



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA CARTOGRÁFICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS GEODÉSICAS E
TECNOLOGIAS DA GEOINFORMAÇÃO

LEANDRO LUIZ SILVA DE FRANÇA

**PROCESSO DE VALIDAÇÃO DA CONSISTÊNCIA LÓGICA DE CONJUNTO DE
DADOS GEOESPACIAIS VETORIAIS COM ABORDAGEM DA CONSISTÊNCIA
SEMÂNTICA**

Recife
2021

LEANDRO LUIZ SILVA DE FRANÇA

**PROCESSO DE VALIDAÇÃO DA CONSISTÊNCIA LÓGICA DE CONJUNTO DE
DADOS GEOESPACIAIS VETORIAIS COM ABORDAGEM DA CONSISTÊNCIA
SEMÂNTICA**

Dissertação apresentada ao Programa de Pós-Graduação em Ciências Geodésicas e Tecnologias da Geoinformação da Universidade Federal de Pernambuco, como parte dos requisitos para obtenção do título de Mestre em Ciências Geodésicas e Tecnologias da Geoinformação.

Área de concentração: Ciências Geodésicas e Tecnologias da Geoinformação.

Orientador: Prof. Dr. José Luiz Portugal.

Recife

2021

Catálogo na fonte
Bibliotecária Sandra Maria Neri Santiago, CRB-4 / 1267

F814p França, Leandro Luiz Silva de.
Processo de validação da consistência lógica de conjunto de dados geoespaciais vetoriais com abordagem da consistência semântica / Leandro Luiz Silva de França. – Recife, 2021.
125 f. : il., figs., quads., tabs.

Orientador: Prof. Dr. José Luiz Portugal.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Ciências Geodésicas e Tecnologias da Geoinformação, 2021.
Inclui referências e apêndices.

1. Engenharia cartográfica. 2. Consistência lógica. 3. Controle de qualidade. 4. Topologia. 5. Validação. I. Portugal, José Luiz (Orientador). II. Título

UFPE

526.1 CDD (22. ed.)

BCTG/2022-25

LEANDRO LUIZ SILVA DE FRANÇA

**PROCESSO DE VALIDAÇÃO DA CONSISTÊNCIA LÓGICA DE CONJUNTO DE
DADOS GEOESPACIAIS VETORIAIS COM ABORDAGEM DA CONSISTÊNCIA
SEMÂNTICA**

Dissertação apresentada ao Programa de Pós-Graduação em Ciências Geodésicas e Tecnologias da Geoinformação da Universidade Federal de Pernambuco, Centro de Tecnologia e Geociências como parte dos requisitos para obtenção do título de Mestre em Ciências Geodésicas e Tecnologias da Geoinformação. Área de concentração: Ciências Geodésicas e Tecnologias da Geoinformação.

Aprovada em: 07 / 07 / 2021.

BANCA EXAMINADORA

Prof. Dr. José Luiz Portugal (Orientador)
Universidade Federal de Pernambuco

Profa. Dra. Andrea Flávia Tenório Carneiro (Examinadora Interna)
Universidade Federal de Pernambuco

Prof. Dr. Ivanildo Barbosa (Examinador Externo)
Instituto Militar de Engenharia

Dedico este trabalho a minha esposa **Mairla** e filho **Lael**.

AGRADECIMENTOS

Primeiramente, ao Senhor Deus, Criador dos céus e da Terra, pela sua imensa bondade e misericórdia. Mesmo em momentos difíceis, Ele sempre esteve ao meu lado para me guiar e colocar as pessoas certas em meu caminho.

Aos meus pais e familiares, pelo carinho e por compreender os momentos de ausência durante esse período de mestrado, além da força e apoio que recebi através de suas orações.

Ao meu orientador, Prof. Dr. José Luiz Portugal, pela confiança, paciência e compreensão. Sou sobremaneira grato por todo o conhecimento e experiência compartilhada no direcionamento e elaboração desta dissertação.

Aos professores que fizeram parte das bancas de Projeto, Qualificação e Dissertação: Profa. Dra. Andrea Flávia Tenório Carneiro, Prof. Dr. Ivanildo Barbosa e Profa. Dra. Simone Sayuri Sato, pelas sugestões de melhorias, somando sabedoria e qualidade a este trabalho.

À Profa. Dra. Andréa de Seixas, Coordenadora do Programa de Pós-Graduação em Ciências Geodésicas e Tecnologias da Geoinformação, pelos inúmeros favores atendidos e por não permitir que eu desistisse durante o período que encontrei dificuldade para realizar este mestrado.

Ao Prof. Dr. Francisco Jaime Bezerra Mendonça e a Profa. Dra. Lucilene Antunes Correia Marques de Sá pela enorme compreensão no processo de conclusão de suas disciplinas.

Aos profissionais da Diretoria de Serviço Geográfico que participaram do Projeto de Mapeamento do Estado da Bahia para a construção do conjunto de dados utilizados nesta pesquisa.

Enfim, a todos aqueles que de alguma forma contribuíram para a realização deste trabalho.

Meu muito obrigado!

RESUMO

Para corretas análises e tomadas de decisões baseadas em dados geoespaciais, é essencial que esses dados tenham sua confiabilidade e integridade atestadas. No que se refere à Consistência Lógica, ainda são poucas as normas e pesquisas que tratem do assunto de forma completa e sistemática, não havendo um consenso sobre os procedimentos adequados para o controle de qualidade desta categoria. Conseqüentemente, alguns produtores de dados geoespaciais apresentam diversas dificuldades para garantir a integridade e consistência dos seus produtos cartográficos devido à verificação inadequada dos relacionamentos lógicos, ocasionando retrabalhos com verificações repetidas e pouco eficientes. Esta dissertação, portanto, teve como finalidade explorar e aprofundar os conceitos inerentes aos elementos de qualidade da consistência lógica, sistematizando e propondo uma sequência otimizada de avaliação para um Conjunto de Dados Geoespaciais Vetoriais (CDGV) de forma a garantir a consistência das informações extraídas desses dados pelos diversos usuários e analistas de SIG. Além disso, foi discutido neste trabalho o conceito de Consistência Semântica, um importante elemento de qualidade que possibilita a verificação automática de regras lógicas entre atributos ou entre atributo e topologia. As regras lógicas elencadas nesta pesquisa foram aplicadas em um CDGV do Projeto de Mapeamento do estado da Bahia, detalhando-se todo o procedimento metodológico para a validação topológica e semântica desses dados. Dos resultados, verificou-se uma quantidade considerável de inconsistências topológicas e semânticas, indicando a necessidade de modernização dos procedimentos de validação pelos produtores de dados geoespaciais. No que se refere às contribuições deste trabalho, além da compilação de uma base teórica abrangente e sistematização do fluxo dos procedimentos de avaliação da consistência lógica de um CDGV, destaca-se também o desenvolvimento de soluções para a verificação das inconsistências topológicas e semânticas com consultas SQL baseadas no software livre PostgreSQL e sua extensão espacial PostGIS.

Palavras-chave: consistência lógica; controle de qualidade; topologia. validação.

ABSTRACT

For correct analysis and decision making based on geospatial data, it is essential that this data has its reliability and integrity attested. With regard to Logical Consistency, there are still few standards and research that address the subject in a complete and systematic way, and there is no consensus on the proper procedures for quality control in this category. Consequently, some geospatial data producers have several difficulties to ensure the integrity and consistency of their cartographic products due to inadequate verification of logical relationships, causing rework with repeated and inefficient verifications. This dissertation, therefore, aimed to explore and deepen the concepts inherent to the quality elements of logical consistency, systematizing and proposing an optimized sequence of evaluation (validation) for a Geospatial Vector Data Set (*Conjunto de Dados Geoespaciais Vetoriais* - CDGV) in order to guarantee the consistency of the information extracted from this data by GIS users and analysts. In addition, the concept of Semantic Consistency was discussed in this work, an important quality element that enables the automatic verification of logical rules between attributes or between attribute and topology. The validation of this research took place in a case study for a CDGV in Brazilian State of Bahia Mapping Project, detailing the entire methodological procedure for the topological and semantic validation of these data. From the results, there was a considerable number of topological and semantic inconsistencies, indicating the need for modernization of validation procedures by geospatial data producers. Regarding the contributions of this work, in addition to the compilation of a comprehensive theoretical basis and systematization of the flow of procedures for evaluating the logical consistency of a CDGV, the development of solutions for the verification of topological and semantic inconsistencies with queries is also highlighted. SQL based on the open-source software PostgreSQL and its spatial extension PostGIS.

Keywords: logical consistency; quality control; topology; validation.

