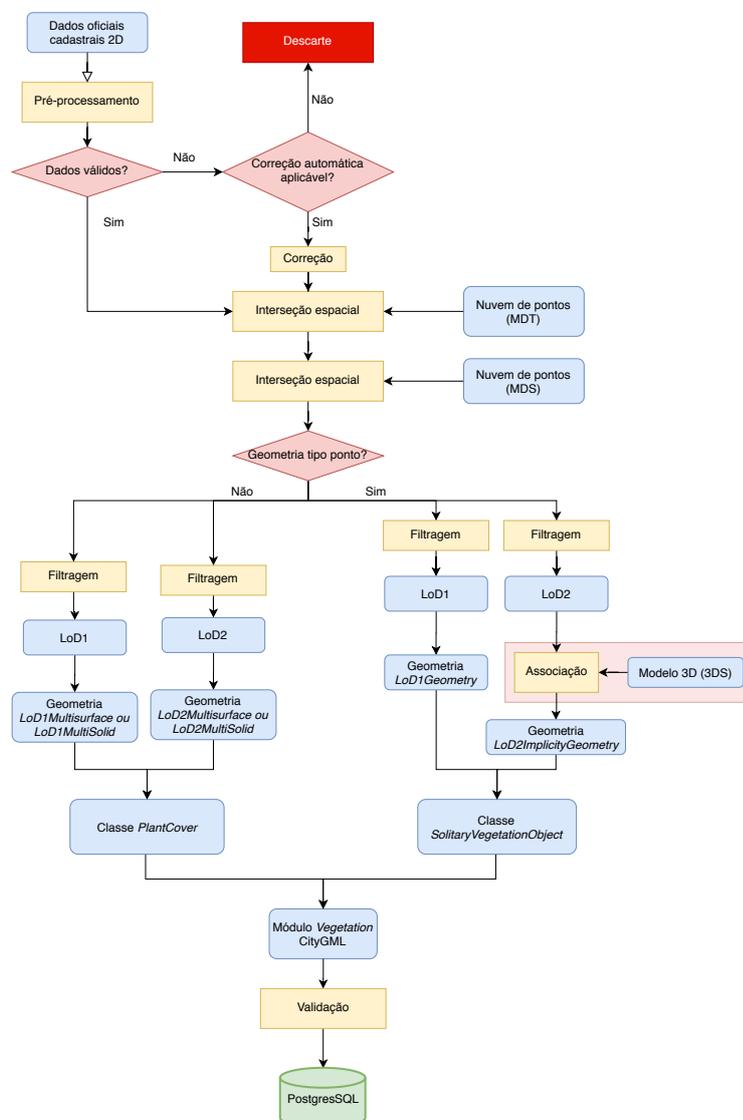
Fonte: O autor.

3.3 Fluxograma para o módulo *Vegetation*

Um fluxo de trabalho similar é aplicado no módulo *Vegetation*. Todas as etapas se equiparam com as do módulo *CityFurniture*. Uma etapa adicional de filtragem de acordo com o tipo de geometria da feição é feita para identificar as feições da classe *SolitaryVegetationObject*, com geometria do tipo ponto, e as feições da classe *PlantCover*, com geometria do tipo área, caso não haja classificação inicial na base de dados oficial.

Feições com geometria do tipo linha não são utilizadas nesse módulo, não sendo validadas na fase de pré-processamento. A Tabela 2.2 é utilizada como referência para a filtragem em LoD1 e 2. A associação com geometrias implícitas é feita somente em LoD2 na classe *SolitaryVegetationObject*, de acordo com a Tabela 2.2. Feições da classe *PlantCover* podem ser implementadas com geometria *gml:MultiSurface* ou *gml:MultiSolid*. Caso se opte por implementar as geometrias com *MultiSurface*, não há necessidade de se realizar a etapa de interseção com o MDS. Por fim, as duas classes são agregadas para gerar o módulo *Vegetation*, sendo validado e armazenado na base de dados 3D. O fluxo de trabalho do módulo *Vegetation* é mostrado na Figura 3.4.

Figura 3.4: Fluxo de trabalho para o módulo *Vegetation*



Fonte: O autor.

3.4 Fluxograma para o módulo *WaterBody*

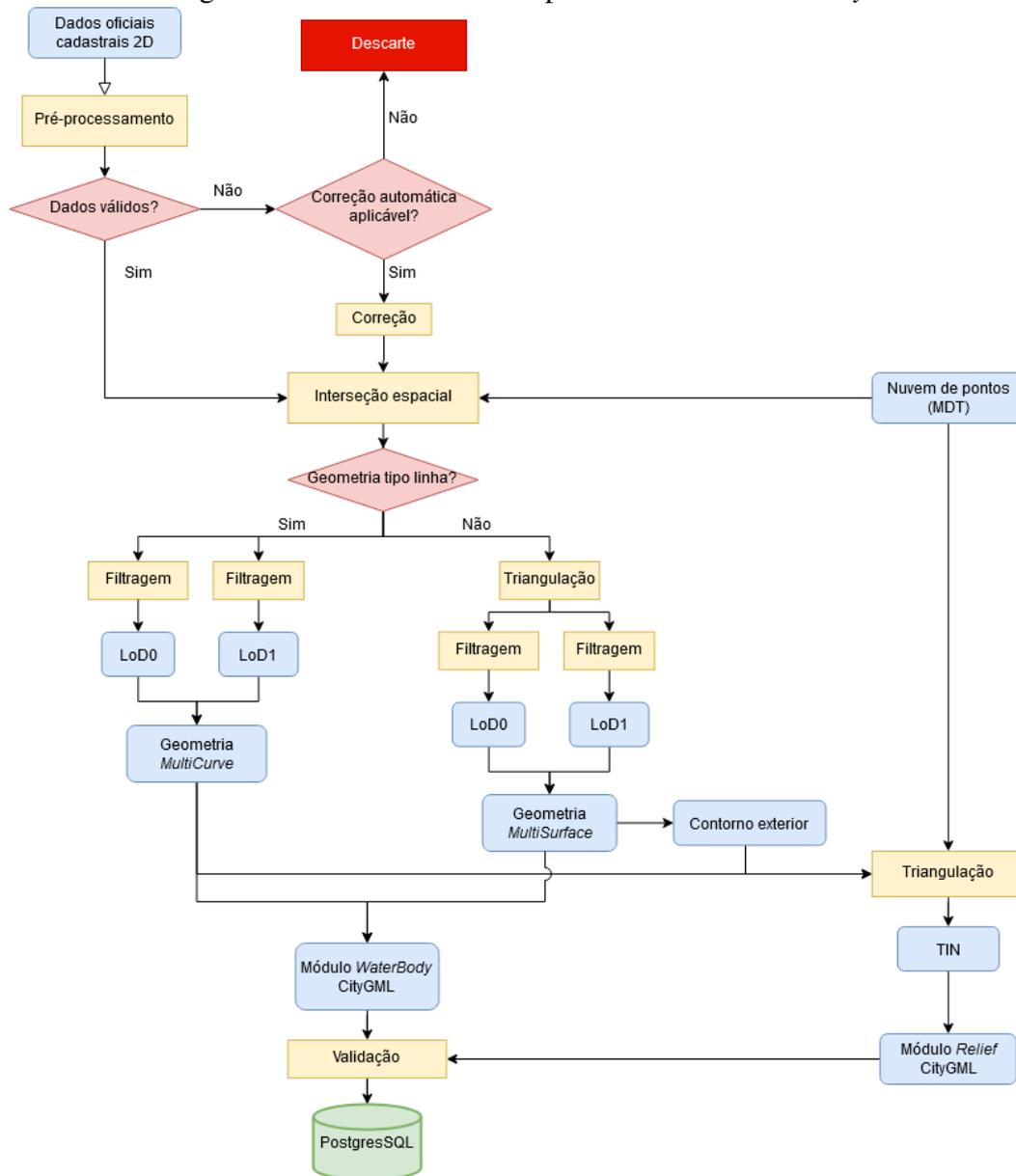
As etapas iniciais de validação e interseção com o MDT do fluxo de trabalho do módulo *Waterbody* são similares às dos módulos anteriores.

Neste módulo, a interseção das feições poligonais com o MDT gera superfícies curvas, uma vez as feições hidrográficas acompanham o contorno do relevo, que é acidentado por natureza. A presença de superfícies curvas é uma inconsistência geométrica no CityGML. Para contornar esse problema é feita a triangulação das superfícies curvas, subdividindo-a em múltiplos triângulos planos.

Em seguida são realizadas filtragens somente para LoD0 e 1. Em LoD2 os corpos d'água devem, necessariamente, ser representados como sólidos, onde a base é a interseção com o relevo e o topo é o nível do espelho d'água. Para que a implementação em LoD2 seja possível, é necessário que existam dados de batimetria para modelar a base dos corpos d'água, conforme abordado na seção 2.4.6, o que dificilmente se encontra nas bases de dados abertas. Dessa forma, tal modelagem não foi contemplada nessa abordagem.

Nas especificações técnicas não há uma definição específica de generalização das feições de acordo com o LoD neste módulo. Há a indicação que feições com "extensão significativa" devem ser modeladas como superfícies e os demais elementos do módulo como curvas. Dessa forma, fica a critério do modelador definir a extensão considerada significativa no contexto da implementação.

A integração com o módulo *Relief* se faz por parte das feições com geometria *MultiCurve* e pelo contorno externo das feições com geometria *MultiSurface*. Posteriormente as feições são agregadas para formar o módulo *WaterBody*, que passa pela fase de validação e armazenamento na base de dados 3D. O fluxo de trabalho do módulo *Vegetation* é mostrado na Figura 3.5.

Figura 3.5: Fluxo de trabalho para o módulo *WaterBody*

Fonte: O autor.

3.5 Fluxograma para o módulo *Transportation*

Assim como nos módulos anteriores, o módulo *Transportation* também passa por etapas iniciais de pré-processamento e interseção com o MDT para obtenção dos dados de altimetria. A forma mais comum de se encontrar as feições de transporte nas bases de dados abertas é na forma de redes lineares topologicamente conectadas, as chamadas malhas viárias, onde os segmentos lineares da rede representam o eixo central das vias e os nós, a interseção entre elas Hu et al. (2007). A Figura 3.6 mostra um exemplo de malha

viária.

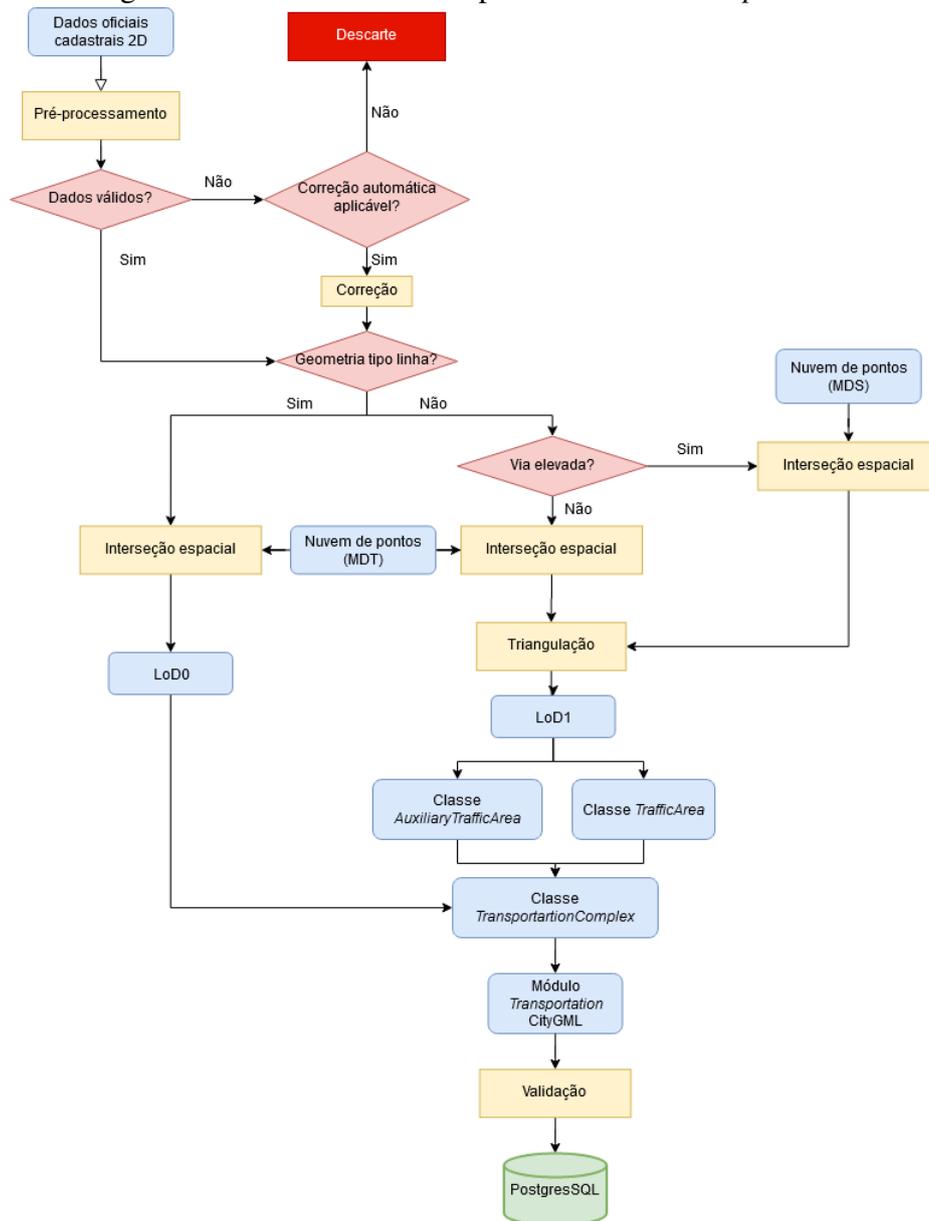
Figura 3.6: Exemplo de malha viária em amarelo sobreposta em imagem aérea



Fonte: Adaptado de Bastani et al. (2018).

De posse desse conjunto de dados, a conversão para LoD0Network é aplicada de forma direta. Para implementação em níveis de detalhamento superiores é preciso obter o contorno das vias na forma de superfícies. Para isso, é possível derivar as superfícies das malhas viárias com suporte de dados auxiliares (KOLBE; BURGER; CANTZLER, 2015) ou obter de forma automática através da classificação de imagens. Uma revisão de técnicas de obtenção de dados de transporte, tanto como malha viária como em superfícies, pode ser encontrada em Wang et al. (2016).

De acordo com a qualidade dos dados obtidos, é possível categorizá-los em *TrafficArea* ou *AuxiliaryTrafficArea*, de acordo com exposto na seção 2.4.4. Para obter maior realismo no modelo gerado, as vias elevadas como viadutos, passam pela etapa de aquisição da altimetria com interseção com o MDS. Por fim, todas feições são agregadas para gerar o módulo *Transportation*, sendo validado e armazenado na base de dados 3D. O fluxo de trabalho do módulo *Transportation* é mostrado na Figura 3.7.