SUMÁRIO

1	INTRODUÇÃO	11
1.1	OBJETIVO GERAL	13
1.2	OBJETIVOS ESPECÍFICOS	13
1.3	ESTRUTURAÇÃO DA DISSERTAÇÃO	14
2	ESPECIFICAÇÕES TÉCNICAS PARA DADOS GEOESPACIAIS VETORIAIS	16
2.1	ISO 19.157:2013	16
2.2	OPEN GEOSPATIAL CONSORTIUM (OGC)	20
3	CONSISTÊNCIA LÓGICA	25
3.1	CONSISTÊNCIA CONCEITUAL	25
3.2	CONSISTÊNCIA DE DOMÍNIO	33
3.3	CONSISTÊNCIA DE FORMATO	40
3.4	CONSISTÊNCIA TOPOLÓGICA	42
3.4.1	Estrutura de Dados Vetoriais	43
3.4.1.1	Estrutura não-topológica	43
3.4.1.2	Estrutura Topológica	44
3.4.1.3	Comparação entre estruturas não-topológicas e topológicas	45
3.4.2	Relacionamento Topológico com uso da matriz DE-9IM	47
3.4.3	Validação Topológica	53
3.4.3.1	Validação da Geometria	55
3.4.3.1.1	<i>Geometria Vazia</i>	55
3.4.3.1.2	<i>Geometria não-simples</i>	55
3.4.3.1.3	<i>Geometria inválida</i>	56
3.4.3.1.4	<i>Geometria Limpa</i>	57
3.4.3.1.5	<i>Geometria multiparte</i>	59
3.4.3.1.6	<i>Geometria com ângulos pequenos</i>	60
3.4.3.1.7	<i>Geometria fora das medidas</i>	61
3.4.3.1.8	<i>Fechamento de anel linear</i>	62
3.4.3.2	Validação intraclasse (interna)	63
3.4.3.2.1	<i>Feições com geometrias duplicadas (repetidas)</i>	63
3.4.3.2.2	<i>Sobreposição entre geometrias</i>	64

3.4.3.2.3	<i>Cruzamento entre Linhas</i>	65
3.4.3.2.4	<i>Lacunas entre polígonos</i>	66
3.4.3.2.5	<i>Feição retalhada</i>	67
3.4.3.2.6	<i>Ponta de linha solta</i>	67
3.4.3.2.7	<i>Ausência de ponto de conectividade</i>	68
3.4.3.2.8	<i>Estrutura de rede</i>	69
3.4.3.3	Validação interclasses (externa)	71
4	CONSISTÊNCIA SEMÂNTICA	76
4.1	REGRAS ENTRE ATRIBUTOS	77
4.2	REGRAS ENTRE ATRIBUTOS E TOPOLOGIA	78
4.3	LIDANDO COM EXCEÇÕES	78
5	AVALIAÇÃO DA CONSISTÊNCIA LÓGICA	80
5.1	MEDIDAS DE QUALIDADE PARA A CONSISTÊNCIA LÓGICA	80
5.2	PROCEDIMENTO DE AVALIAÇÃO DE QUALIDADE	83
5.3	RELATÓRIO DE QUALIDADE	84
6	METODOLOGIA DA PESQUISA	87
6.1	ÁREA DE ESTUDO	87
6.2	DADOS UTILIZADOS	88
6.3	PROCEDIMENTO METODOLÓGICO	89
7	RESULTADOS E DISCUSSÕES	94
7.1	INCONSISTÊNCIAS TOPOLÓGICAS INTERCLASSES	94
7.2	INCONSISTÊNCIAS SEMÂNTICAS ENTRE ATRIBUTOS	99
7.3	INCONSISTÊNCIAS SEMÂNTICAS ENTRE ATRIBUTO E TOPOLOGIA	102
8	CONCLUSÕES E RECOMENDAÇÕES	104
	REFERÊNCIAS	106
	APÊNDICE A - MEDIDA DE QUALIDADE PARA VALIDAÇÃO INTERCLASSES	112
	APÊNDICE B - MEDIDA DE QUALIDADE PARA VALIDAÇÃO DE REGRAS ENTRE ATRIBUTOS	114
	APÊNDICE C - MEDIDA DE QUALIDADE PARA VALIDAÇÃO DE REGRAS ENTRE ATRIBUTO E TOPOLOGIA	116
	APÊNDICE D - SQL PARA INCONSISTÊNCIAS	117

TOPOLÓGICAS**APÊNDICE E – SQL PARA INCONSISTÊNCIAS SEMÂNTICAS 121****ENTRE ATRIBUTOS****APÊNDICE F – SQL PARA INCONSISTÊNCIAS SEMÂNTICAS 124****ENTRE ATRIBUTO E TOPOLOGIA**

1 INTRODUÇÃO

A cada dia se torna mais comum que entidades governamentais contratem empresas ou firmem convênios para a construção ou atualização da sua base cartográfica digital, a qual é indispensável para o controle ambiental, planejamento urbano e gestão territorial de sua área de responsabilidade.

Entretanto, nos contratos para os serviços de mapeamento topográfico, pouco é especificado sobre os critérios de avaliação da qualidade dos dados geoespaciais, principalmente no que tange a consistência lógica.

A consistência lógica, objeto de estudo deste trabalho, é definida pela ISO 19.157 (2013) como o grau de aderência a regras lógicas da estrutura dos dados, atribuições e relacionamentos (seja na estrutura conceitual, lógica ou física).

As regras lógicas devem ser documentadas, por exemplo, nas especificações do produto ou no dicionário de dados geoespaciais, além de ser referenciada no relatório de avaliação da qualidade.

Os Sistemas de Informações Geográficas (SIG) possuem características específicas relacionadas à sua localização geográfica e relacionamentos espaciais. A definição de regras capazes de tratar das peculiaridades dos dados espaciais, em especial dos relacionamentos entre objetos espaciais, é um desafio quando comparado aos bancos de dados convencionais (Stempliuc, 2008).

Com a popularização dos SIG e o aumento do mapeamento colaborativo por pessoas menos especializadas, a preocupação sobre a garantia da qualidade desses dados vem ganhando destaque (França, 2018), também requerida pela constante demanda de mapeamento atualizado seja por métodos manuais ou semiautomáticos (Siejka et al., 2013).

Para os usuários avançados de SIG, que necessitam realizar atividades que vão além da simples exploração visual dos dados como, por exemplo, em análises de geoprocessamento, codificação, roteamento, etc., é necessário garantir as propriedades geométricas e a manutenção dos relacionamentos entre as feições (Wadembere & Ogao, 2014).

De acordo com Servigne et al. (2000), a qualidade dos dados é a chave para qualquer sistema de informação e, em um SIG, a consistência lógica dos dados

geográficos deve ser verificada para garantir a confiabilidade de decisões baseadas nas geoinformações.

Para Longley et al. (2013), boas análises em SIG não podem substituir concepções ou representações geográficas “pobres”, ou seja, por melhor que sejam os métodos e técnicas aplicadas ao dado geográfico para gerar uma informação desejada, esta terá como fator limitador a qualidade do dado original (Araújo, 2015).

Deste modo, a identificação e correção de inconsistências são essenciais para manter a integridade dos dados espaciais, considerando a sua origem e suas resoluções de visualização (Shen & Ai, 2017).

A detecção e correção completa dos erros é indispensável para provar que todos os procedimentos foram empregados para que os dados geoespaciais modelem e representem adequadamente o mundo real, permitindo que estudos, consultas e análises sejam mais confiáveis (Maraş, 2010).

Vários recursos para a validação dos dados já vêm sendo disponibilizados no mercado de softwares de SIG. No entanto, diversas empresas e órgãos produtores de geoinformação ainda não possuem uma metodologia consolidada para o processo completo de validação da consistência lógica.

Muito embora trabalhos para a implementação de restrições de integridade espaciais em banco de dados como os apresentados por Borges et al. (2002), Stempluc (2008) e Lizardo & Davis Jr. (2017) sejam ideais para a garantia da qualidade no processo de aquisição e alimentação do banco, verifica-se que essas restrições de integridade necessitam ser sistematicamente listadas como requisitos (ou regras) para a avaliação final do conjunto de dados pelo contratante (França et al., 2020).

Ariza-López et al. (2019) afirmam que as regras lógicas podem ser verificadas por rotinas computacionais automáticas. Mas, para que essa validação ocorra, é necessário o detalhamento pormenorizado dos elementos de qualidade da consistência lógica, os quais serão tratados nesta dissertação.

Além disso, algumas lacunas não abordadas explicitamente na literatura são definidas nesta pesquisa. É o caso da consistência semântica de atributos, que verifica a relação lógica entre atributos ou entre atributo e topologia das feições.

Diante da crescente e relevante demanda de dados geoespaciais por diversos setores da sociedade, os quais necessitam realizar consultas e operações de geoprocessamento a partir de dados consistentes, este trabalho tem como finalidade

explorar e aprofundar os conceitos inerentes aos elementos de qualidade da consistência lógica, sistematizando e propondo uma sequência otimizada de validação para um Conjunto de Dados Geoespaciais Vetoriais (CDGV) de forma a garantir a consistência e integridade das informações extraídas desses dados pelos diversos usuários de SIG.

O intuito desta pesquisa é, portanto, servir de referência para o desenvolvimento de procedimentos de controle de qualidade de um CDGV, principalmente no que se refere à validação da consistência topológica e semântica.

A metodologia aqui desenvolvida surgiu no contexto do Mapeamento do Estado da Bahia, um projeto patrocinado pela Superintendência de Estudos Econômicos e Sociais da Bahia (SEI) e executado pelo Exército Brasileiro (EB), por intermédio da Diretoria de Serviço Geográfico (DSG). A pesquisa e aplicação dos procedimentos de avaliação foram feitas se considerando as principais classes da categoria Hidrografia desse projeto, em um banco de dados na modelagem ET-EDGV 2.1.3 previamente validado pelo sistema Gothic (Passos et al., 2017).

Além da contribuição metodológica, este trabalho também demonstra na prática como realizar a validação topológica e semântica, utilizando-se do Sistema de Gerenciamento de Bancos de Dados (SGBD) PostgreSQL com sua extensão espacial PostGIS, a partir de consultas em SQL, as quais estão todas abertamente disponíveis nos apêndices, para que possam ser reutilizadas em outros projetos.

1.1 OBJETIVO GERAL

Sistematizar os procedimentos de validação da consistência lógica de um CDGV, especificando-se regras de integridade para a garantia da consistência dos dados e direcionamento do controle de qualidade do mapeamento topográfico.

1.2 OBJETIVOS ESPECÍFICOS

- Analisar, descrever e sequenciar os procedimentos de validação da consistência lógica e suas medidas de qualidade;
- Propor um elemento de qualidade para a categoria consistência lógica, a ser denominado de consistência semântica;

- Relacionar de forma otimizada os procedimentos e medidas de avaliação de qualidade para os elementos de qualidade da consistência lógica; e
- Executar a validação topológica e semântica de um CDGV.

1.3 ESTRUTURAÇÃO DA DISSERTAÇÃO

A presente dissertação é composta por 8 capítulos. No primeiro capítulo consta uma contextualização do tema, sendo também apresentados o objetivo geral e os específicos deste trabalho.

O segundo capítulo apresenta as principais especificações técnicas para dados geoespaciais vetoriais, nos tópicos ISO 19.157:2013 e *Open Geospatial Consortium* (OGC).

O terceiro capítulo aborda a categoria Consistência Lógica, núcleo deste trabalho, apresentando a definição de seus elementos de qualidade: Consistência Conceitual, Consistência de Domínio, Consistência de Formato e Consistência Topológica. Sendo este último, Consistência Topológica, mais aprofundado no que se refere às estruturas de dados vetoriais (topológica e não-topológica).

Ainda sobre a Consistência Topológica, é abordada a utilização da matriz de 9 interseções dimensionalmente estendida (DE-9IM) na determinação dos relacionamentos topológicos de estruturas “não-topológicas” e os Níveis de Validação Topológica, a saber: Validação da geometria, Validação Intraclasse e Validação Interclasse.

No quarto capítulo, é discutido e proposto um elemento de qualidade denominado de Consistência Semântica, que verifica regras lógicas entre atributos de uma feição, como também regras de atributo relacionado à topologia dessa feição.

Como regras lógicas podem ter exceção, também no quarto capítulo é apresentada uma breve discussão sobre como lidar com esses casos de exceção e como apresentá-los no relatório de qualidade do CDGV.

O quinto capítulo trata da Avaliação da Consistência Lógica, abordando as principais medidas de qualidade, o procedimento de avaliação e o relatório de qualidade.

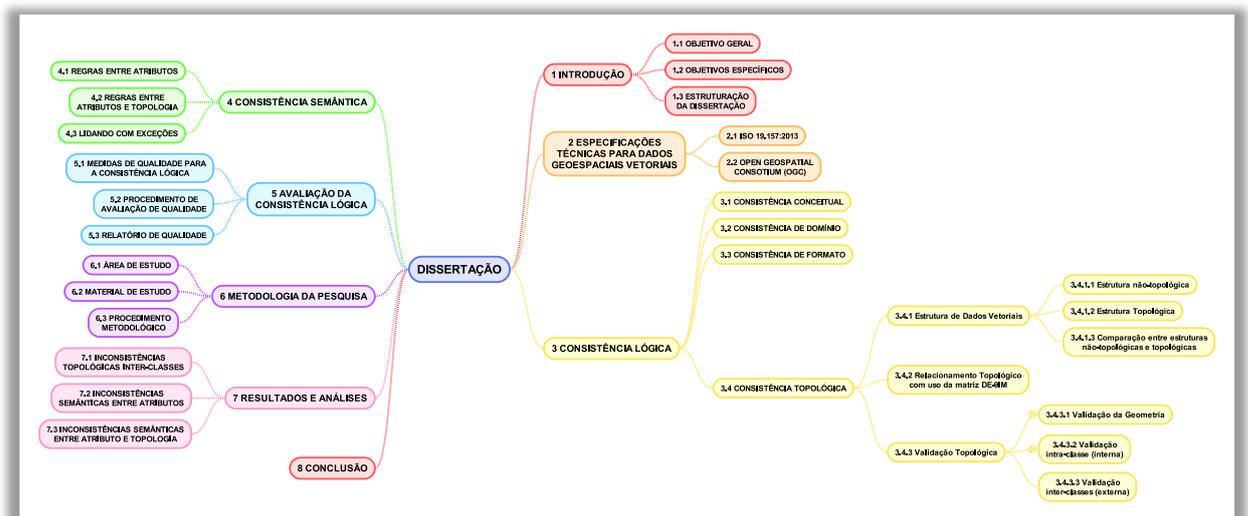
No sexto capítulo, ocorre a aplicação desta pesquisa em um CDGV do Oeste da Bahia, detalhando-se todo o procedimento metodológico para a validação topológica e semântica desses dados.

Já no sétimo capítulo, são apresentados os resultados do exemplo de aplicação na validação dos dados, juntamente com as análises das diversas inconsistências topológicas e semânticas detectadas.

Por fim, o oitavo capítulo é destinado à conclusão, considerações finais e propostas de desenvolvimento e aprofundamento de temas correlatos.

A Figura 1 sintetiza através de um mapa mental os tópicos trabalhados nesta dissertação, para um melhor entendimento e acompanhamento de todo o conteúdo.

Figura 1 - Mapa mental da dissertação ([ver ampliada](#))



Fonte: O autor (2021).

2 ESPECIFICAÇÕES TÉCNICAS PARA DADOS GEOESPACIAIS VETORIAIS

Para a perfeita compreensão da consistência lógica de dados geoespaciais, é essencial o conhecimento prévio da ISO 19.157 que trata dos elementos de qualidade e seus procedimentos de avaliação, bem como das especificações para construção de feições vetoriais da *Open Geospatial Consortium* (OGC), documentadas na *Simple Feature Access* (SFA) – 1ª parte.

2.1 ISO 19.157:2013

A Organização Internacional de Normalização (ISO), do inglês *International Organization for Standardization*, é uma entidade sediada em Genebra, na Suíça, que conta com membros de vários países para a discussão de meios para facilitar internacionalmente a coordenação e unificação de padrões industriais.

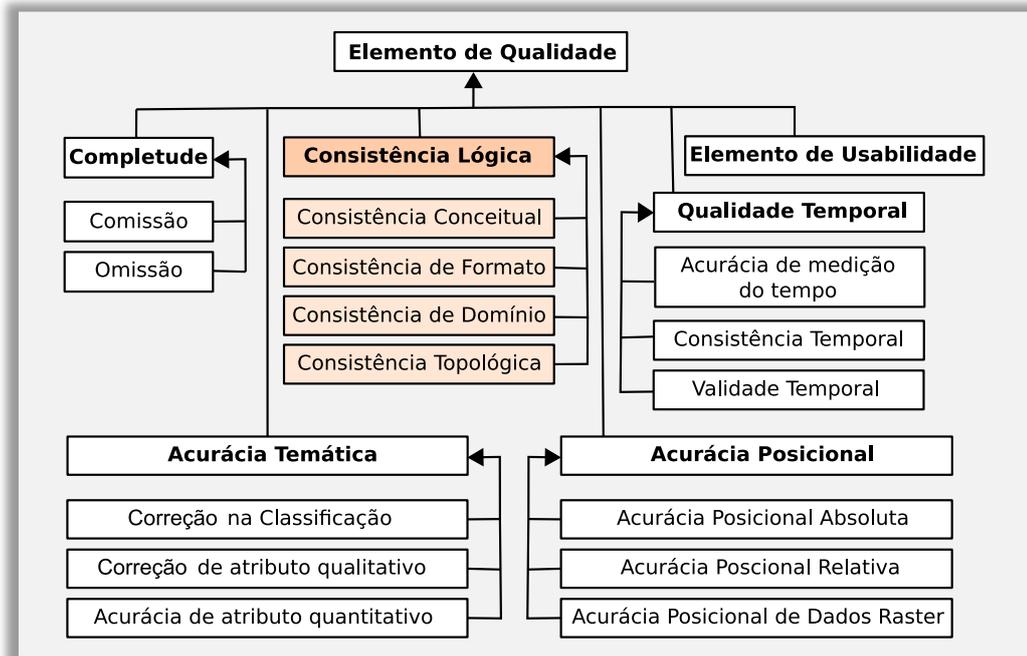
O comitê técnico TC 211, composto por especialistas em SIG e Geomática, foi o responsável, em 2013, pela documentação da ISO 19.157, a qual substituiu as ISO/TS 19.138:2006, ISO 19.114:2003 e ISO 19.113:2002 (ISO, 2013).