Figura 3.7: Fluxo de trabalho para o módulo *Transportation*

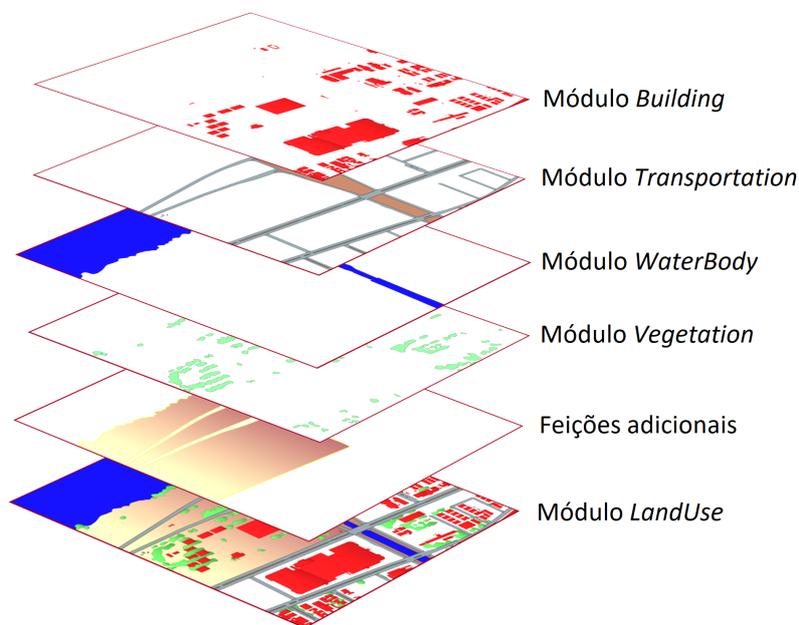
Fonte: O autor.

3.6 Fluxograma para o módulo *LandUse*

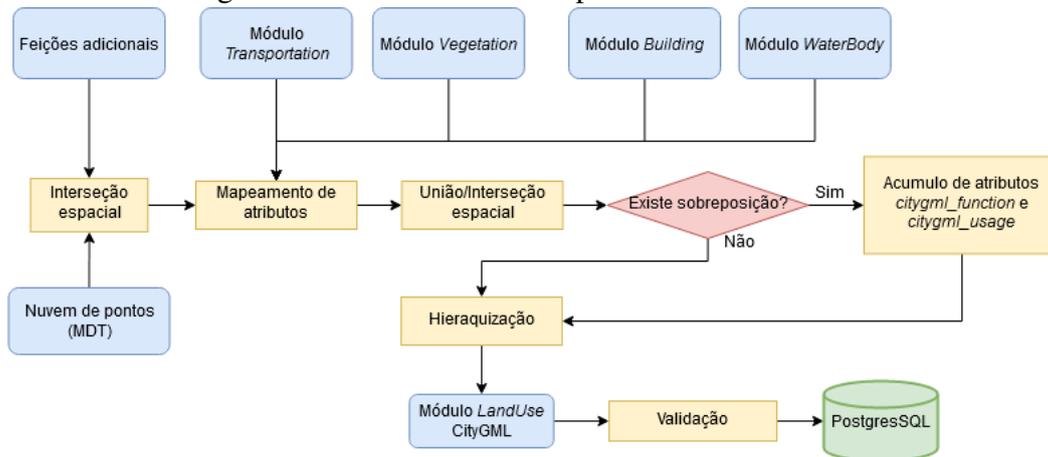
O módulo *LandUse* pode ser derivado da união/interseção de todos os demais módulos. Inicialmente deve ser feito o mapeamento dos atributos das feições de entrada para se adequarem à *codelist* prevista para o módulo. Na geometria final do módulo não deve haver superfícies sobrepostas. Caso existam nos dados de entrada regiões com sobreposição, as mesmas são identificadas e os atributos de função e uso, *citygml_function* e *citygml_usage*, respectivamente, de abas são acumulados na geometria de saída, con-

forme mencionado em 2.4.7. Para visualização final e definição do atributo de classe, *citygml_class*, que deve possuir um único valor, é proposta uma hierarquia de classes conforme mostrado na Figura 4.13, onde as feições de maior destaque visual na paisagem recebem prioridade de visualização. A ausência do módulo *CityFurniture* na composição do módulo *LandUse* é devida ao fato de não haver feições nesse módulo que tenham recobrimento espacial significativo. Além disso, é prevista a incorporação de feições adicionais como parques, cemitérios, quadras de esporte, entre outras que somente são previstas para serem implementadas nesse módulo. O fluxo de trabalho do módulo *LandUse* é mostrado na Figura 3.9.

Figura 3.8: Hierarquia de camadas no módulo *LandUse*



Fonte: O autor.

Figura 3.9: Fluxo de trabalho para o módulo *LandUse*

Fonte: O autor.

3.7 Considerações Finais

Neste capítulo foi apresentada a abordagem para aplicar transformações e integrações nos dados de entrada para gerar, semi-automaticamente, o modelo 3D semântico baseado em CityGML. O módulo *Relief* não foi apresentado separadamente pois tem integração com todos os outros módulos, sendo abordado em paralelo nos fluxos de trabalho apresentados. Para implementação do módulo *Relief* foi utilizada uma sequência de passos adaptada da proposta por Yan et al. (2019), sendo:

- Obter a altimetria da base das feições projetando seu contorno 2D no MDT de entrada;
- Definir as TIC como sendo a base das feições em 3D;
- Reconstruir o terreno considerando as TIC como condições de contorno.

A abordagem proposta foi formalizada no formato de fluxos de trabalho, visando facilitar sua implementação em diferentes contextos. É importante salientar que diversos cenários de implementação podem existir, desde os mais simples, com dados de entrada íntegros, homogêneos e atualizados, aos mais complexos, onde se dispõe de dados dados com diferentes formatos, estrutura de metadados e níveis de qualidade. Dessa forma, as abordagens aqui apresentadas são propostas, e não têm o intuito de ser universais, podendo receber alterações de acordo com o contexto de aplicação.

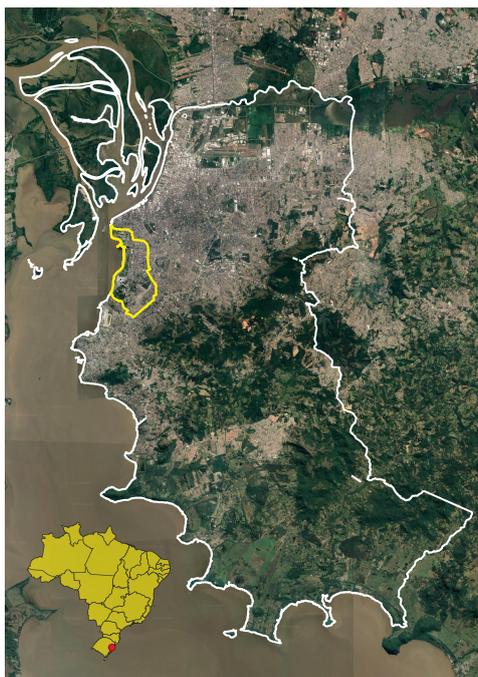
4 ESTUDO DE CASO

Neste capítulo é realizada uma aplicação prática da abordagem proposta utilizando dados de entrada que representam feições do mundo real. Inicialmente é feita uma caracterização da área de teste e dos dados de entrada, seguida da modelagem 3D dividida por módulos do CityGML. Por fim, são apresentados os dados de validação do modelo gerado em três níveis: esquema, geométrica e semântica.

4.1 Área de teste

Porto Alegre é a capital do estado do Rio Grande do Sul - BR. Sua população estimada em 2020 é de 1.488.252 pessoas e sua área total é de 495.390 km² (IBGE, 2020). A divisão administrativa é feita em 94 bairros, dos quais foram escolhidos três como área de teste: Menino Deus, Santa Tereza e Praia de Belas (Figura 4.1). Tal escolha foi feita por se tratar de uma região da cidade caracterizada por apresentar relevo relativamente acidentado e com feições heterogêneas, envolvendo complexos esportivos, centros comerciais, parques, áreas residenciais entre outras. A área englobando os três bairros é de 9.394 km².

Figura 4.1: Área de teste em destaque na cor amarela e contorno externo do município em branco



Fonte: O autor. Imagem: *Google*.

4.2 Dados de entrada

Em seu último mapeamento cadastral datado de 2010, o município foi dividido em 1999 cartas na escala de 1:1000. A Prefeitura Municipal de Porto Alegre (PMPA), através da Secretaria Municipal do Meio Ambiente e da Sustentabilidade (SMMAS/PMPA), disponibiliza os dados geográficos resultantes desse mapeamento de forma aberta, seguindo as diretrizes de governança inteligente em relação aos dados governamentais. Os arquivos podem ser encontrados em SMMAS/PMPA (2020a) nos formatos PDF e DWG. Essa base de dados fornece o contorno geométrico em 2D das feições, cuja classificação foi feita em 9 categorias: Sistema de Transportes, Energia, Hidrografia, Vegetação, Mineração, Estrutura Urbana, Altimetria, Pontos de Referência e Limites. Cada categoria, por sua vez, é dividida em temas. A categoria Sistemas de Transportes, por exemplo, é dividida nos temas: Sistema Viário, Meio Fio, Separadores Físicos, Rótulas e Acostamentos, Quadras Viárias, Ferrovias, Pontes e Viadutos, Passarelas, Escadarias e Rampas, Aeroportos, Paradas de Ônibus e Terminais Rodoviários. O produto cartográfico final gerado recebeu classificação "Classe A" de acordo com os níveis de tolerância planimétrica estipulados em BRASIL (1984), sendo compatíveis como insumos para implementação de modelos em CityGML até LoD3. Demais informações técnicas relativas ao projeto de mapeamento em questão podem ser encontradas em Consórcio Guaíba (2011).

A base disponibilizada no portal é derivada de um conjunto de arquivos vetoriais no formato ShapeFile. Este conjunto de dados possui como informações semânticas o tema no qual a feição foi classificada e, as de maior relevância, possuem o nome. Além disso, existe uma maior integridade espacial nos dados originais, pois não passaram pelo processo de edição e recorte necessários para dividi-los em cartas de 1:1000. Dessa forma, visando otimizar as etapas de pré-processamento, o conjunto de dados original foi requisitado e disponibilizado diretamente pela SMMAS/PMPA.

Na categoria Sistemas de Transporte não foram mapeados o eixo central das vias, necessários para obtenção do módulo *Transportation* em LoD0. Dessa forma, também foi inserido um arquivo do tipo shapefile contendo os eixos centrais das vias disponibilizado também em SMMAS/PMPA (2020b). Outros conjuntos de dados geográficos com diferentes temáticas também podem ser acessados no portal da SMMAS/PMPA (2020c), mas não foram utilizados por terem caráter predominantemente administrativo, fugindo do escopo da pesquisa.

Os dados de altimetria são oriundos de nuvem de pontos obtida para o mesmo

projeto de mapeamento, sendo dividida em MDT e MDS. Os arquivos no formato LAS foram requisitados e disponibilizados diretamente pela SMAMS/PMPA. A sua densidade média é de 2 pt/m².

Existem duas grandes bases de dados abertas que podem ser utilizadas para o enriquecimento semântico das feições: OSM e Wikimapia¹. O OSM foi escolhido por ter maior completude e qualidade temporal na área de estudo. Por ser uma fonte de dados aberta, de amplitude global e desenvolvida por mapeadores colaborativos, o produto final do OSM, apesar de passar por uma série de processos de validação, não é obtido através de métodos de produção cartográfica formais e bem estabelecidos. Com isso, a acurácia posicional e os níveis de generalização das feições difere das bases de dados disponibilizada pela PMPA. Por outro lado, o detalhamento das feições por meio de atributos é muito maior nos dados do OSM quando comparado com os dados oficiais. Além das divisões em categorias, as feições são descritas por meio de *tags*, onde o usuário mapeador pode incluir informações adicionais. De um total de 38 categorias de mapeamento disponibilizadas, foram utilizadas 22: *aerialway*, *aeroway*, *amenity*, *barrier*, *building*, *craft*, *emergency*, *highway*, *landuse*, *leisure*, *man_made*, *military*, *natural*, *office*, *power*, *public_transport*, *railway*, *route*, *shop*, *sport*, *tourism* e *waterway*. As demais não foram incluídas por estarem fora do escopo dos módulos do CityGML. Uma descrição detalhada de cada categoria pode ser encontrada em OSM (2019).

O OSM possui um sistema aberto de *tags*, permitindo que o colaborador inclua um número ilimitado de atributos para descrever cada feição. A plataforma de consulta *TagInfo*² apresenta a frequência de utilização de cada *tag*, bem como a descrição semântica das *tags* mais frequentes para servir de referência aos colaboradores.

Uma consequência do sistema aberto é o elevado número de *tags* diferentes (12.672 somente na categoria *building*). Isso possibilita uma descrição mais precisa das feições, porém ocasiona criação de *tags* redundantes, por exemplo, *house* e *residence*, além de outras inconsistências como utilização de idiomas nativos e erros de grafia.

¹<https://wikimapia.org/>

²<https://taginfo.openstreetmap.org/>

4.3 Modelagem 3D

A principal ferramenta utilizada na modelagem e integração dos dados no escopo da presente pesquisa é a FME³. A ferramenta foi escolhida por possuir suporte para dados geoespaciais e disponibilizar diversas funções chamadas de *transformers* que realizam manipulações nos dados de entrada. Além disso, existe compatibilidade com a linguagem *Python*, onde *scripts* podem ser executados utilizando o *transformer PythonCaller*. Os dados de entrada são carregados através de *Readers* específicos para cada formato de dado e, após manipulados, são salvos em *Writers*, onde é definido o formato de saída. As implementações são feitas em áreas de trabalho chamadas (*workspaces*). Dessa forma, a ferramenta atende a grande parte dos requisitos para a implementação do estudo de caso.

Para a realização de algumas etapas de digitalização que foram realizadas de forma manual é utilizado o *software* livre e de código aberto QGIS⁴.

4.3.1 Módulo *Building*

Seguindo o fluxo de trabalho mostrado na Figura 3.2, é feita a etapa de pré-processamento, iniciando com a validação das geometrias do ponto de vista topológico, verificando possíveis inconsistências como pontos duplicados, colineares, auto-interseções e sobreposições. Os *transformers* utilizados nessa etapa são *GeometryValidator* e *Area-GapAndOverlapCleaner*. Visando otimizar processamentos posteriores, também foi aplicado um filtro de área, selecionando somente as edificações com metragem superior a 16 m², seguindo as especificações da Tabela 2.2 para edificações LoD2, nível de detalhamento mais elevado implementado nessa pesquisa.

A base oficial possui na área de estudo 24.129 edificações. A filtragem de área excluiu 3.340, restando 20.789 edificações. Nenhuma instância apresentou inconsistências geométricas.