A ISO número 19.157 é um padrão internacional de controle de qualidade de dados geoespaciais. O emprego desse padrão significa que as unidades de qualidade estão em conformidade com os elementos de qualidade e o escopo definido, seguindo-se o método de avaliação e nível de conformidade das medidas de qualidade previamente padronizadas (Ariza-López et al., 2019).

O objetivo deste padrão é esclarecer o escopo de qualidade de dados; definir os elementos e as medidas de qualidade; descrever os procedimentos de controle de qualidade; fornecer diretrizes para documentar através de relatórios os resultados da avaliação; e introduzir o conceito de metaqualidade (Mõisja, 2018).

Contudo, ainda ocorre muita confusão sobre os termos relacionados a qualidade (Mõisja, 2018). A Figura 2 apresenta todos os elementos de qualidade da ISO 19157 (2013), os quais podem ser organizados em seis categorias: completude, acurácia temática, consistência lógica, qualidade temporal, acurácia posicional e usabilidade.

Figura 2 - Categorias e elementos de qualidade



Fonte: ISO (2013).

A **Completude** (*DQ_Completeness*) consiste na presença ou ausência de objetos, seus atributos e relacionamentos (ISO, 2013). Esta categoria se especializa nos seguintes elementos de qualidade:

- **Comissão** (*DQ_CompletenessCommission*): excesso de dados presentes em um conjunto de dados;
- **Omissão** (*DQ_CompletenessOmission*): ausência de dados em um conjunto de dados.

A **Acurácia Temática** (*DQ_ThematicAccuracy*) é definida como a acurácia de atributos quantitativos e a correção de atributos qualitativos, além da classificação de feições e seus relacionamentos (ISO, 2013). Esta categoria se especializa nos seguintes elementos de qualidade:

- **Correção de Classificação** (*DQ_ThematicClassificationCorrectness*): comparação dos objetos classificados e seus atributos com um universo de referência;
- **Correção de atributos qualitativos** (*DQ_NonQuantitativeAttributeCorrectness*): forma de saber se um atributo qualitativo (não quantitativo) está correto ou incorreto;
- **Acurácia de atributos quantitativos** (*DQ_QuantitativeAttributeAccuracy*): proximidade de um valor quantitativo com o valor de referência.

A Consistência lógica (*DQ_LogicalConsistency*) corresponde ao grau de aderência a regras lógicas da estrutura dos dados, atribuições e relacionamentos (seja na estrutura conceitual, lógica ou física) (ISO, 2013). Esta categoria se especializa nos seguintes elementos de qualidade:

- Consistência conceitual (*DQ_ConceptualConsistency*): aderência às regras do modelo conceitual;
- Consistência de domínio (*DQ_DomainConsistency*): aderência às restrições de valores definidos no domínio;
- Consistência de formato (*DQ_FormatConsistency*): grau de aderência do tipo de formato de armazenamento do dado ao definido no modelo físico do conjunto de dados;
- Consistência topológica (*DQ_TopologicalConsistency*): coerência das características topológicas explicitamente definidas para um conjunto de dados.

A Acurácia Posicional (*DQ_PositionAccuracy*) é definida como a acurácia da posição de feições considerando um sistema de referência espacial (ISO, 2013). Esta categoria se especializa nos seguintes elementos de qualidade:

- Acurácia Posicional Absoluta (*DQ_AbsoluteExternalPositionalAccuracy*): proximidade do valor das coordenadas coletadas com o valor de referência, aceito como verdadeiro;
- Acurácia Posicional Relativa (*DQ_RelativeInternalPositionalAccuracy*): proximidade das posições relativas entre feições em um conjunto de dados com suas respectivas posições relativas, aceitas como verdadeiras.
- Acurácia Posicional de Dados Raster (*DQ_GridDEDDataPositionalAccuracy*): proximidade posicional dos valores da grade matricial com valores aceitos como verdadeiros.

A Qualidade Temporal (*DQ_TemporalQuality*) é definida como a qualidade dos atributos temporais e do relacionamento temporal das feições (ISO, 2013). Esta categoria se especializa nos seguintes elementos de qualidade:

- Acurácia de medição do tempo (*DQ_AccuracyOfATimeMeasurement*): proximidade do valor das medições do tempo coletadas com os valores de referência;

- Consistência temporal (*DQ_TemporalConsistency*): coerência na ordem do acontecimento dos eventos;
- Validade temporal (*DQ_TemporalValidity*): validade do dado quanto ao tempo.

O Elemento de Usabilidade (*DQ_UsabilityElement*) é baseado em requisitos do usuário, descrevendo a qualidade para um caso específico, no qual os outros elementos da qualidade não o contemplam, ou pela combinação de mais de um elemento de qualidade (ISO, 2013).

Para os elementos de qualidade que possuem a palavra em inglês “*Correctness*” foi considerado a tradução como “Correção”, mas este termo não pode trazer o falso entendimento de que o dado deve ser corrigido, pois essa tarefa não faz parte de uma inspeção. Ou seja, neste caso, a palavra correção significa um indicador de quão correto o dado está, considerando-se sua conformidade com o seu valor de referência.

O objeto deste trabalho é o aprofundamento dos elementos de qualidade da categoria consistência lógica, buscando-se a otimização da ordem de validação desses elementos e a definição dos métodos de avaliação para cada caso com as respectivas medidas de referência para os resultados.

Embora a ISO 19.157 possibilite a expansão dos elementos de qualidade através “elemento de usabilidade” (Mõisja et al., 2018), esta pesquisa também identificou a necessidade de propor um novo elemento de qualidade na categoria da consistência lógica, o qual checará relacionamento lógicos entre os atributos das feições e atributos relacionados à topologia. Tal elemento foi denominado de Consistência Semântica e será explanado no Capítulo 4.

2.2 OPEN GEOSPATIAL CONSORTIUM (OGC)

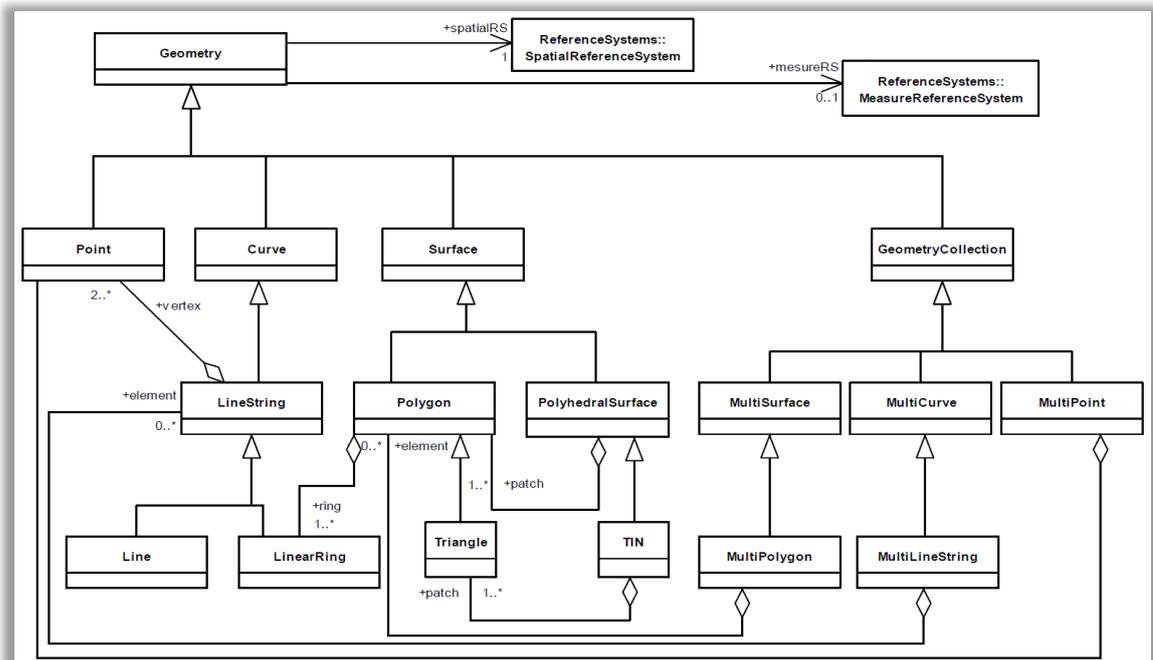
O *Open Geospatial Consortium* (OGC) é um consórcio internacional de mais de 530 empresas, agências governamentais, organizações de pesquisa e universidades orientadas a tornar informações e serviços geoespaciais FAIR - *Findable, Accessible, Interoperable, and Reusable* (OGC, 2020).