Posteriormente é feito o mapeamento inicial dos atributos de acordo com a sua classificação oficial. As edificações na base de dados oficial utilizada são classificadas nos temas edificações genéricas, notáveis, escolas, creches, faculdades/universidades, hospitais e igrejas/templos. Com base nas *codelists*⁵ do módulo *Building* do CityGML foi

³<<https://www.safe.com/>>

⁴<<https://qgis.org/en/site/>>

⁵Todas *codelists* do módulo *Building* podem ser encontradas em <<http://www.sig3d.org/codelists/standard/building/2.0/>>

Tabela 4.1: Mapeamento dos atributos da base oficial para *codelists* do módulo *Building*

Atributo oficial	Class	Class_code	Function	Function_code
Escolas	<i>Schools, education, research</i>	1100	<i>Building for education and research</i>	2070
Creches	<i>Schools, education, research</i>	1100	<i>Kindergarten or nursery</i>	2380
Faculdades/Universidades	<i>Schools, education, research</i>	1100	<i>College or university</i>	2100
Hospitais	<i>Healthcare</i>	1120	<i>Hospital</i>	2310
Igrejas/Templos	<i>Church institution</i>	1080	<i>Church</i>	2220

Fonte: O autor.

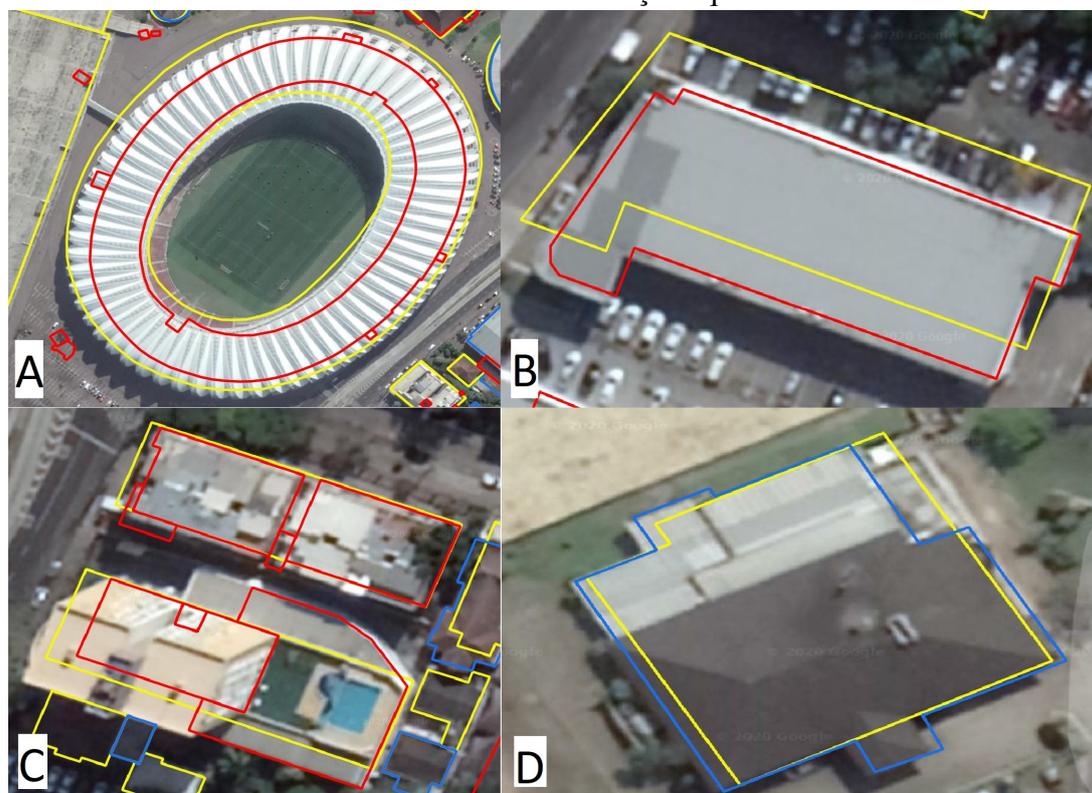
possível fazer o relacionamento direto para os atributos *AbstractBuilding_class*, *AbstractBuilding_function* e *AbstractBuilding_usage* conforme Tabela 4.1. Como nesse módulo os atributos *function* e *usage* são redundantes, o atributo *usage* foi omitido na tabela. O atributo "nome" é mapeado diretamente para o atributo *gml_name*.

Algumas edificações classificadas inicialmente como genéricas possuíam o atributo "nome", possibilitando atribuir-lhes os atributos classe e função. Devido à impossibilidade de se inferir a classe e função de uma edificação de forma automática a partir do nome, este passo foi feito de forma manual.

Como a base oficial não possui dados de endereço, a geocodificação reversa é realizada de forma automática utilizando o *transformer geocoder*. O *transformer* possui suporte para 12 serviços de geocodificação, de onde deve ser obtida uma *API Key* para execução das requisições. Foi selecionado o serviço da plataforma *Bing* devido à facilidade de obtenção da *API Key*, ser de acesso gratuito e ter o limite de requisições diárias compatível com o escopo da pesquisa. Foi utilizado um *script* em *Python* para adequar o formato de endereço retornado aos padrões do formato xAL utilizado no CityGML.

Apesar de existir uma classificação inicial das edificações da base de dados oficial, 95,08% delas são classificadas como genéricas, carecendo de descrição semântica. Dessa forma, uma alternativa encontrada para minimizar essa ausência de informações é a integração com a base de dados do OSM. Inicialmente é feita a associação geométrica entre as feições e, posteriormente, é feito o mapeamento de atributos, associando o atributo da feição do OSM com os existentes nas *codelists* no modelo CityGML. Para serem consideradas feições correspondentes, os critérios utilizados na categoria *building* são a comparação da área total dos polígonos e o distanciamento do ponto central da edificação. Feições cuja área varie em até 36% e ponto cujo central não diste 5m são consideradas correspondentes. Essa margem mais abrangente foi necessária devido à diferença de alguns aspectos de mapeamento entre as duas bases. A Figura 4.2 mostra casos particulares encontrados no processo de integração geométrica.

Figura 4.2: No mosaico, os contornos amarelos representam a base de dados do OSM, os contornos vermelhos representam as feições da base de dados oficial que não foram associadas à base do OSM e o contorno azul as feições que tiveram relacionamento



Fonte: O autor. Imagem: *Google*.

Feições da base de dados oficial que não apresentam nenhum relacionamento topológico com feições do OSM são excluídas do fluxo. Tal medida é adotada pois a acurácia temporal da base do OSM é melhor que da base oficial. Se uma instância da base oficial não se relaciona de alguma forma com outra da base do OSM, infere-se que ela deixou de existir. Após essa etapa, 696 feições foram excluídas, restando 20.093.

O caso A da Figura 4.2 mostra a diferença de acurácia temporal entre as bases de dados. A base do OSM mais recente representa o contorno do Estádio Beira-Rio após a reforma para a Copa do Mundo de 2014, enquanto a base oficial mostra o contorno antigo de 2010. Tal diferença impossibilitou a associação das feições de forma automática.

Já o caso B mostra a diferença de acurácia posicional entre as bases. A base de dados oficial foi gerada a partir de um processo formal de produção cartográfica, sendo as feições obtidas através da fotointerpretação de ortoimagens realizada por profissionais especializados que seguem vários protocolos de aquisição de feições para garantir uma padronização e validade do produto final. Além disso, as coordenadas geográficas foram referenciadas ao Sistema Geodésico de Referência SIRGAS2000 e suas coordenadas

representadas na Projeção Transversa de Mercator para Porto Alegre (TM-POA)⁶. Já a aquisição das feições da base do OSM é feita por colaboradores, não necessariamente com afinidade a atividade de produção cartográfica, utilizando como insumo imagens de satélite disponibilizadas pela plataforma *Bing* e suas coordenadas armazenadas em um sistema de referência global, WGS-84. Logo, todos esses fatores divergentes, além de pequenas diferenças que possam ser geradas na transformação de coordenadas entre sistemas de referência podem gerar uma discrepância na acurácia posicional entre as feições.

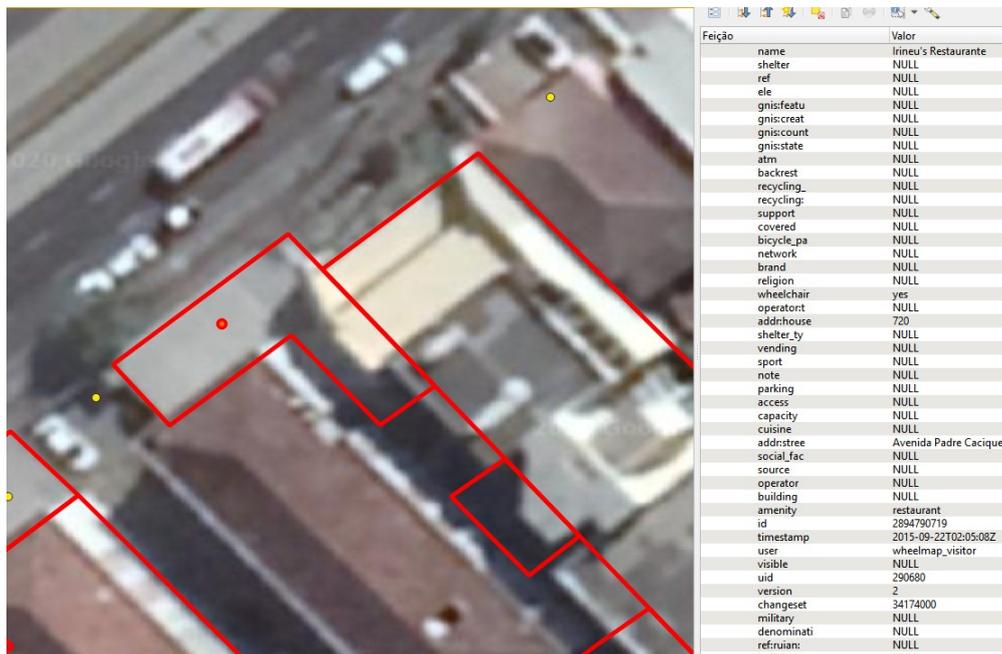
A diferença de entre os níveis de generalização entre as feições é evidenciada no caso C. A base de dados oficial diferencia mais as feições, fazendo uma separação maior entre os elementos e possui um contorno mais detalhado, adequado para produtos na escala 1:1000, enquanto a base do OSM disponibiliza um contorno mais genérico, impossibilitando a associação automática de alguns elementos.

Por fim, o caso D mostra um exemplo de associação entre feições. Apesar das diferenças de generalização e acurácia posicional, as edificações passaram nos critérios de comparação.

Outra forma de integração encontrada além da polígono - polígono foi a ponto - polígono. Em alguns casos a geometria do tipo ponto foi utilizada na base do OSM para mapear não feições pontuais, mas sim para mapear estabelecimentos comerciais. Na Figura 4.3 é mostrado um caso de restaurante mapeado como feição pontual.

⁶Os parâmetros que caracterizam este sistema de referência local são disponibilizados na forma de metadados no portal da SMAMS/PMPA

Figura 4.3: Feição do tipo ponto da base do OSM fornecendo atributos de restaurante conforme tabela lateral



Fonte: O autor. Imagem: Google.

Os polígonos em vermelho da base oficial não tiveram associação polígono – polígono, porém pôde ser feito o relacionamento com a feição do tipo ponto da base do OSM, que possuía atributos relacionados à edificação.

A base de dados do OSM possui 21.807 feições classificadas como edificação na área de estudo. Foram encontradas 19.405 associações polígono – polígono e 155 associações ponto – polígono.

Para limitar o número de *tags* utilizadas no mapeamento de atributos entre a base do OSM e os previstos nas *codelists* do modelo CityGML, foi feito um filtro de frequência, onde foram selecionadas as *tags* que tinham, pelo menos, 0,1% de utilização.

Após a filtragem, os dados do *TagInfo* foram separados por categoria e consolidados em planilhas. O relacionamento entre os valores das *tags* e os valores de atributos das *codelists* foi feito prioritariamente de forma automática utilizando o processo *token_sort_ratio* da biblioteca *fuzzywuzzy*⁷ disponível para linguagem *Python*. A biblioteca compara a grafia de duas palavras usando a distância de *Levenshtein*, fornecendo um valor de 0 a 100. Quanto maior o valor, maior a similaridade. A associação direta foi feita para valores acima de 90, aceitando pequenas diferenças de grafia como *industrial* (OSM) e *industry* (CityGML).

Como no modelo CityGML os valores de atributos previstos nas *codelists* são

⁷<<https://pypi.org/project/fuzzywuzzy/>>

Tabela 4.2: Número de feições da base oficial que tiveram atributos do OSM associados

Atributo	<i>citygml_class</i>	<i>citygml_function</i>	<i>gml_name</i>	<i>SAG</i>
Feições	19	147	116	14
Atributo	<i>operator*</i>	<i>website*</i>	<i>gml_description</i>	<i>YOC</i>
Feições	29	73	31	10

Fonte: O autor.

mais genéricos quando comparados com as *tags* do OSM e também são divididos em *class*, *function* e *usage*, parte do mapeamento foi feita de forma manual, para que houvesse o maior número de relacionamentos possível, incluindo relacionamentos como *apartments* (OSM), *habitation* (*AbstractBuilding_class* - CityGML) e *residential building* (*AbstractBuilding_function* - CityGML), cuja grafia é completamente diferente, mas que semanticamente estão relacionados.

Dessa forma, após feito o relacionamento geométrico entre as feições, é verificado se a feição da base do OSM possuía *tags*. Caso positivo, o valor do atributo correspondente da *odelist* é agregado à feição da base oficial.

Como a cardinalidade dos atributos *function* e *usage* é de [0..*], conforme observado na Figura 2.5, mesmo que a feição já possuísse atributos obtidos da base de dados oficial, foi considerada a complementação desses atributos caso o valor do atributo encontrado no mapeamento fosse diferente do já existente.

As planilhas com o mapeamento das *tags* do OSM para os atributos das *odelists* do modelo CityGML estão disponibilizadas no repositório de implementações dessa pesquisa⁸. A Tabela 4.2 mostra o número de feições com atributos mapeados após o processo. Os atributos *SAG* (*storeys above ground*) e *YOC* (*year of construction*) significam níveis acima do solo e ano de construção, respectivamente. Os atributos marcados com * não pertencem à estrutura de atributos base do CityGML mas foram associados como atributos genéricos por agregar informações importantes sobre as feições. O atributo *operator* indica o responsável por administrar ou operar a edificação. Além disso, o atributo que indica o identificador único da base de dados do OSM também foi deixado como atributo genérico por ser considerado uma referência externa indicando de qual feição dessa base de dados foram retirados os atributos para "enriquecimento" semântico das instâncias de edifícios do modelo em CityGML.

Percebe-se um baixo número de atributos agregados às feições da base oficial. Isso mostra que, apesar de apresentarem boa completude geométrica e qualidade temporal,

⁸https://github.com/mcmaieron/DWG_CityGML/tree/master/OSM_attribute_mapping

Tabela 4.3: Comparativo de completude temática nos dados do OSM

Cidade	# de edificações	Atributo "nome" (%)	Categorização da edificação (%)	YOC (%)	SAG (%)	Média de tags por edificação
Porto Alegre	21.807	0,8	3,4	0	0	2,14
Londres	8.470	12,6	58,5	0,2	17,8	4,71
Berlim	8.398	5,2	70,1	1,0	61,8	3,7
Amsterdam	22.981	0,6	57,9	97,9	0,3	5,1

Fonte: O autor.

os dados do OSM apresentam baixa completude temática na área de estudo. A Tabela 4.3 mostra um comparativo de completude temática de dados OSM em quatro cidades. Nota-se que as demais cidades possuem uma melhor cultura de mapeamento colaborativo, o que permitiria uma agregação mais significativa de atributos nessas regiões. Nessa comparação, foram utilizadas áreas de teste que eram visualmente semelhantes à área de teste deste trabalho, não abrangendo as cidades como um todo.

Dando sequência ao fluxo de trabalho, é realizada a obtenção dos dados de altimetria das edificações com a interseção com o MDS para obtenção da altimetria do topo e com o MDT para agregar a altimetria da base. A diferença entre as duas medidas resulta na altura da edificação, correspondente ao atributo *citygml_measured_height* do CityGML. Nessa etapa são utilizados os *transformers Clipper*, para identificar os pontos da nuvem que se intersectam com os polígonos das edificações, e *PointCloudStatistics-Calculator*, para agregar os valores de altimetria desses pontos. As medidas estatísticas utilizadas para gerar o valor de altimetria final foram mediana, no caso da altimetria de topo e menor valor, no caso da altimetria da base. A mediana foi utilizada por minimizar a influência de pontos que pertençam ao solo mas que, pela proximidade, foram considerados no cálculo da altimetria da edificação e o valor mais baixo foi adotado por garantir que todas as demais superfícies estejam ligadas ao solo (KOLBE; BURGER; CANTZLER, 2015).

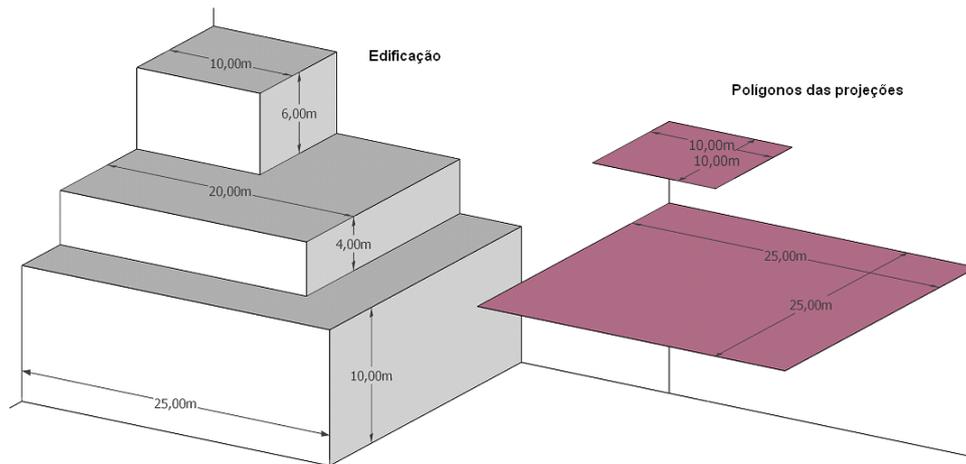
Uma característica geométrica do *dataset* de entrada é o detalhamento das edificações que possuam blocos construtivos consecutivos separados por alturas maiores que 6m, conforme Figura 4.4.

Tabela 4.4: Atributos e filtros de acordo com o nível de detalhamento do módulo *Building*

LoD	Filtro de área (m ²)	Filtro de altura (m)	Atributo <i>citygml_lod_name</i>	Feições implementadas
0	50	NA	<i>Lod0FootPrint</i>	12.345
1	36	3	<i>Lod1Solid</i>	14.392
2	16	2	<i>Lod2Solid</i>	19.662

Fonte: Adaptado de Gröger et al. (2012).

Figura 4.4: Detalhamento de feições com blocos construtivos consecutivos separados por alturas maiores que 6m



Fonte: Consórcio Guaíba (2011)

Apesar de se verificar uma relação entre as feições, não há menção de hierarquia entre as geometrias geradas. Dessa forma, para identificar tal relacionamento, é feita a análise espacial com o *transformer SpatialRelator*, sendo configurado para identificar feições que atendam ao predicado "estar no interior de" (*within*)⁹. A identificação dessas feições é importante para que, na etapa de interseção com o MDS, a área dos polígonos sobrepostos sejam desconsideradas no cálculo da altimetria dos polígonos da base.

A próxima etapa é a filtragem das feições para definir o nível de detalhamento que podem ser implementadas. Para isso, a Tabela 2.2 é usada como referência. A Tabela 4.4 sintetiza os parâmetros dos filtros aplicados, bem como o preenchimento de atributos. Nos filtros de dimensões entende-se que as feições devem ter, pelo menos, as dimensões estipuladas.

Cabe salientar que a mesma feição pode ser implementada em múltiplos LoD simultaneamente, por esse motivo a soma das feições implementadas é superior ao número de feições iniciais.

⁹Os relacionamentos espaciais são descritos por predicados - palavras-chave que indicam o tipo de relacionamento que cada geometria tem com outra. Essas relações são definidas pelo OGC, e são interpretadas no FME. Informações mais detalhadas sobre cada predicado podem ser encontradas em SAFE (2020).

A filtragem é a última etapa das feições em LoD0. A partir das geometrias finais, são geradas as TIC para posterior integração com o módulo *Relief*.

Em LoD1 as edificações são representadas com sólidos simples, de forma que o *transformer Extruder* é utilizado para a criação desses sólidos a partir dos polígonos de contorno da base e da altura baseada no atributo *citygml_measured_height*. Para as edificações que possuam blocos construtivos consecutivos é utilizado o *transformer SolidMerger* com a função de unir as geometrias dos sólidos geradas individualmente.

No nível LoD2 e superiores a fachada externa das edificações pode ser diferenciada semanticamente nas classes *_BoundarySurface* e *BuildingInstallation*. A classe *_BoundarySurface* representa parte da edificação com funções especiais como paredes (*WallSurface*), telhado (*RoofSurface*) e piso (*GroundSurface*). A classe *BuildingInstallation* é utilizada para elementos como sacadas, escadas externas ou chaminés, ou seja, objetos externos que afetem de forma significativa a aparência da edificação. Além disso, pequenos anexos às edificações como garagens, podem ser modelados na classe *BuildingPart*.

Como já mencionado, na base de dados utilizada no estudo de caso, não há menção de hierarquia entre as feições, de forma que todas são tratadas de forma individualizada. Mesmo que existam feições que representem elementos da classe *BuildingPart* ou *BuildingInstallation*, as mesmas devem ser identificadas e classificadas manualmente para posterior associação à edificação principal. Tal etapa não foi incluída no escopo da pesquisa.

Todavia, a diferenciação entre os elementos da classe *_BoundarySurface* pode ser feita, aumentando, o nível de descrição semântica do modelo final. Com o uso do *transformer GeometryCoercer* é possível separar os sólidos simples em superfícies individuais (*CompositeSurfaces*) e, com o uso do *transformer PlanarityFilter*, classificá-las em *WallSurface*, *RoofSurface* e *GroundSurface*. As 19.662 edificações geraram 235.274 superfícies individuais. A elas é atribuído um novo identificador único *gml_id*. O identificador único *gml_id* das edificações às quais fazem parte é armazenado no atributo *gml_parent_id*, para assegurar a hierarquia entre as geometrias.

Visando explorar ao máximo as potencialidades do CityGML dentro das possibilidades dos dados de entrada, é implementada a relação topológica entre superfícies e edificações utilizando o mecanismo *Xlink* do GML3. Com isso, a geometria das superfícies são implementadas separadamente no esquema do XML final e referenciadas na declaração das edificações. O exemplo da Figura 4.5 mostra a estrutura do XML utili-

zando essa abordagem.

Figura 4.5: Estrutura do XML utilizando *Xlink:href* para referenciamento de geometrias

```

<?xml version="1.0" encoding="UTF-8"?>
<core:CityModel xmlns:brid="http://www.opengis.net/citygml/bridge/2.0" xmlns:tran="http://www.opengis.net/citygml/transportation/2.0" xmlns:frn="http://www.opengis.net/citygml/furniture/2.0" xmlns:st="http://www.opengis.net/citygml/street/2.0" xmlns:trn="http://www.opengis.net/citygml/transportation/2.0" xmlns:frn="http://www.opengis.net/citygml/furniture/2.0" xmlns:st="http://www.opengis.net/citygml/street/2.0">
  <gml:boundedBy>
    <gml:Envelope srsDimension="3">
      <gml:lowerCorner>277319.491 1674176.202 0</gml:lowerCorner>
      <gml:upperCorner>278273.17 1675712.266 50</gml:upperCorner>
    </gml:Envelope>
  </gml:boundedBy>
  <core:cityObjectMember>
    <bldg:Building gml:id="bld_0c585224-1725-437c-9b6d-6741e0ed7505">
      <gml:name>HOTEL INTERCITY PREMIUM</gml:name>
      <bldg:class codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/AbstractBuilding_class.xml">1100</bldg:class>
      <bldg:function codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/AbstractBuilding_function.xml">2380</bldg:function>
      <bldg:usage codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/AbstractBuilding_usage.xml">2380</bldg:usage>
      <bldg:measuredHeight uom="urn:ogc:def:uom:UCUM: m">35.000</bldg:measuredHeight>
      <bldg:lod2Solid>
        <gml:Solid srsDimension="3">
          <gml:exterior>
            <gml:CompositeSurface>
              <gml:surfaceMember xlink:href="sfc_9bb80fa8-83ff-4296-8ed1-80c9d249870f"/>
              <gml:surfaceMember xlink:href="sfc_12a43271-a956-4f0c-b629-1e9919e2b9a0"/>
              <gml:surfaceMember xlink:href="sfc_74897064-4199-4159-a60d-a469d91c6085"/>
              <gml:surfaceMember xlink:href="sfc_ab825161-9aa7-4a40-a57b-1107aa1cbe5a"/>
              <gml:surfaceMember xlink:href="sfc_5eb461bf-2f3e-48c1-b18b-7261dd1e9d2d"/>
              <gml:surfaceMember xlink:href="sfc_ad8049af-595f-43a9-a15b-5b704dcad23f"/>
            </gml:CompositeSurface>
          </gml:exterior>
        </gml:Solid>
      </bldg:lod2Solid>
      <bldg:boundedBy>
        <bldg:GroundSurface gml:id="main_sfc_4f00c4bf-d50e-479a-6890-698296dead5d">
          <bldg:lod2MultiSurface>
            <gml:MultiSurface srsDimension="3">
              <gml:surfaceMember>
                <gml:Polygon gml:id="sfc_9bb80fa8-83ff-4296-8ed1-80c9d249870f">
                  <gml:exterior>
                    <gml:LinearRing>
                      <gml:posList>277882.459 1674896.548 0 277883.36 1674904.643 0 277886.223 1674904.315 0 277885.313 1674896.
                    </gml:LinearRing>
                  </gml:exterior>
                </gml:Polygon>
              </gml:surfaceMember>
            </gml:MultiSurface>
          </bldg:lod2MultiSurface>
        </bldg:GroundSurface>
      </bldg:boundedBy>
    </bldg:Building>
  </core:cityObjectMember>
</core:CityModel>

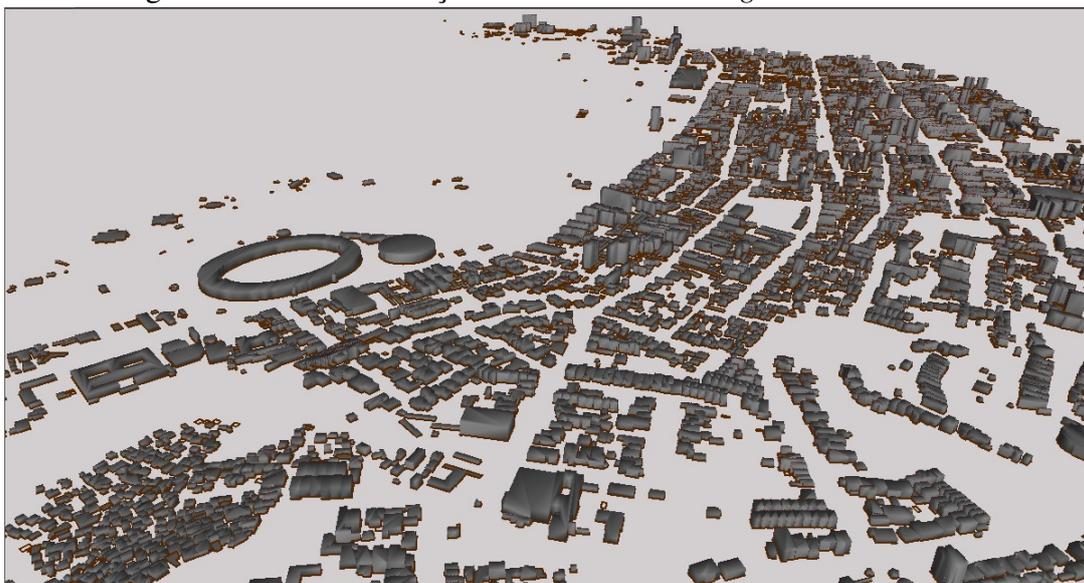
```

Fonte: O autor.

As superfícies que possuem compartilhamento entre duas edificações são identificadas com o *transformer SurfaceOnSurfaceOverlayer* e a elas é atribuído um novo identificador único. Para mudar a orientação da superfície referenciada em uma das edificações é criado o atributo *gml_geometry_reversed* e seu valor preenchido com "yes" antes de sua associação à uma das edificações.

Durante a implementação do estudo de caso houve a identificação de falhas na identificação das superfícies compartilhadas pelo *transformer SurfaceOnSurfaceOverlayer*. Essas falhas foram reportadas à equipe de suporte e desenvolvimento da ferramenta que executaram melhoramentos, aumentando a eficácia nessa etapa do fluxo de trabalho. Apesar disso, algumas inconsistências geométricas, a serem apresentadas na seção 4.3.7, ainda são geradas com o uso do *transformer*.

Por fim, são geradas as TIC e as geometrias nos 3 níveis de detalhamento são agregadas novamente através do identificador único de cada feição. O arquivo XML final é estruturado de forma automática utilizando o *Writer* específico do modelo CityGML. A Figura 4.6 mostra feições da área de estudo implementadas nesse módulo.