A abordagem da OGC baseia-se na participação ativa “*hands-on*” através da engenharia colaborativa levando a padrões e implementações mais maduros, engajando uma comunidade de patrocinadores e participantes com base em um modelo econômico que beneficia todos os envolvidos. Cada iniciativa começa com uma abordagem inovadora para identificar as necessidades de interoperabilidade seguidas pelo desenvolvimento ágil de software para avançar o estado da tecnologia em benefício da sociedade (Percivall et al., 2015).

Os produtos trabalhados pelo OGC são apresentados sob a forma de especificações de interfaces e padrões de intercâmbio de dados (Davis Jr. et al., 2005).

Dentre as especificações da OGC, destaca-se, para este trabalho, a *Simple Feature Access* (SFA) – 1ª parte: arquitetura comum. A SFA, também conhecida como ISO 19.125, descreve a arquitetura comum para feições simples com geometria. O objeto feição simples com geometria é representado por notação UML. A classe base *Geometry* tem como subclasses o *Point*, *Curve*, *Surface* e *GeometryCollection* (Herring, 2011).

O modelo estendido da classe *Geometry* (Figura 3) se especializa em coleção de classes com dimensão 0, 1 e 2, denominadas *MultiPoint*, *MultiLineString* e *Multipolygon* para modelar geometrias correspondentes a coleções de *Points*, *LineStrings* e *Polygons*, respectivamente.

Figura 3 – Hierarquia das Geometrias ([ver ampliada](#))

Fonte: Herring (2011).

Cada objeto geométrico está associado a um sistema de referência espacial, que define como as coordenadas do objeto geométrico devem ser localizadas espacialmente.

Os objetos da classe Geometria podem ser construídos com coordenadas 2D, 3D ou 4D. Geometrias 2D possuem apenas as coordenadas **x** e **y**. Já as geometrias 3D podem possuir as coordenadas **x**, **y** e **z** ou as coordenadas **x**, **y** e **m**. As geometrias 4D têm pontos com coordenadas **x**, **y**, **z** e **m**.

A coordenada **z** de um ponto é tipicamente, mas não necessariamente, a representação da altitude geodésica (em referência ao elipsoide) ou altitude ortométrica (em relação ao geóide). A coordenada **m** pode representar qualquer outra medida como, por exemplo, a data/hora no formato “dia juliano” correspondente à época que o dado foi coletado.

Vale salientar que, para as aplicações de SIG, as geometrias 2D atendem grande parte das análises espaciais. A utilização desnecessária das coordenadas 3D e 4D podem exigir a necessidade de até o dobro de espaço em um banco de dados, considerando-se que cada uma dessas coordenadas é armazenada no formato *double*, ou seja, ponto flutuante de dupla precisão que requer 8 bytes.

O Quadro 1 mostra um comparativo do espaço em disco dado em número de bytes necessários para se armazenar uma geometria considerando-se os seguintes tipos: *Point* (2D), *PointZ* ou *PointM* (3D), e *PointZM* (4D) (Esri, 1998).

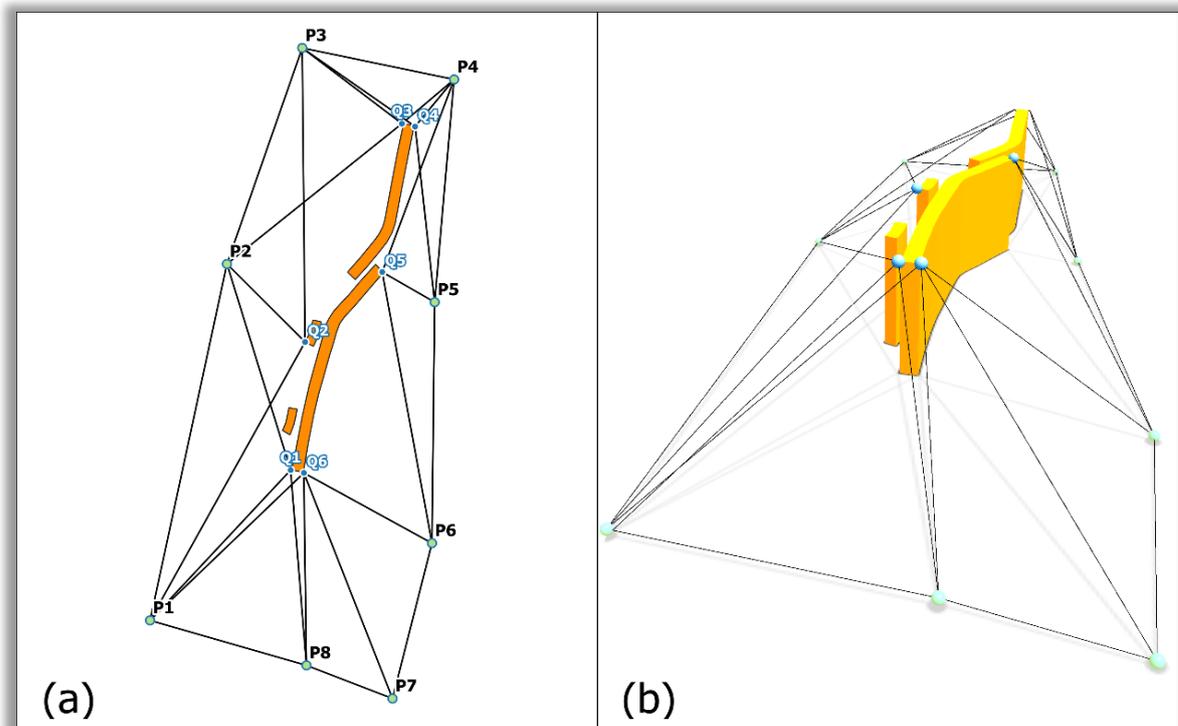
Quadro 1 – Comparativo entre dimensão da geometria e espaço em disco

Tipo	Coordenadas	Dimensão	Espaço
<i>Point</i>	x e y	2D	16 bytes
<i>PointZ</i> ou <i>PointM</i>	x, y e z ou m	3D	24 bytes
<i>PointZM</i>	x, y, z e m	4D	32 bytes

Fonte: Esri (1998).

Apesar do maior espaço em disco, geometrias 3D já vêm sendo amplamente utilizadas no contexto urbano para a visualização em perspectiva de características altimétricas de edificações em relação ao terreno. A Figura 4 apresenta exemplos de objetos geométricos construídos com geometrias do tipo *PointZ*, *LineStringZ* e *PolygonZ* no software de QGIS.

Figura 4 - (a) visualização 2D; (b) visualização 3D do mesmo conjunto de dados



Fonte: França et al. (2021).

A OGC também padroniza a forma de representar as instâncias de cada tipo de geometria, exibindo-as de forma alfanumérica ou binária.

A representação *Well-Known Text* (WKT) tem por objetivo a facilitação de leitura e construção das geometrias e a interpretação de seus elementos pelo ser humano.

O Quadro 2 apresenta exemplos de diferentes tipos de geometrias representadas em WKT com as respectivas descrições. Os exemplos de *PointZ*, *PointM* e *PointZM*, apresentadas nas três últimas linhas deste quadro, podem ser aplicados para os tipos *LineString*, *Polygon*, *MultiPoint*, *MultiLineString* e *MultiPolygon*.

Quadro 2 - Exemplos de geometrias representadas em WKT

Tipo	Representação WKT	Descrição
<i>Point</i>	Point (10 30)	Um ponto
<i>LineString</i>	LineString (10 10, 20 20, 30 40)	Uma linha com 3 pontos
<i>Polygon</i>	Polygon ((10 10, 10 20, 20 20, 20 15, 10 10))	Um polígono com 1 anel exterior e 0 anel interior
<i>Polygon</i>	Polygon ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))	Um polígono com 1 anel exterior e 1 anel interior
<i>Multipoint</i>	MultiPoint ((10 10), (20 20))	Um multiponto com 2 pontos
<i>MultiLineString</i>	MultiLineString ((10 10, 20 20), (15 15, 30 15))	Uma multilinha com 2 linhas.
<i>MultiPolygon</i>	MultiPolygon (((10 10, 10 20, 20 20, 20 15, 10 10)), ((60 60, 70 70, 80 60, 60 60)))	Um multipolígono com 2 polígonos
<i>Point</i>	Point Z (10 10 5)	Um ponto 3D
<i>Point</i>	Point ZM (10 10 5 40)	O mesmo ponto 3D com a coordenada M de valor 40
<i>Point</i>	Point M (10 10 40)	Um ponto 2D com a coordenadas M de valor 40

Fonte: Herring (2011).

A representação *Well-Known Binary* (WKB) possibilita que objetos geométricos sejam codificados em uma forma contínua de bytes, permitindo seu

3 CONSISTÊNCIA LÓGICA

A consistência lógica é uma categoria que contém os elementos de qualidade responsáveis por verificar se os dados estão coerentes com regras lógicas e estruturalmente adequados para a modelagem do mundo real.

De acordo com a ISO (2013), a consistência lógica se especializa nos seguintes elementos de qualidade: consistência conceitual, consistência de domínio, consistência de formato e consistência topológica.

3.1 CONSISTÊNCIA CONCEITUAL

A consistência conceitual indica a aderência a regras do esquema conceitual (ISO, 2013; INSPIRE, 2014).

O esquema conceitual, também denominado de modelo conceitual, é a parte fundamental do processo de desenvolvimento de sistemas de informação (Davis Jr. & Laender, 2000), sendo o mais alto nível de descrição (ou abstração) de relacionamento entre as entidades.

Segundo Câmara (2005), o problema fundamental da geoinformação é a produção de representações computacionais do espaço geográfico. Para Borges et al. (2001), é necessário sistematizar o entendimento a respeito de objetos e fenômenos geográficos em um sistema informatizado, de modo a obter uma forma de representação conveniente, embora simplificada, que seja adequada às finalidades das aplicações do banco de dados.

Para Singh & Singh (2014), a modelagem é um processo de abstração que foca nos elementos essenciais e ignora os desnecessários considerando um objetivo específico.

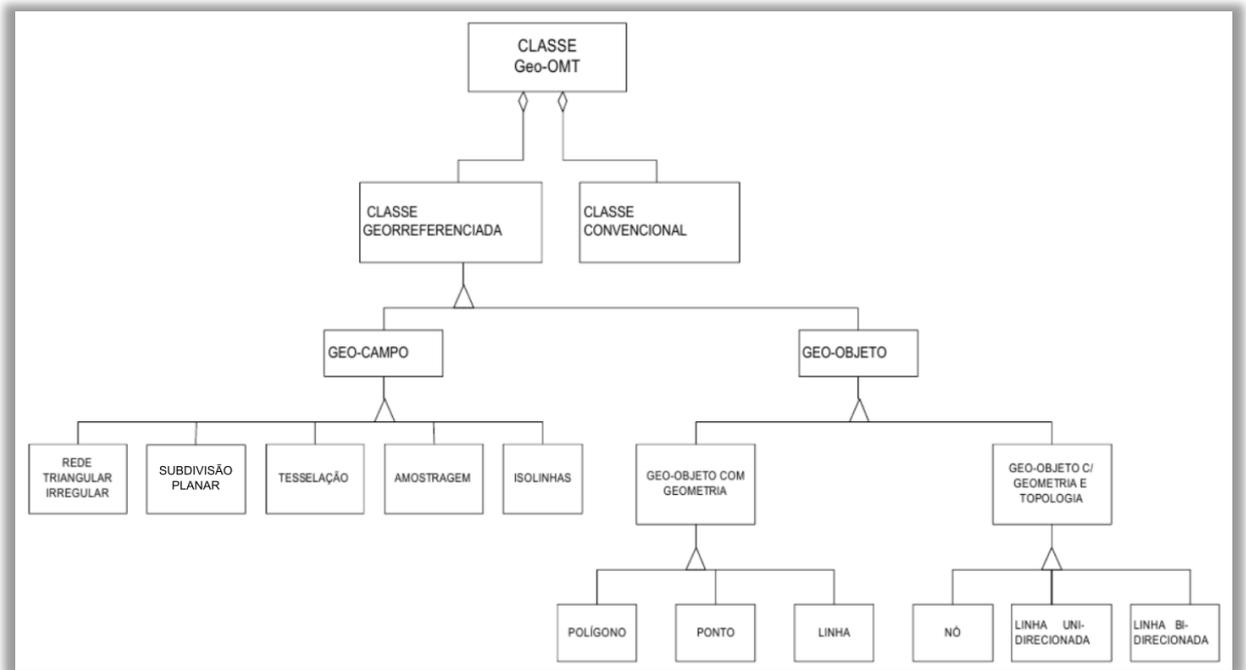
Na modelagem de dados geoespaciais, uma feição é considerada como uma abstração de um fenômeno do mundo real. Quando essa feição está associada a uma localização relativa à Terra, ou seja, georreferenciada, então trata-se de uma feição geográfica (Davis Jr et al., 2005).

As classes de feições se especializam em geo-campo ou geo-objeto. As classes geo-campo são utilizadas para representar objetos ou fenômenos que estejam distribuídos sobre o espaço geográfico de forma contínua, como a altitude,

temperatura e pluviometria. As classes geo-objeto são utilizadas para representar objetos geográficos particulares de forma individual (discreta) como, por exemplo, edificações, rios e árvores (Araújo et al., 2015; Borges et al., 2005).

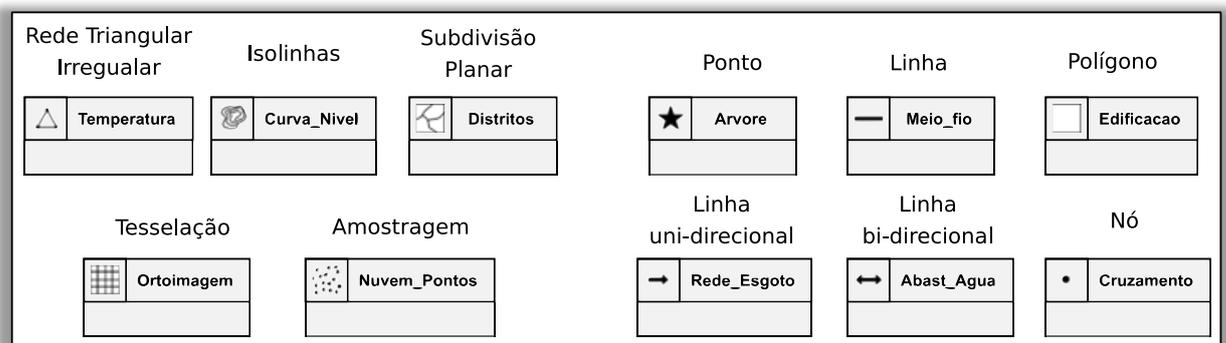
A Figura 6 mostra as especializações das classes geo-campo e geo-objeto e a Figura 7 apresenta exemplos de classes de feições para cada caso.

Figura 6 - Classes geo-campo e geo-objeto



Fonte: Casanova et al. (2005).

Figura 7 - Exemplos de classes de feições



Fonte: Adaptado de Lizardo & Davis Jr. (2017).

De acordo com Borges et al. (2005), a modelagem de dados para aplicações geográficas tem necessidades adicionais, tanto com relação à abstração de

conceitos e entidades, quanto ao tipo de entidades representáveis e seu inter-relacionamento.

No Brasil, o modelo OMT-G tem se tornado a opção mais empregada para a especificação do banco de dados geográficos em razão de ser o modelo adotado pela Infraestrutura Nacional de Dados Espaciais (INDE) (Araújo et al., 2015).

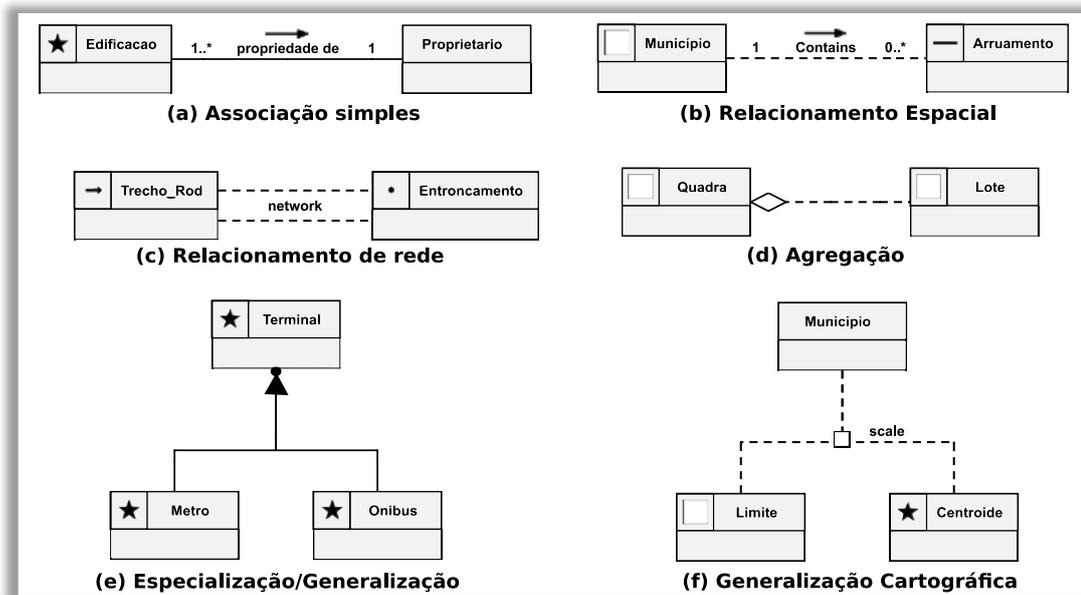
O *Object Modeling Technique for Geographic Applications* (OMT-G) é definido como um modelo orientado a objetos que fornece primitivas geométricas apropriadas para representar dados geoespaciais, atendendo relacionamentos espaciais, representação múltipla escalas de objetos, estruturas de redes e especificações de regras de integridade espacial (Borges et al., 2001; Lizardo et al., 2014).