Figura 4.6: Visão das feições do módulo *Building* na área de estudo

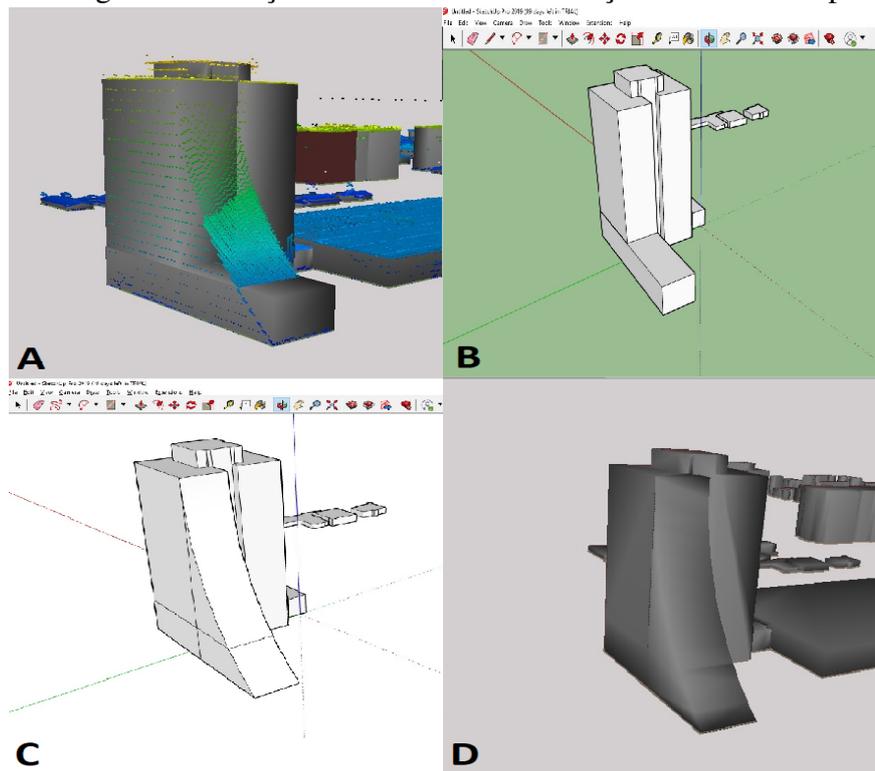
Fonte: O autor.

Algumas edificações que possuam um contorno geométrico particular são generalizadas nesse processo. Esses casos, caso haja interesse e disponibilidade de recursos, podem ser contornados com o uso de plataformas de edição em 3D com suporte para CityGML.

Casos particulares são mostrados nas Figuras 4.7 e 4.8. A plataforma SketchUp¹⁰ é utilizada para exemplificar esse processo. O programa não possui suporte completo nativo para o CityGML, somente realiza a importação. Existem *plugins* proprietários que dão maior suporte nos processos de edição e exportação mas não foram utilizados.

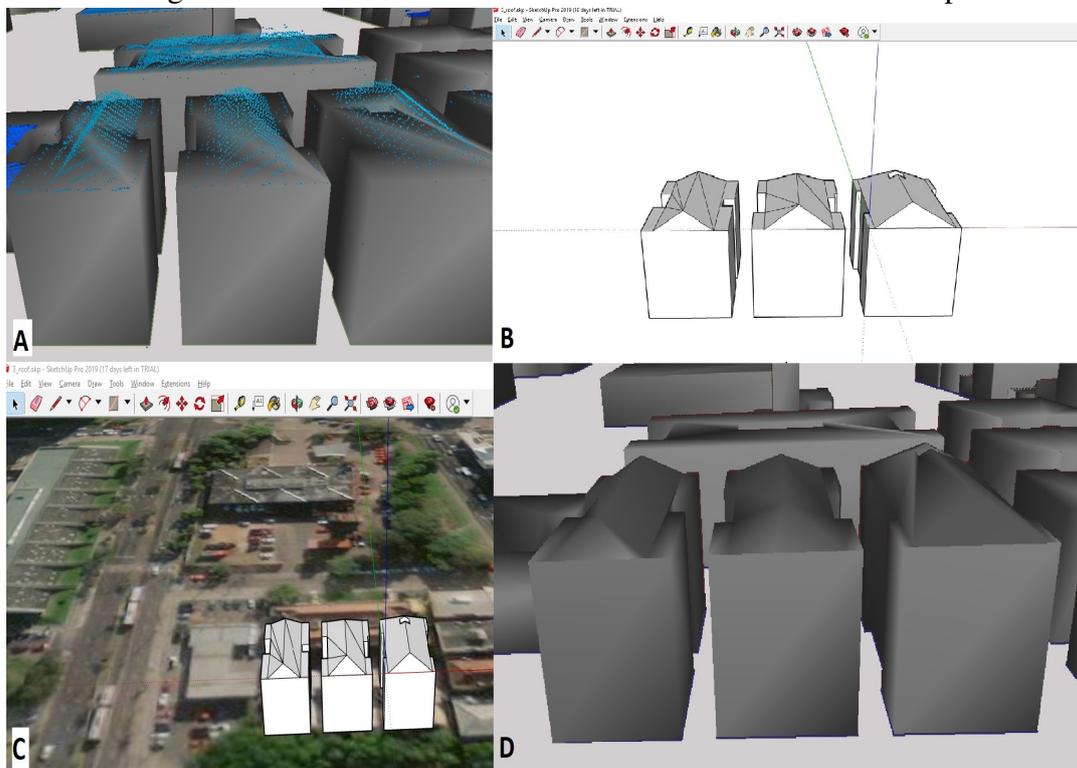
¹⁰<https://www.sketchup.com/>

Figura 4.7: Edição do contorno da edificação com SkechtUp



Fonte: O autor.

Figura 4.8: Detalhamento do formato do telhado com SkechtUp



Fonte: O autor.

Na parte A de ambas figuras, observa-se a diferença da geometria gerada no processo semi-automático e seu real contorno dado pela nuvem de pontos. As geometrias são importadas para o programa (Figura 4.7 B) porém, nesse processo são perdidas as informações do sistema de referência, mantendo-se somente as dimensões originais. São realizadas as edições necessárias com as ferramentas da plataforma (Figura 4.7 C e 4.8 B). Para minimizar a perda do sistema de referência, a plataforma dispõe de uma ferramenta chamada "*Geo-location*", onde é possível posicionar as geometrias sobre uma imagem de satélite de referência (Figura 4.8 C). Dessa forma, ao realizar a exportação no formato KML (*Keyhole Markup Language*), as geometrias recebem as coordenadas do local onde foram posicionadas. Essa etapa introduz erro nas coordenadas originais, pois não se pode obter a precisão do local exato onde as edificações estão posicionadas. Para corrigir os erros, outro *workspace* é criado para reprojetar as coordenadas do arquivo KML gerado na edição para o sistema de referência original utilizando o *transformer reprojector*. A defasagem nas coordenadas é corrigida utilizando o *transformer offsetter*, as geometrias finais são salvas em *citygml* e incorporadas no modelo original, conforme parte D de ambas figuras.

Por se tratar de um processo individualizado, os exemplos dados foram realizados em caráter ilustrativo, não englobando as demais feições da área de estudo.

4.3.2 Módulo *CityFurniture*

Observando o previsto na Figura 3.3, inicialmente é realizada a fase de pré-processamento. As feições da base de dados oficial pertencentes ao módulo *CityFurniture* são originalmente classificadas nos temas *parada de ônibus*, *antena*, *torre de energia*, *torre de telecomunicações*, *muro/grade*, *cerca* e *bueiro*. Com base nas *codelists*¹¹ do módulo *CityFurniture* do CityGML e na classificação temática inicial, foi possível fazer o relacionamento direto para os atributos *CityFurniture_class*, *CityFurniture_function* e *CityFurniture_usage* conforme Tabela 4.5. Como nesse módulo os atributos *function* e *usage* são redundantes, o atributo *usage* foi omitido na tabela.

A base oficial possui na área de estudo 30.542 feições que se enquadram no módulo *CityFurniture*, sendo 119 *paradas de ônibus*, 5.201 *postes* e 653 *cercas*, 24.490 *muro/grades*, 01 *torre de energia*, 13 *torres de telecomunicações*, 23 *antenas* e 37 *buei-*

¹¹Todas *codelists* do módulo *CityFurniture* podem ser encontradas em <<http://www.sig3d.org/codelists/standard/cityfurniture/2.0/>>

Tabela 4.5: Mapeamento dos atributos da base oficial para *codelists* do módulo *CityFurniture*

Atributo oficial	Class	Class_code	Function	Function_code
Parada de ônibus	<i>Traffic</i>	1000	<i>Bus Stop</i>	1110
Poste	<i>Others</i>	1030	<i>Lamp Post</i>	1170
Antena	<i>Communication</i>	1010	<i>Radio Mast</i>	1280
Torre de telecomunicações	<i>Communication</i>	1010	<i>Radio Mast</i>	1280
Torre de energia	<i>Others</i>	1030	<i>Others</i>	1540
Muro/Grade	<i>Security</i>	1020	<i>Fence</i>	1440
Cerca	<i>Security</i>	1020	<i>Fence</i>	1440
Bueiro	<i>Others</i>	1030	<i>Others</i>	1540

Fonte: O autor.

ros.

As feições classificadas no tema *parada de ônibus* que se encontram em quadrantes de cartas distintas precisam ter suas geometrias do tipo linha conectadas para formar polígonos fechados. Não foram encontradas inconsistências na validação inicial das geometrias.

A Tabela 2.2 é usada como referência para definição dos níveis de detalhamento de cada feição. Como não há menção na base oficial sobre a importância de cada feição, todas foram implementadas em LoD2.

A altimetria das feições do tipo polígono é obtida da mesma forma que no módulo *Building*. Já as feições do tipo linha e ponto obtém a altimetria do ponto da nuvem que esteja mais próximo à feição. Essa operação é feita com o *transformer SurfaceDraper*.

As feições do tipo *antena*, *torre de energia* e *torre de telecomunicações* são consideradas de relevância por terem destaque na paisagem do ambiente urbano. Com isso, tais feições passam pelo passo adicional de interseção com o MDS para agregar a altimetria do topo.

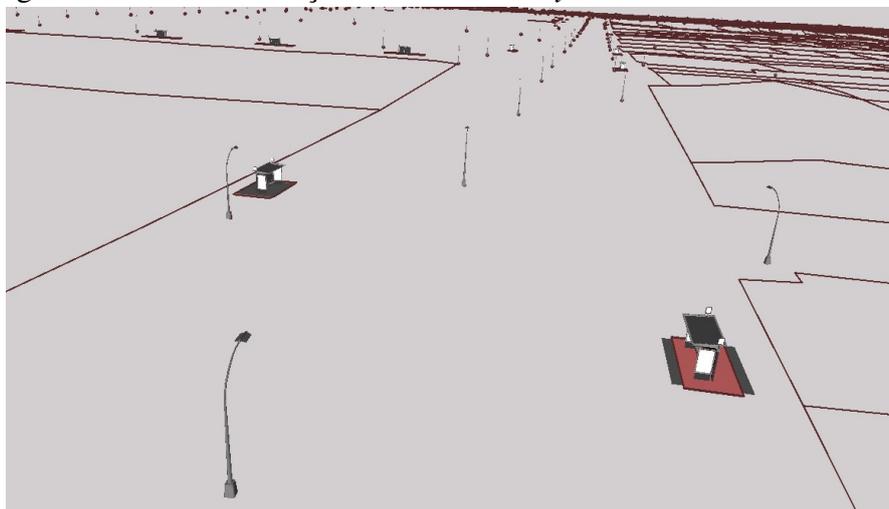
Em seguida são geradas as TIC de cada feição e, visando dar uma estética mais realista ao modelo 3D final, é feita a implementação de geometrias implícitas nos temas *poste*, *torre de energia* e *parada de ônibus* com um modelo 3D pré existente no formato 3DS. Esse modelos podem ser encontrados de forma gratuita em alguns *websites*¹². Dessa forma, a geometria do modelo 3DS é referenciada, tendo como base a coordenada geográfica da feição original.

A orientação da geometria 3DS das feições do tema *parada de ônibus* para que haja alinhamento com o polígono original é feita de forma automática, sendo agregado como atributo o valor do azimute da linha central do polígono original e rotacionando a

¹²O modelo utilizado na pesquisa foi obtido no endereço www.turbosquid.com/

geometria 3DS em torno do eixo Y em uma angulação igual ao valor obtido. Por fim, as feições são agregadas e salvas no formato citygml. A Figura 4.9 mostra um extrato das feições na área de estudo.

Figura 4.9: Visão das feições do módulo *CityFurniture* na área de estudo



Fonte: O autor.

4.3.3 Módulo *Vegetation*

As etapas mostradas na Figura 3.4 são seguidas na implementação do módulo *Vegetation*. A base oficial possui na área de estudo 19.481 feições classificadas no tema *árvores isoladas* e 3.637 classificadas no tema *vegetação*. Assim como outras feições poligonais, as feições classificadas como *vegetação* que se encontram em quadrantes de cartas distintas precisam ter suas geometrias do tipo linha conectadas para formar polígonos fechados. Nesse processo foram identificadas 184 linhas que não formavam polígonos fechados, restando 3.453 feições nessa classe. Essa inconsistência ocorre quando uma feição de outro tema é utilizada como segmento do polígono de vegetação. Esta categoria não possui outros atributos além da classificação inicial que possam ser mapeados para as *codelist* previstas neste módulo do CityGML.

A obtenção da altimetria segue a mesma sistemática do módulo *CityFurniture* porém, nesse caso, todas as feições da classe *árvore isolada* passam pela interseção com o MDS para agregar a altimetria da copa das árvores. Este passo é necessário para que os filtros de nível de detalhamento possam ser aplicados. Tal etapa não foi realizada nas feições da classe *PlantCover* por ter sido optado por implementar as feições com geometria *MultiSurface*.

Tabela 4.6: Atributos e filtros de acordo com o nível de detalhamento do módulo *Vegetation*

Classe	LoD	Filtro de área (m ²)	Filtro de altura	Atributo <i>citygml_lod_name</i>	Feições implementadas
<i>SolitaryVegetationObject</i>	1	-	8	<i>Lod1Geometry</i>	3.542
	2	-	6	<i>Lod2ImplicitRepresentation</i>	5.845
<i>PlantCover</i>	1	50	-	<i>Lod1MultiSurface</i>	1.845
	2	5	-	<i>Lod2MultiSurface</i>	3.142

Fonte: Adaptado de Gröger et al. (2012).

A Tabela 2.2 é usada como referência para definição dos níveis de detalhamento de cada feição. A Tabela 4.6 sintetiza os filtros e preenchimento de atributos por LoD.

Na sequência a implementação de geometrias implícitas é realizada nas feições da classe *SolitaryVegetationObject*. Devido ao elevado número de feições dessa classe nos dados de entrada, o formato 3DS utilizado deve ser simples para que não sobrecarregue as plataformas de visualização.

Por fim, ambas classes são agregadas para formar o módulo *Vegetation*. A Figura 4.10 mostra um extrato das feições na área de estudo.

Figura 4.10: Visão das feições do módulo *Vegetation* na área de estudo



Fonte: O autor.

4.3.4 Módulo *WaterBody*

No módulo *WaterBody*, segue-se o fluxo de trabalho previsto na Figura 3.5. Na base de dados oficial existem 3 classes de dados que, em conjunto, formam a classe *hidrografia*. A primeira contém as feições de lagos/lagos com geometria do tipo polígono. As duas seguintes possuem geometria do tipo linha mas descrevem as feições de uma forma diferente. Em uma, os rios e canais são descritos pelo seu eixo central e na outra,

Tabela 4.7: Mapeamento dos atributos da base oficial para *codelists* do módulo *WaterBody*

Classificação temática	Class	Class_code
Rio perene	Watercourse	1020
Rio intermitente	Ephemeral watercourse	1130
Lago/Lagoa	Lake/Pont	1060
Canal	Canal	1170

Fonte: O autor.