O modelo OMT-G toma por base a modelagem *Unified Modeling Language* (UML), introduzindo às classes as respectivas primitivas geográficas (Borges et al., 2005) com o objetivo de diminuir a distância entre o modelo mental do espaço e o modelo de representação (Araújo et al., 2015).

Além das onze alternativas de representação da modelagem OMT-G para geo-campos (isolinhas, tesselação, rede triangular irregular, amostragem e subdivisão planar) e para geo-objetos (ponto, linha, polígono, nó, linha uni-direcional e linha bi-direcional).

A Figura 8 apresenta as seguintes representações de relacionamento entre as classes: (a) associações simples; (b) relacionamento espacial; (c) relacionamento de rede; (d) agregação espacial; (e) especialização/generalização; e (f) generalização cartográfica.

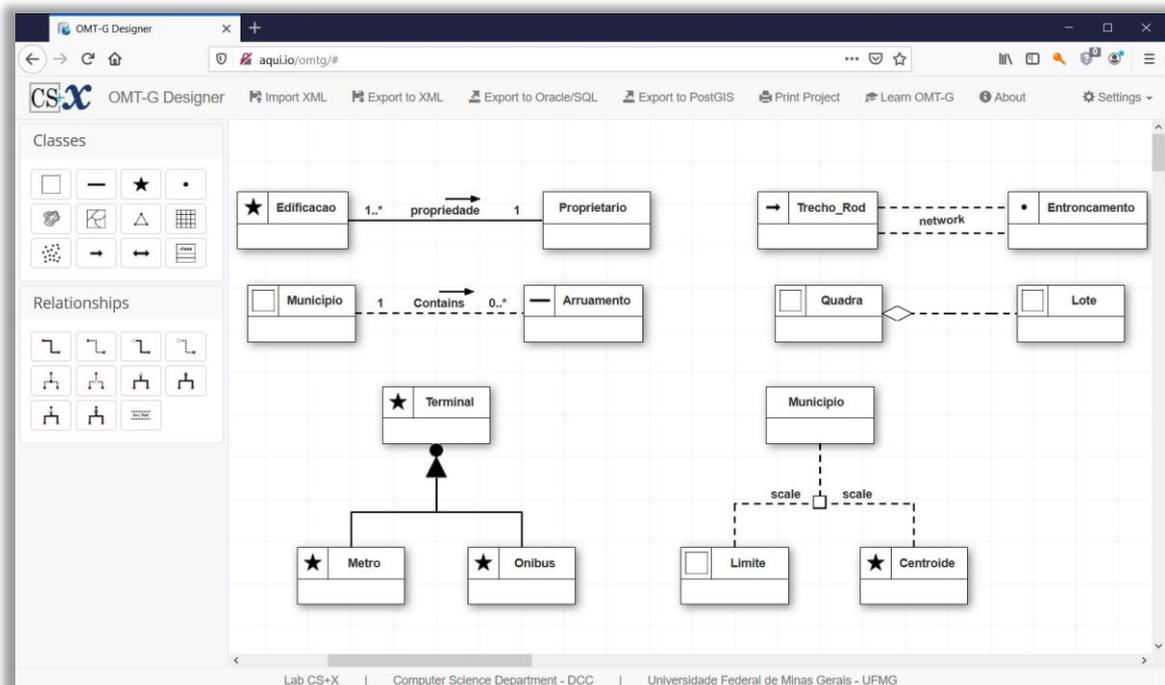
Figura 8 - Notações da OMT-G para relacionamentos



Fonte: Lizardo et al. (2014).

Vale ressaltar que, para a modelagem de dados geoespaciais no padrão OMT-G, foi desenvolvido por Lizardo & Davis Jr. (2014) a ferramenta livre “OMT-G Designer” (Figura 9), disponível na internet, através do endereço eletrônico: <http://aqui.io/omtg/>.

Figura 9 - Ferramenta livre “OMT-G Design”



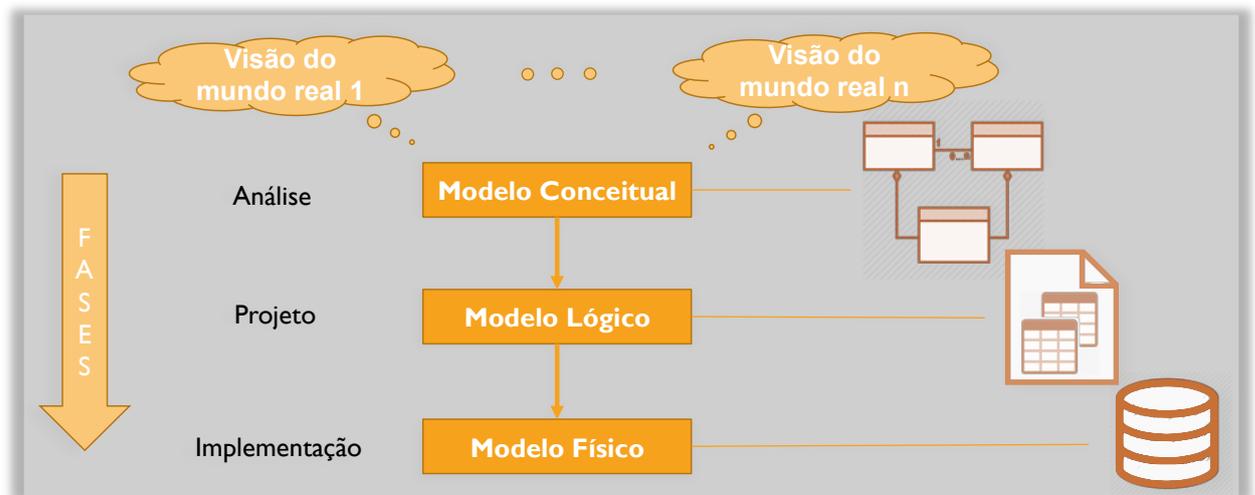
Fonte: Adaptado de Lizardo & Davis Jr. (2014).

Contudo, para a avaliação de qualidade, muitas vezes o modelo conceitual não oferece recursos completos para a verificação completa da aderência a estrutura desejada, requerendo que as regras e detalhes de construção da estrutura física sejam apresentadas e documentadas de forma mais explícita.

Nesse contexto, o entendimento das fases de modelagem de um banco de dados se torna imprescindível. De acordo com Singh & Singh (2014), o processo de modelagem de um banco de dados consiste em modelagem: conceitual, lógica e física.

A Figura 10 sintetiza as fases de modelagem de um banco de dados, iniciando-se pela abstração do mundo real a partir das visões de n indivíduos, passando pela fase de documentação do projeto através do modelo lógico, até a implementação do banco, na fase de construção do modelo físico.

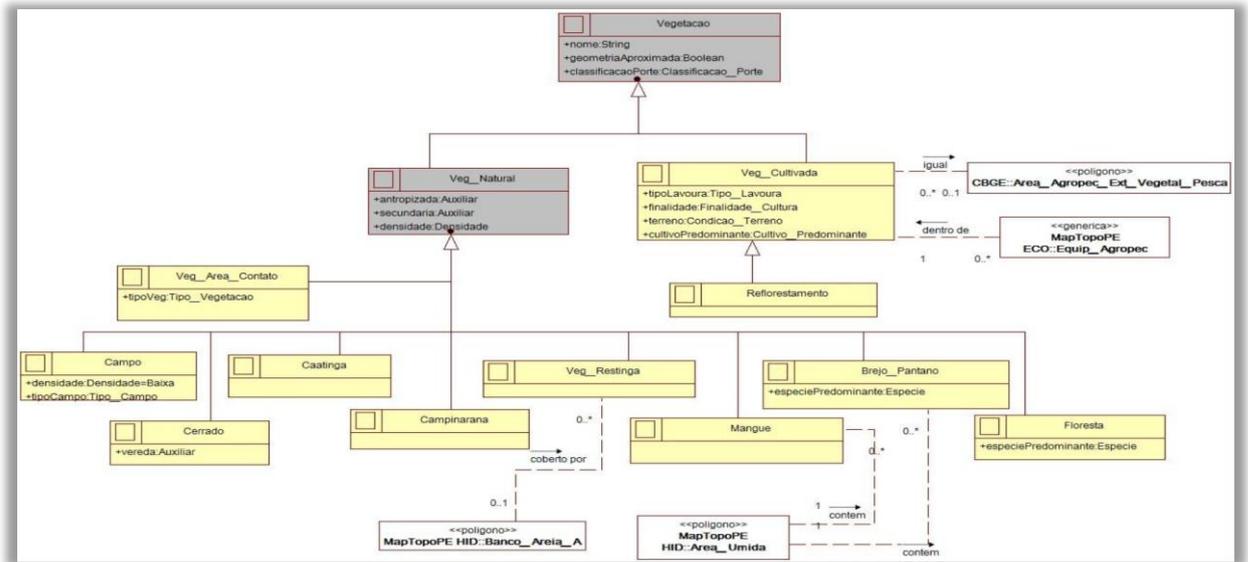
Figura 10 - Fases de modelagem de um banco de dados



Fonte: O autor (2021).