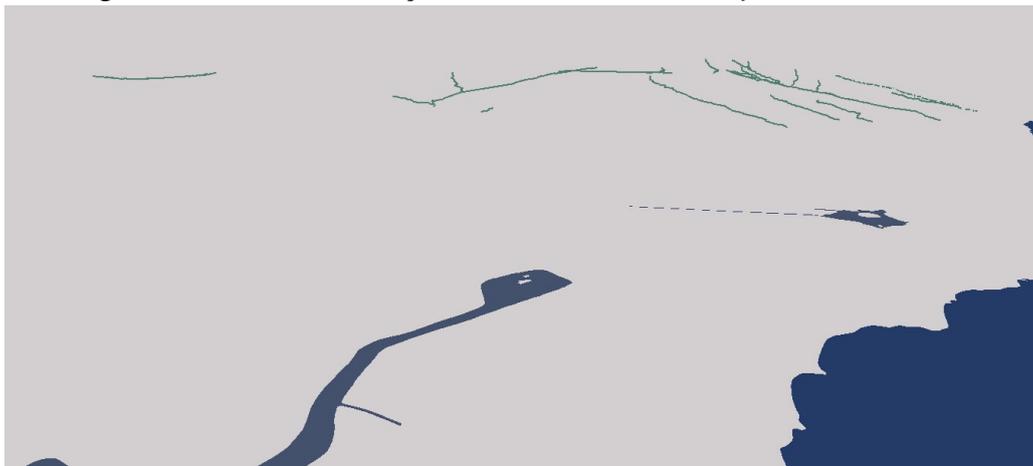
as mesmas feições são descritas pelo contorno das margens. Para possibilitar a implementação das feições com geometria *MultiSurface*, uma etapa manual foi realizada de criação de geometrias do tipo polígono com base na geometria linear das margens. Nessa etapa foi utilizado o *software* QGIS.

Com base na classificação inicial, é feito o mapeamento de atributos para a *codelist* do atributo *WaterBody_class*¹³, conforme Tabela 4.7. Não existem outros atributos além da classificação temática inicial que transmitam mais informações sobre as feições, impossibilitando o mapeamento dos atributos *WaterBody_function*, e *WaterBody_usage* de forma automática.

Após a interseção das feições com o MDT, é utilizado o *transformer triangulator* para realizar a triangulação das superfícies curvas. Conforme mencionado na seção 3.4, os filtros de generalização ficam à critério do modelador. Esta definição é feita adotando o atributo "MARGEM", que possui dois valores: SIMPLES e DUPLA. De acordo com a documentação técnica do órgão produtor dos dados de entrada, são classificados como cursos d'água de margem dupla aqueles que possuem espelhos d'água com largura superior a 1m, sendo consequentemente implementados como superfícies. As demais feições com medida inferior a 1m são classificados como margem simples e implementadas como curvas. Em seguida é extraído o contorno externo das geometrias *MultiSurface* que, em conjunto com as geometrias *MultiCurve*, são passadas como condição de contorno para a triangulação do módulo *Relief*. Por fim, todas geometrias são agregadas para gerar o módulo *WaterBody*. A Figura 4.11 mostra um extrato das feições na área de estudo.

¹³http://www.sig3d.org/codelists/standard/waterbody/2.0/WaterBody_class.xml

Figura 4.11: Visão das feições do módulo *WaterBody* na área de estudo



Fonte: O autor.

4.3.5 Módulo *Transportation*

O *dataset* de entrada do módulo *Transportation* é similar ao do módulo *WaterBody*, pois existe um conjunto de dados que representa as vias pelo seu eixo central e outro com a geometria do contorno externo das vias. O primeiro é mais completo em termos semânticos, com diversos atributos descrevendo as feições, ao passo que o segundo carece de descrição semântica.

Da mesma forma que no módulo *Waterbody*, foi feita uma etapa de digitalização manual para gerar geometrias do tipo polígono com base no *dataset* contendo o contorno externo das vias. Assim, é possível a implementação em LoD1. É importante ressaltar que o uso das ferramentas de aderência do QGIS possibilita que, no processo de digitalização, todos os vértices originais das feições sejam mantidos QGIS (2020). Além disso, nesse processo também foram agregados os atributos do *dataset* de eixos das vias. Devido ao detalhamento dos dados oficiais, no processo de digitalização é possível diferenciar as feições da classe *TrafficArea* das feições da classe *AuxiliaryTrafficArea*.

Neste módulo são feitos dois mapeamentos de atributos distintos. Um com as informações do atributo *CDIDECAT*, que contém a classificação das vias dentro de categorias pré-estabelecidas, para as *codelists*¹⁴ das classes *TransportationComplex* e *TrafficArea*, conforme Tabela 4.8, e outro com as informações da própria classificação temática das feições para o atributo *TrafficArea_surfaceMaterial*, conforme Tabela 4.9. Os me-

¹⁴Todas *codelists* do módulo *Transportation* podem ser encontradas em <<http://www.sig3d.org/codelists/standard/transportation/2.0/>>

Tabela 4.8: Mapeamento dos atributos da base oficial para *codelists* da classe *TransportationComplex* e *TrafficArea*

CDIDECAT	<i>TransportationComplex</i> <i>class_code</i>	<i>TransportationComplex</i> <i>function_code</i>	<i>TrafficArea</i> <i>function_code</i>	<i>TrafficArea</i> <i>usage_code</i>
Acesso (AC)	1010	1280	2	1, 6
Alameda (AL)	1040	1210	1	2, 3, 4, 7, 13
Arroio (AR)	1070	2620	-	-
Avenida (AV)	1040	1210	1	2, 3, 4, 7, 13
Beco (BC)	1010	1220	2	1
Boulevard (BV)	1010	1220	2	1, 6
Cais (CA)	1040	1210	1	2, 3, 4, 7, 13
Ciclovía (CICL)	1010	1240	3	6
Conexão (CN)	1040	1210	1	2, 3, 4, 7, 13
Escadaria (ESC)	1010	1220	22	1
Esplanada (EP)	1010	1220	2	1, 6
Estação (EST)	1010	1920	2	1
Estrada (ESTR)	1040	1000	1	2, 3, 4, 7, 13
Galeria (GAL)	1010	1500	1	1
Linha Exclusiva (LE)	1000	1210	1	2, 3, 4, 7, 13
Largo (LG)	1010	1220	2	1, 6
Mercado (MER)	1010	1620	2	1
Parque (PQ)	1010	1220	2	1
Passagem (PSG)	1010	1220	2	1, 6
Passarela (PSL)	1010	1220	2	1
Praça (PC)	1010	1220	2	1
Rua (R)	1040	1210	1	2, 3, 4, 7, 13
RIO	1070	2622	-	9
Rua Particular (RP)	1000	1220	37	1, 6
Rótula (RTL)	1040	1110	1	2, 3, 4, 7, 13
Terminal (TERM)	1010	1920	2	1
Travessa (TRAV)	1010	1220	2	1, 6
Travessia (TRVS)	1010	1220	2	1, 6
Via de acesso (VA)	1040	1210	1	2, 3, 4, 7, 13
Viaduto (VDT)	1040	1210	1	2, 3, 4, 7, 13
VIA	1040	1210	1	2, 3, 4, 7, 13
Via particular (VP)	1000	1210	37	2, 3, 4, 7, 13
VTC	1010	1220	2	1

Fonte: O autor.

tadados disponibilizados pelo órgão fornecedor facilitam o mapeamento, porém faltam informações sobre o significado dos códigos utilizados para preencher o atributo *CDIDECAT*. O atributo original *NMIDELOG* foi relacionado ao atributo *gml_name*.

No *dataset* de entrada, o único tipo de *AuxiliaryTrafficArea* existente são separadores físicos de faixas de rodagem, recebendo, por tando, o atributo *AuxiliaryTrafficArea_function_code* 1300 ("*traffic island*")

Após a validação inicial, onde não foram encontradas inconsistências, as feições são separadas de acordo com sua geometria. As feições com geometria do tipo linha passam pela interseção com o MDT e são convertidas diretamente para *LoDONetwork*. Nas feições poligonais é aplicado um filtro para identificar as feições elevadas, como viadu-

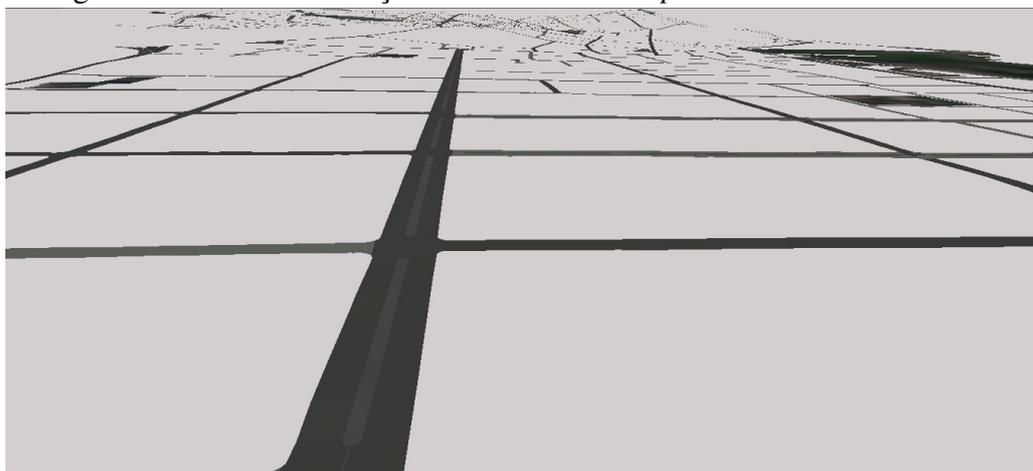
Tabela 4.9: Mapeamento dos atributos da base oficial para o atributo *SurfaceMaterial*

Classificação temática	<i>Surface_Material_code</i>
Pavimentação não asfáltica	4
Não pavimentada	8
Pavimentação asfáltica	1
Escadaria/Rampa	2

Fonte: O autor.

tos, para agregar os dados de altimetria com a interseção com o MDS. As demais feições poligonais recebem os dados de altimetria do MDT. A exemplo do módulo *WaterBody*, é realizada a triangulação nas superfícies geradas visando eliminar as superfícies não planares. Finalmente, todas geometrias são agregadas para gerar o módulo *Transportation*. A Figura 4.12 mostra uma visão geral das feições implementadas em LoD1 nesse módulo.

Figura 4.12: Visão das feições do módulo *Transportation* na área de estudo



Fonte: O autor.

4.3.6 Módulo *LandUse*

A implementação do módulo *LandUse* é mais simples, uma vez que grande parte das geometrias são copiadas de outros módulos. Isso gera duplicação de geometrias, por outro lado, garante a característica de modularização do modelo. Caso o usuário necessite somente do módulo *LandUse* para alguma aplicação, o mesmo pode ser exportado isoladamente. Se as geometrias fossem implementadas com a ferramenta *Xlink*, todos os outros módulos existentes deveriam ser exportados em conjunto. Nesse caso, o modelo perderia a característica de modularização.

Dessa forma, somente as feições que não pertencem a nenhum outro módulo passam pela interseção com o MDT para agregar os dados de altimetria. Após isso, é feita a

Tabela 4.10: Mapeamento dos atributos de outros módulos e feições adicionais para *codelists* do módulo *LandUse*

Atributos de outros módulos/ Feições adicionais	<i>LandUse_class_code</i>	<i>LandUse_function_code</i>
<i>WaterBody_class - 1060</i>	4000	4020
<i>WaterBody_class - 1020</i>	4000	4010
<i>WaterBody_class - 1170</i>	4000	4010
<i>Transportation - Todas</i>	2000	2010
<i>Building_class - 1000</i>	1000	1010
<i>Building_class - 1020</i>	1000	1020
<i>Building_class - 1030</i>	1000	1020
<i>Building_class - 1160</i>	1000	1020
<i>Building_class - 1050</i>	1000	1130
<i>Building_class - 1060</i>	1000	1130
<i>Building_class - demais</i>	1000	1030
<i>Vegetation_class - Todas</i>	3000	3030
Parques	1100	1110
Praças	2000	2060
Quadras esportivas	1000	1130

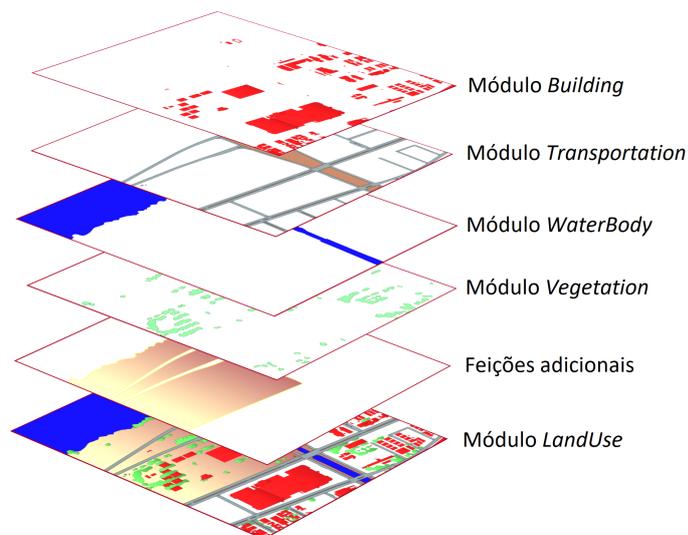
Fonte: O autor.

conversão dos atributos para as *codelists*¹⁵ do módulo *LandUse*. A Tabela 4.10 mostra a conversão dos atributos de outros módulos e das feições adicionais.

Depois do mapeamento de atributos, é realizada a análise de relacionamento espacial entre as camadas. Caso haja sobreposição de superfícies e os atributos de função e uso sejam diferentes, os mesmos são acumulados. Por fim, é feita a hierarquização das feições conforme Figura 4.13 com o auxílio dos transformes *AreaOnAreaOverlayer* e *Clipper*, definindo a classe final da superfície e consolidando o módulo *LandUse*, mostrado na Figura 4.14.

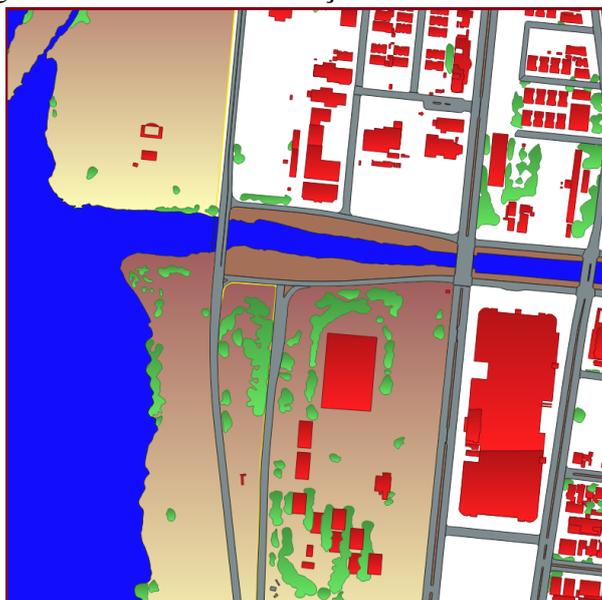
¹⁵As *codelists* do módulo *LandUse* são encontradas em <http://www.sig3d.org/codelists/standard/landuse/2.0/>

Figura 4.13: Hierarquia do módulo *LandUse* para definir o atributo 'classe'.



Fonte: O autor.

Figura 4.14: Amostra de feições do módulo *LandUse*.



Fonte: O autor.

A implementação do módulo *Relief* é direta. Insere-se a geometria da base das feições implementadas nos demais módulos como condições de contorno para a triangulação e a nuvem de pontos do MDT como insumo base no *transformer TINGenerator*. Dessa forma, o modelo final se adequará automaticamente ao contorno das feições, mantendo a integridade topológica entre os módulos.

Visando trazer maior realismo ao modelo final, uma imagem de satélite é utilizada

como insumo de aparência no *transformer AppearanceSetter*. Com isso, a imagem é associada ao modelo para visualização. A Figura 4.15 apresenta uma visão geral do módulo *Relief* implementado.

Figura 4.15: Visão do módulo *Relief* na área de estudo



Fonte: O autor. Imagem: Google

4.3.7 Validação

A validação é realizada em três níveis: semântico, geométrico e de esquema para verificar a integridade do modelo gerado. A análise semântica inclui verificações de plausibilidade de atributos, permitindo apenas atributos compatíveis com a estrutura e tipo de atributo CityGML. A verificação de esquema garante que os dados validados estejam em conformidade com estrutura padrão OGC para CityGML 2.0 e o nível geométrico valida primitivas 3D de acordo com o padrão internacional ISO19107 (Wagner et al., 2013).

A primeira etapa de validação é feita marcando a opção de validação no *Writer CityGML* do FME. Com isso, o CityGML XSD é usado para validar a implementação no nível semântico e de esquema. Exemplos de aspectos semânticos verificados são identificadores únicos duplicados e preenchimento de atributos de acordo com o tipo especificado. Nesta etapa, não foram identificadas inconsistências em nenhum dos módulos implementados.

Dando continuidade à validação, duas ferramentas de código aberto foram usadas: Val3dity (LEDOUX, 2018) e 3DCityDB (YAO et al., 2018).

Val3dity lida com a validação geométrica das feições. Apenas o módulo *Building* apresentou inconsistências geradas na etapa de identificação de superfícies com-

partilhadas. Foram encontrados treze erros do tipo 101 (TOO FEW POINTS) gerados quando uma feição do tipo de linha é criada quando as superfícies sobrepostas são identificadas. Isso ocorre por erro de arredondamento na 4ª casa decimal. A eliminação das feições lineares pode ser feita sem perda de informações. Todos os outros erros estão relacionados à orientação das superfícies. 60,5% das feições apresentaram erros do tipo 302 (SHELL_NOT_CLOSED), 303 (NON_MANIFOLD_CASE) ou 307 (POLYGON_WRONG_ORIENTATION). Para corrigir este tipo de erro, deve-se inverter a ordem dos vértices do polígono indicado, porém o relatório de validação mostra o identificador único do edifício, não da superfície, dificultando a identificação das superfícies para correção. Além disso, em inspeções visuais, verificou-se que outras superfícies que são compartilhadas entre os edifícios não foram identificadas pelo *transformader SurfaceOn-SurfaceOverlayer*. Esta etapa pode ser complementada manualmente com editores 3D com suporte para CityGML, porém não foi incluída do escopo desta pesquisa. O sumário de validação gerado pela ferramenta Val3dity pode ser visualizado na Figura 4.16

Figura 4.16: Sumário da validação geométrica utilizando a ferramenta Val3dity no módulo *Building* em LoD2

```

Windows PowerShell
val3dity Copyright (c) 2011-2020, 3D geoinformation research group, TU Delft
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions; for details run val3dity with the '--license' option.
---
Reading file: AE_Building_sharing_fix.gml
Translating all coordinates by (-276697, -1.67029e+06)
XLinks found, resolving them...done.
CityGML input file
# City Objects found: 19662
Primitive(s) validated: All
(CityGML/CityJSON/IndoorGML have all their 3D primitives validated)
Parameters used for validation:
snap_tol          0.0001
planarity_d2p_tol 0.01
planarity_n_tol   20
overlap_tol       none

Validation of 19662 feature(s):
[=====] 100%

+++++ SUMMARY +++++
Input file type:
CityGML
[watchout: CityGML support is deprecated]
[future version will not support it]
[upgrade to CityJSON]
+++++
Total # of Features:      19662
# valid:                  7767 (39.5%)
# invalid:                11895 (60.5%)
Types:
Building
+++++
Total # of primitives:   19662
# valid:                  7767 (39.5%)
# invalid:                11895 (60.5%)
Types:
Solid
+++++
Errors present:
101 -- TOO_FEW_POINTS
      13 primitive(s)
302 -- SHELL_NOT_CLOSED
      7787 primitive(s)
303 -- NON_MANIFOLD_CASE
      1684 primitive(s)
307 -- POLYGON_WRONG_ORIENTATION
      3900 primitive(s)
+++++

```

Fonte: O autor.

Para contornar as inconsistências encontradas, o módulo *Building* também foi implementado sem a etapa de identificação de superfícies compartilhadas, que é opcional. Não houve inconsistências apresentadas pela ferramenta nesta versão, garantindo a integridade do modelo final.

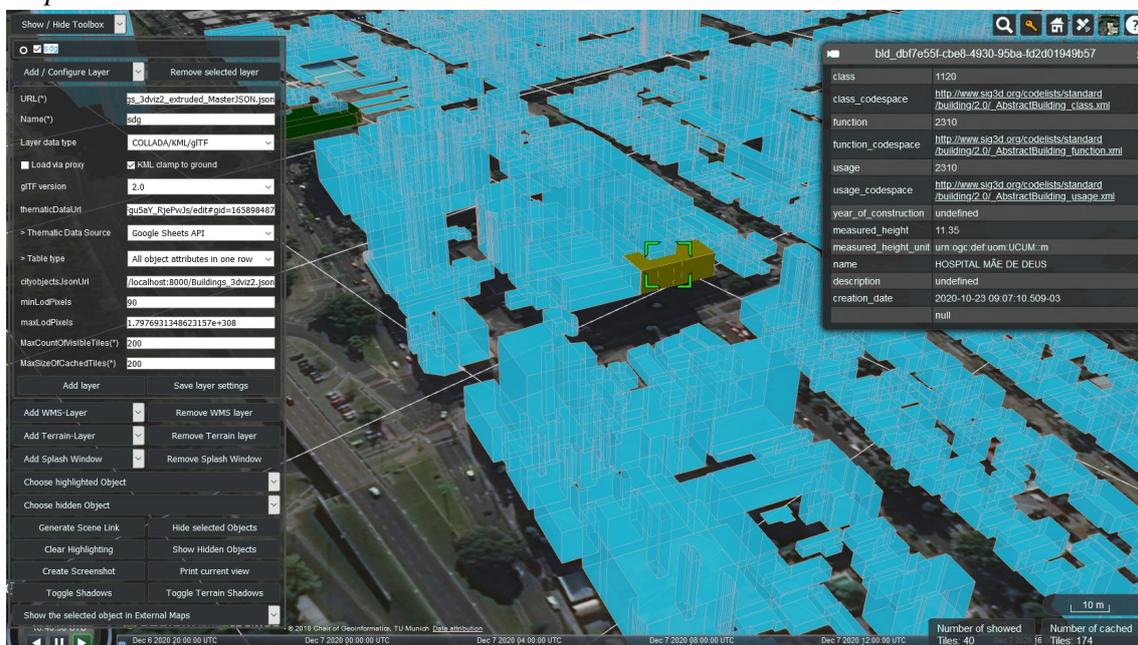
Uma vez gerados, os modelos 3D podem ser armazenados em bancos de dados espaciais, facilitando sua manutenção e gerenciamento. Para isso, foi utilizada a plataforma 3DCityDB¹⁶, uma solução de banco de dados geográficos 3D gratuita para importar, gerenciar, analisar, visualizar e exportar modelos virtuais de cidades em 3D de acordo com o padrão CityGML. Ao carregar o modelo, os dados passam por uma fase de validação do esquema novamente, confirmando a qualidade do modelo final.

O software permite aos usuários exportar dados CityGML nos formatos KML/-

¹⁶<https://www.3dcitydb.org/3dcitydb/>

COLLADA/ glTF para visualização. Esses dados podem ser visualizados em diversos aplicativos como CesiumJS e Google Earth. O 3DCityDB inclui uma ferramenta de interface *web* baseada em Cesium chamada *3DCityDB Web-map-client* para visualização de dados. Esta ferramenta ajuda os usuários a explorar e visualizar modelos de cidades em 3D com dados semânticos. Um servidor simples HTTP baseado em JavaScript acompanha a ferramenta, cujo objetivo principal é testar a funcionalidade localmente. A Figura 4.17 mostra um extrato do modelo gerado exportado no formato KML e visualizado utilizando o *3DCityDB Web-map-client*, permitindo interação com as feições para obter as informações semânticas sobre elas.

Figura 4.17: Visualização do modelo gerado utilizando a ferramenta *3DCityDB Web-map-client*



Fonte: O autor.

4.4 Considerações Finais

Para demonstrar a exequibilidade da abordagem proposta, foi desenvolvido um estudo de caso com dados oficiais 2D e altimétricos disponibilizados pela PMPA e da base do OSM. Foi possível transformar os dados oficiais em um modelo 3D semanticamente "enriquecido" baseado em CityGML. Apesar do elevado grau de associação geométrica entre a base de dados oficial e a base do OSM, houve um baixo percentual de atributos

agregados devido à baixa completude temática da base do OSM na área de teste. Apesar do resultado nessa etapa não ter sido o esperado, a proposta ainda é válida, uma vez que o número de atributos agregados irá variar de acordo com a área de estudo utilizada. A implementação do módulo *Building* com compartilhamento de superfícies apresentou algumas inconsistências geométricas com possível correção automática. Já a implementação sem compartilhamento de superfícies retornou um modelo completamente íntegro.

5 CONCLUSÃO

Este capítulo apresenta as discussões finais relativas a pesquisa, analisando os objetivos alcançados, contribuições, limitações e trabalhos futuros.

O presente trabalho apresentou uma abordagem híbrida para geração semiautomática de modelos semânticos 3D baseados em CityGML a partir de bancos de dados oficiais e "enriquecimento" semântico com integração com banco de dados OSM. Os objetivos específicos previstos foram atingidos da seguinte forma:

- *Desenvolver uma abordagem geral de integração de dados abertos heterogêneos para geração semiautomática de modelos 3D baseados em CityGML:* Os fluxos de trabalho específicos para cada módulo do CityGML apresentados no Capítulo 3 sevem como modelo genérico para alcançar o objetivo proposto, podendo ser modificados de acordo com o contexto de aplicação;
- *Implementar tal abordagem tendo como base dados disponibilizados pela Prefeitura Municipal de Porto Alegre-RS e dados extraídos da base de dados do OSM:* A implementação do estudo de caso foi concluída com algumas restrições apresentadas na seção 4.3.7, mostrando a exequibilidade da abordagem proposta com algumas limitações;
- *Avaliar a abordagem e seu desempenho realizando a validação em diferentes níveis do modelo final gerado:* Três plataformas (FME, Val3dity e 3DCityDB) foram utilizadas para validar o modelo gerado nos níveis de esquema, geométrico e semântico.

Com a conclusão desse trabalho, tem-se como principal contribuição a de auxiliar os administradores municipais que desejam transformar sua bases de dados 2D e iniciar a implementação de um banco de dados 3D seguindo padrões de modelagem internacionais. A abordagem proposta formalizada no formato de fluxos de trabalho não foi encontrada em nenhum dos trabalhos relacionados e viabiliza sua implementação em diferentes contextos. Os *workspaces* do FME implementados, bem como as planilhas usadas no mapeamento de atributos OSM para listas de códigos CityGML, estarão disponíveis em <<https://github.com/mcmaieron/DWGCityGML>>. Os *workspaces* foram projetados em alinhamento com os dados de entrada do estudo de caso, mas pode ser usado em um cenário geral com algumas adaptações, que dependerão das diferentes estruturas de dados de entrada utilizadas.

5.1 Limitações

Devido à falta de informações sobre a hierarquia entre feições nos dados de entrada, não foi possível para usar todo o potencial do CityGML, como a implementação de forma automática das classes *cityGenericGroup*, *BuildingParts* e *BuildingInstallation*. Além disso, as bases originais em ShapeFile não disponibilizam identificadores únicos para cada feição, o que impossibilitou a criação do *link ExternalReference* previsto no CityGML.

As inconsistências apresentadas na seção 4.3.7 mostram as limitações da fases de identificação e referenciamento de superfícies compartilhadas. Ainda assim, a correção dos erros poderiam ser automatizadas caso o relatório apresentasse os identificadores únicos das superfícies juntamente com os identificadores únicos das edificações.

O FME, a principal plataforma utilizada na implementação, é comercial. Com isso, a para que partes interessadas possam utilizar total ou parcialmente os *Workspaces* gerados, é necessário possuir uma licença para uso do programa.

É importante notar que a abordagem proposta é direcionada para dados de entrada 2D agregando informações de altimetria da interseção com DTM e DSM. No entanto, existem diferentes técnicas para gerar geometrias 3D além desta (por exemplo, fotogrametria).

5.2 Trabalhos Futuros

Devido a complexidade da modelagem e implementação do ambiente urbano, são identificados alguns trabalhos futuros visando complementar a presente pesquisa:

- Possível extensão do CityGML para propósitos nacionais, seguindo Brink, Stoter and Zlatanova (2013), podendo ser acrescentadas *codelists* específicas para o contexto de mapeamento do território nacional, baseando-se nas normatização de mapeamento vigente;
- Implementação e aplicação de outros métodos de validação semântica e geométrica, visando robustecer essa etapa do fluxo de trabalho.
- Identificação automática do formato do telhado a partir de nuvem de pontos, visando trazer o contorno das feições mais próximos à realidade e possibilitando resultados mais acurados de aplicações como potencial energético solar;

- Implementação dos módulos *Bridge* e *Tunnel* automaticamente.

REFERÊNCIAS

- AGUGIARO, G. From sub-optimal datasets to a CityGML-compliant 3D city model: experiences from Trento, Italy. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, XL-4, p. 7–13, abr. 2014. ISSN 2194-9034. Available from Internet: <<http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-4/7/2014/>>.
- AGUGIARO, G. FIRST STEPS TOWARDS AN INTEGRATED CITYGML-BASED 3D MODEL OF VIENNA. **ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences**, III-4, p. 139–146, jun. 2016. ISSN 2194-9050. Available from Internet: <<http://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/III-4/139/2016/isprs-annals-III-4-139-2016.pdf>>.
- BASTANI, F. et al. **RoadTracer: Automatic Extraction of Road Networks from Aerial Images**. 2018.
- BILJECKI, F. et al. Applications of 3D City Models: State of the Art Review. **ISPRS International Journal of Geo-Information**, v. 4, n. 4, p. 2842–2889, dec. 2015. ISSN 2220-9964. Available from Internet: <<http://www.mdpi.com/2220-9964/4/4/2842>>.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **The unified modeling language user guide**. Reading Mass: Addison-Wesley, 1999. (The Addison-Wesley object technology series). ISBN 978-0-201-57168-4.
- BRASIL. **DECRETO Nº 89.817, DE 20 DE JUNHO DE 1984. Estabelece as Instruções Reguladoras das Normas Técnicas da Cartografia Nacional**. Brasília, DF, 1984. Available from Internet: <http://www.planalto.gov.br/ccivil_03/decreto/1980-1989/d89817.htm>.
- BRASIL. **DECRETO Nº 6.666, DE 27 DE NOVEMBRO DE 2008. Institui, no âmbito do Poder Executivo federal, a Infra-Estrutura Nacional de Dados Espaciais - INDE, e dá outras providências**. Brasília, DF, 2008. Available from Internet: <http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/decreto/d6666.htm>.
- BRINK, L. van den; STOTER, J.; ZLATANOVA, S. Establishing a national standard for 3D topographic data compliant to CityGML. **International Journal of Geographical Information Science**, v. 27, n. 1, p. 92–113, jan. 2013. ISSN 1365-8816, 1362-3087. Available from Internet: <<http://www.tandfonline.com/doi/abs/10.1080/13658816.2012.667105>>.
- BUYUKDEMIRCI OGLU, M.; KOCAMAN, S.; ISIKDAG, U. Semi-Automatic 3D City Model Generation from Large-Format Aerial Images. **ISPRS International Journal of Geo-Information**, v. 7, n. 9, p. 339, aug. 2018. ISSN 2220-9964. Available from Internet: <<http://www.mdpi.com/2220-9964/7/9/339>>.
- CARRILHO, A. **Aplicação de técnicas de processamento e análise de imagens para detecção de edificações e vegetação a partir de dados LiDAR**. 2016. 72 p.
- CONSÓRCIO GUAÍBA. **RELATÓRIO DE PLANEJAMENTO DA ELABORAÇÃO DA BASE CARTOGRÁFICA**. Porto Alegre, Brasil, 2011. 122 p.

COTE, P. **Formats for Geographic Data**. 2020. Available from Internet: <http://www.gismanual.com/data_structures/>.

DELFT UNIVERSITY OF TECHNOLOGY. **Cities/regions around the world with open datasets**. 2020. Available from Internet: <<https://3d.bk.tudelft.nl/opendata/opencities/>>.

ENVIRONMENTAL SYSTEMS RESEARCH INSTITUTE. **How To: Determine which NAD_1983_To_WGS_1984 transformation to use**. 2020. Available from Internet: <<https://support.esri.com/en/technical-article/000005929>>.

FONSECA, F. et al. Semantic granularity in ontology-driven geographic information systems. **AMAI Annals of Mathematics and Artificial Intelligence**, v. 36, p. 121–151, 09 2002.

FONTE, C. et al. Assessing vgi data quality. In: _____. Ubiquity Press, 2017. p. 137–163. ISBN EPUB 978-1-911529-18-7. Available from Internet: <<http://www.jstor.org/stable/j.ctv3t5qzc>>.

GOETZ, M. Towards generating highly detailed 3D CityGML models from OpenStreetMap. **International Journal of Geographical Information Science**, v. 27, n. 5, p. 845–865, may 2013. ISSN 1365-8816, 1362-3087. Available from Internet: <<http://www.tandfonline.com/doi/abs/10.1080/13658816.2012.721552>>.

GOETZ, M.; ZIPF, A. Towards Defining a Framework for the Automatic Derivation of 3D CityGML Models from Volunteered Geographic Information:. **International Journal of 3-D Information Modeling**, v. 1, n. 2, p. 1–16, abr. 2012. ISSN 2156-1710, 2156-1702. Available from Internet: <<http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/ij3dim.2012040101>>.

GRÖGER, G. et al. OGC City Geography Markup Language (CityGML) Encoding Standard. p. 344, 2012.

GRÖGER, G.; PLÜMER, L. CityGML – Interoperable semantic 3D city models. **ISPRS Journal of Photogrammetry and Remote Sensing**, v. 71, p. 12–33, jul. 2012. ISSN 09242716. Available from Internet: <<https://linkinghub.elsevier.com/retrieve/pii/S0924271612000779>>.

HU, J. et al. Road network extraction and intersection detection from aerial images by tracking road footprints. **IEEE Transactions on Geoscience and Remote Sensing**, v. 45, n. 12, p. 4144–4157, 2007.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Panorama Porto Alegre**. 2020. (acessado em novembro, 2020). Available from Internet: <<https://cidades.ibge.gov.br/brasil/rs/porto-alegre/panorama>>.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **Geographic information — Data quality**. Geneva, CH, 2013. Available from Internet: <<https://www.iso.org/standard/32575.html>>.

JANEČKA, K. Transformation of 3D geospatial data into CityGML – a case of Prague. **Reports on Geodesy and Geoinformatics**, v. 107, n. 1, p. 41–48, may 2019.

ISSN 2391-8152. Available from Internet: <<https://content.sciendo.com/doi/10.2478/rgg-2019-0005>>.

KHATOON, R.; ZEADALLY, S. Smart cities: concepts, architectures, research opportunities. **Communications of the ACM**, v. 59, n. 8, p. 46–57, jul. 2016. ISSN 0001-0782, 1557-7317. Available from Internet: <<https://dl.acm.org/doi/10.1145/2858789>>.

KILIC, B.; GÜLGEN, F. Investigating the quality of reverse geocoding services using text similarity techniques and logistic regression analysis. **Cartography and Geographic Information Science**, Taylor & Francis, v. 47, n. 4, p. 336–349, 2020. Available from Internet: <<https://doi.org/10.1080/15230406.2020.1746198>>.

KIM, T. J. Metadata for geo-spatial data sharing: A comparative analysis. **The Annals of Regional Science**, v. 33, n. 2, p. 171–181, may 1999. ISSN 0570-1864, 1432-0592. Available from Internet: <<http://link.springer.com/10.1007/s001680050099>>.

KOLBE, T. H.; BURGER, B.; CANTZLER, B. Citygml goes to broadway. In: FRITSCH, D. (Ed.). **Photogrammetric Week '15**. Stuttgart: Wichmann, 2015. p. 343–356. ISBN 978-3-87907-587-4. Project page (incl. data downloads): <http://www.gis.bgu.tum.de/projekte/new-york-city-3d/> Book: <https://www.vde-verlag.de/buecher/537587/photogrammetric-week-15.html>. Available from Internet: <<http://www.ifp.uni-stuttgart.de/publications/phowo15/330Kolbe.pdf>>;>.

KOUNADI, O. et al. Accuracy and privacy aspects in free online reverse geocoding services. **Cartography and Geographic Information Science**, Taylor & Francis, v. 40, n. 2, p. 140–153, 2013. Available from Internet: <<https://doi.org/10.1080/15230406.2013.777138>>.

KUFONIYI, O. **Spatial coincidence modelling, automated database updating, and data consistency in vector GIS**. Enschede: International Institute for Aerospace Survey and Earth Sciences, 1995. (ITC publication, no. 28).

LAM, N. S.-N.; QUATTROCHI, D. A. On the issues of scale, resolution, and fractal analysis in the mapping sciences. **The Professional Geographer**, Routledge, v. 44, n. 1, p. 88–98, 1992. Available from Internet: <<https://doi.org/10.1111/j.0033-0124.1992.00088.x>>.

LAURINI, R. Spatial multi-database topological continuity and indexing: A step towards seamless gis data interoperability. **International Journal of Geographical Information Science**, v. 12, p. 373–402, 06 1998.

LEDOUX, H. val3dity: validation of 3D GIS primitives according to the international standards. **Open Geospatial Data, Software and Standards**, v. 3, n. 1, feb. 2018. ISSN 2363-7501. Available from Internet: <<https://doi.org/10.1186/s40965-018-0043-x>>.

LENZERINI, M. Data integration: A theoretical perspective. In: **Proceedings of the Twentieth ACM SIGMOD SIGACT SIGART Symposium on Principles of Database Systems**. New York, NY, USA: Association for Computing Machinery, 2001. p. 233–246. ISBN 1581133618.

LOTT, R. **The OGC® Abstract Specification**. [S.l.], 2010. 100 p.

MALAMBO, L.; HAHN, M. Lidar assisted citygml creation. In: **Applied Geoinformatics for Society and Environment: AGSE 2010 ; [the Third Applied Geoinformatics for Society and Environment Summer School-cum-Seminar] = Geoinformática Aplicada para la Sociedad y el Medio Ambiente**. Stuttgart: [s.n.], 2010. (Veröffentlichungen der Hochschule für Technik Stuttgart), p. 31–37. ISBN 978-3-940670-19-9.

MOHAMMADI, H. et al. Spatial data integration. In: **Proceedings of the Seventeenth UN Regional Cartographic Conference for Asia and the Pacific**. United Nations, 2006. ISBN 978-9217300936. Available from Internet: <<http://hdl.handle.net/11343/26704>>.

MONTECINO, H. D.; SANTIBANEZ, S.; FREITAS, S. D. Influence of errors in coordinate transformation due to uncertainties of the design matrix. **Applied Geomatics**, v. 5, 12 2013.

MOREIA, J. M. M. et al. From dsm to 3d building models: A quantitative evaluation. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, XL-1/W1, p. 213–219, 2013. Available from Internet: <<https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-1-W1/213/2013/>>.

OPEN GEOSPATIAL CONSORTIUM. **GeoPackage Encoding Standard**. 2020. Available from Internet: <<https://www.ogc.org/standards/geopackage>>.

OPENSTREETMAPS. **Map Features**. 2019. Available from Internet: <https://wiki.openstreetmap.org/wiki/Map_Features>.

ORGANIZATION FOR THE ADVANCEMENT OF STRUCTURED INFORMATION STANDARDS. **Extensible Address Language (xAL) Standard Description Document for W3C DTD/Schema, Version 2.0**. [S.l.], 2002. 90 p. Available from Internet: <https://www.immagic.com/eLibrary/ARCHIVES/TECH/OASIS/XAL_V2.PDF>.

PRANDI, F. et al. USING CITYGML TO DEPLOY SMART-CITY SERVICES FOR URBAN ECOSYSTEMS. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, XL-4/W1, p. 87–92, may 2013. ISSN 1682-1777. Available from Internet: <<http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-4-W1/87/2013/>>.

QGIS PROJECT. **Setting the snapping tolerance and search radius**. 2020. Available from Internet: <https://docs.qgis.org/3.16/en/docs/user_manual/working_with_vector/editing_geometry_attributes.html#setting-the-snapping-tolerance-and-search-radius>.

SAFE SOFTWARE. **Spatial Relations Defined**. 2020. Available from Internet: <http://docs.safe.com/fme/2020.2/html/FME_Desktop_Documentation/FME_Transformers/Transformers/spatialrelations.htm#DE9IM_Matrix>.

SECRETARIA DO MEIO AMBIENTE E DA SUSTENTABILIDADE - PREFEITURA DE PORTO ALEGRE. **CARTAS DE 2010**. 2020. Available from Internet: <<https://www.google.com/maps/d/viewer?mid=14bprojDayiljOUD360Z8SJT2JVYVMZ4H8&usp=sharing>>.

SECRETARIA DO MEIO AMBIENTE E DA SUSTENTABILIDADE - PREFEITURA DE PORTO ALEGRE. **Eixos de Logradouros**. 2020. Available from Internet: <http://lproweb.procempa.com.br/pmpa/prefpoa/spm/usu_doc/eixoslogradouros.zip>.

SECRETARIA DO MEIO AMBIENTE E DA SUSTENTABILIDADE - PREFEITURA DE PORTO ALEGRE. **Mapas Digitais Oficiais**. 2020. Available from Internet: <http://www2.portoalegre.rs.gov.br/spm/default.php?p_secao=310>.

SENGUL, A. **EXTRACTING SEMANTIC BUILDING MODELS FROM AERIAL STEREO IMAGES AND CONVERSION TO CITYGML**. Thesis (Dissertação de mestrado) — İSTANBUL TECHNICAL UNIVERSITY - INSTITUTE OF SCIENCE AND TECHNOLOGY, İSTANBUL, may 2010. Available from Internet: <http://www.citygmlwiki.org/images/3/30/MScThesis_Ahmet_SENGUL.pdf>.

SOON, K. H.; KHOO, V. H. S. CITYGML MODELLING FOR SINGAPORE 3D NATIONAL MAPPING. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, XLII-4/W7, p. 37–42, oct. 2017. ISSN 2194-9034. Available from Internet: <<https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-4-W7/37/2017/>>.

STADLER, A. et al. Making Interoperability Persistent: A 3D Geo Database Based on CityGML. In: LEE, J.; ZLATANOVA, S. (Ed.). **3D Geo-Information Sciences**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 175–192. ISBN 978-3-540-87394-5 978-3-540-87395-2. ISSN: 1863-2246, 1863-2351 Series Title: Lecture Notes in Geoinformation and Cartography. Available from Internet: <http://link.springer.com/10.1007/978-3-540-87395-2_11>.

STOTER, J. et al. Towards sustainable and clean 3D Geoinformation. VDE Verlag, p. 13, 2016. Available from Internet: <<http://resolver.tudelft.nl/uuid:750cab40-386f-4cdc-9182-4dcabe0e8ad1>>.

THEOBALD, D. M. Topology revisited: representing spatial relations. **International Journal of Geographical Information Science**, Taylor & Francis, v. 15, n. 8, p. 689–705, 2001. Available from Internet: <<https://doi.org/10.1080/13658810110074519>>.

UNITED NATIONS. The World's cities in 2018 : data booklet. UN, [New York], p. 29, 2018. Available online (viewed 11 Apr. 2019). Available from Internet: <<http://digitallibrary.un.org/record/3799524>>.

USERY, E.; FINN, M.; STARBUCK, M. Data layer integration for the national map of the united states. **Cartographic Perspectives**, v. 0, 03 2009.

WANG, W. et al. A review of road extraction from remote sensing images. **Journal of Traffic and Transportation Engineering (English Edition)**, v. 3, n. 3, p. 271 – 282, 2016. ISSN 2095-7564. Available from Internet: <<http://www.sciencedirect.com/science/article/pii/S2095756416301076>>.

YAN, J. et al. Integration of 3D objects and terrain for 3D modelling supporting the digital twin. **ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences**, IV-4/W8, p. 147–154, sep. 2019. ISSN 2194-9050. Available from Internet: <<https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-4-W8/147/2019/>>.

YAO, Z. et al. 3DCityDB - a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. **Open Geospatial Data, Software and Standards**, v. 3, n. 1, p. 5, dec. 2018. ISSN 2363-7501.

Available from Internet: <<https://opengeospatialdata.springeropen.com/articles/10.1186/s40965-018-0046-7>>.