O Modelo Conceitual descreve como um sistema é organizado e como ele deve operar. Neste processo, todo o conjunto de dados (espaciais e não-espaciais), seus atributos e relacionamentos devem ser identificados. Para um SIG, as camadas temáticas centrais devem se basear nessas entidades básicas (Singh & Singh, 2014).

A Figura 11 é um exemplo de modelo conceitual que consiste no diagrama de classes para a categoria vegetação do modelo ET-EDGV 3.0 (CONCAR, 2017).

Figura 11 - Exemplo de modelo conceitual ([ver ampliada](#))

Fonte: CONCAR (2017).

O Modelo Lógico expressa como o banco de dados deve ser implementado computacionalmente (Elmasri e Navathe, 2000), detalhando as classes, seus atributos, operações, relacionamentos e restrições de integridade.

A documentação do Modelo Lógico traduz-se através do dicionário de dados, também chamado de metadados do repositório. Neste documento, os significados de todos os elementos do banco são descritos, apresentando os domínios, valores padrões e requisitos de integridade.

A Figura 12 apresenta um extrato da documentação da ET-EDGV 3.0 apresentando o modelo lógico para as classes “Patio” e “Ponte”.

Figura 12 - Exemplo para Modelo Lógico

1.8.11 Patio				
Classe	Descrição		Código	Geometria
Patio	Pátio é uma área livre destinada a manobras, manutenção, estacionamento de veículos automotores, composições ferroviárias, aeronaves e portuários.		1.8.11	□
Atributo	Tipo (tamanho)	Descrição	Domínio	Requisito
Nome	Alfanumérico (80)	Indica o nome completo da instância.	A ser preenchido	0..1
geometriaAproximada	Booleano	Indica que a geometria adquirida é aproximada em relação à escala prevista para o produto cartográfico.	-	1
modalUso	Modal_Uso	Indica o tipo de modal viário do qual faz parte.	Seção 3.35	0..*
administração	Administracao	Identifica a esfera administrativa responsável pelo atracadouro.	Seção 3.1	0..*
operacional	Auxiliar	Indica a situação em relação ao uso.	Seção 3.4	1
situacaoFisica	Situacao_Fisica	Identifica a situação, quanto à atividade atual, do pátio.	Seção 3.57	0..
finalidadePatio	Finalidade_Patio	Indica a finalidade do pátio.	Seção 3.25	1..*
1.8.12 Ponte				
Classe	Descrição		Código	Geometria
Ponte	Ponte é uma obra de arte cuja finalidade é permitir a transposição de um trecho de drenagem.		1.8.12	★ – □
Atributo	Tipo (tamanho)	Descrição	Domínio	Requisito
tipoPonte	Tipo_Ponte	Indica o tipo de ponte.	Seção 3.122	1
vaoLivreHoriz	Real	Indica a largura, do trecho mais largo da ponte (entre as pilastros), em metros (m).	A ser preenchido	0..1
vaoVertical	Real	Indica a altura, do trecho mais alto da ponte, em metros (m).	A ser preenchido	0..1
cargaSuportMaxima	Real	Indica a carga máxima admissível sobre a ponte, em toneladas (ton).	A ser preenchido	0..1

Fonte: CONCAR (2017).

O Modelo Físico envolve, de fato, a construção do banco de dados de acordo com os requisitos que foram estabelecidos no modelo lógico. Nesta fase, uma linguagem de definição de dados, do inglês *Data Definition Language* (DDL), é utilizada, seguindo os critérios da *Structured Query Language* (SQL) do respectivo Sistema Gerenciador de Banco de Dados (SGBD).

A Figura 13 é um extrato de DDL/SQL para a criação da classe “rel_curva_nivel_I” no PostgreSQL. Observa-se que na criação da tabela, é definida a chave primária e as chaves estrangeiras, bem como outras restrições de checagem. Os índices espaciais também podem ser criados nesta fase.

Figura 13 - Modelo Físico para a classe “rel_curva_nivel_l” utilizando DDL/SQL

```

1 CREATE TABLE rel_curva_nivel_l
2 (
3     id serial,
4     geom geometry(MultiLineString,4674),
5     cota integer,
6     depressao smallint NOT NULL DEFAULT 0,
7     indice smallint NOT NULL DEFAULT 0,
8     CONSTRAINT rel_curva_nivel_l_pk PRIMARY KEY (id),
9     CONSTRAINT rel_curva_nivel_l_depressao_fk FOREIGN KEY (depressao)
10    REFERENCES dominios.depressao (code),
11    CONSTRAINT rel_curva_nivel_l_indice_fk FOREIGN KEY (indice)
12    REFERENCES dominios.indice (code),
13    CONSTRAINT valid_cota CHECK (cota > 0 and cota < 1500)
14 );
15
16 CREATE INDEX rel_curva_nivel_l_geom_idx
17 ON cb.rel_curva_nivel_l USING GIST(geom);

```

Fonte: DSG (2015).

O Modelo Físico de um banco de dados pode ser traduzido através de um arquivo na extensão .sql, contendo a DDL para a criação de toda a estrutura do banco. Este tipo de arquivo é costumeiramente chamado de “banco semente”.

O Quadro 4 apresenta um comparativo entre os modelos conceitual, lógico e físico de acordo com suas principais características nas fases de construção de banco de dados.

Quadro 4 - Comparativo entre os modelos conceitual, lógico e físico

Conceitual	Lógico	Físico
<ul style="list-style-type: none"> • Identificação de classes e relacionamentos; • Alto nível de abstração; • Não há obrigatoriedade de especificar atributos e suas restrições de integridade; • Alta flexibilidade; • Boa representação da semântica entre classes; • Abstração do objeto e fenômenos independente da implementação. • Documentação como diagramas. 	<ul style="list-style-type: none"> • Descrição dos dados com o máximo de detalhes; • Linguagem genérica, a ser aplicada em qualquer banco de dados; • Definição de todos os atributos e suas restrições de integridade; • Pouca flexibilidade; • Descrição dos dados através de tabelas; • Documentação como dicionário de dados. 	<ul style="list-style-type: none"> • Definição da estrutura de criação de todo o esquema do banco de dados através de uma linguagem DDL/SQL; • Dependência do SGBD; • Definição dos nomes e tipos das colunas, restrições, chaves primárias e estrangeiras dos relacionamentos; • Nenhuma flexibilidade; • Padronização de “banco semente” no formato .sql.

Fonte: O autor (2021).

Devido à distância entre o modelo conceitual e o físico para dados geoespaciais, muitas vezes a riqueza das representações e sua semântica acabam não sendo captadas (Lizardo & Davis Jr., 2017).

Além disso, a modelagem física de banco de dados pode variar de acordo com o SGBG utilizado, onde, por exemplo, um tipo de dado pode ser diferente no MySQL, SQL Server, PostgreSQL, Oracle, entre outros. Destaca-se a importância de se aderir a padrões básicos como o (*American National Standards Institute*) ANSI SQL e o OGC SFA.

Por isso, para uma boa especificação de um banco de dados, a documentação que conste o modelo conceitual e o modelo lógico, é indispensável para catalogar e comunicar a estrutura e o tipo dos dados, além de descrever o significado das classes e seus atributos.

Todavia, para a validação do controle de qualidade, a comparação do banco de dados produzido pelo contratado com o banco definido pelo contratante (banco semente) se torna a melhor opção para se avaliar o elemento de qualidade consistência conceitual.

Na prática, essa comparação entre o produto e o banco de dados de referência consiste nas verificações:

- Se as classes correspondem (tem o mesmo nome);
- Se para cada classe os seus atributos e o respectivo tipo de dado estão de acordo com o modelo.
- Se para cada classe a coluna de geometria está de acordo com o padrão *WKBBGeometryType* especificado.

3.2 CONSISTÊNCIA DE DOMÍNIO

A consistência de domínio verifica a aderência dos valores ao respectivo domínio dos valores (ISO, 2013).

Matematicamente, a interpretação do domínio de uma função é o conjunto de valores de entrada para os quais a função é definida. Dessa forma, valores fora do domínio não devem ser permitidos. Neste contexto, o raciocínio é análogo quando empregado na modelagem de dados geoespaciais.

Para o melhor entendimento dos possíveis domínios para dados geoespaciais, é necessário o conhecimento prévio dos principais tipos de dados, os quais podem ser classificados em:

- Numérico: dados mensurados em um subconjunto dos números inteiros \mathbb{Z} ou reais \mathbb{R} ;
- Textual: dados representados por uma sequência de caracteres (*string*);
- Temporal: dados no formato de data e/ou hora; e
- Booleano: verdadeiro (*true*) ou falso (*false*).

Para os tipos numéricos e textuais, existem variações que estão associadas ao número de bytes utilizados para seu armazenamento em disco, intervalo de valores ou número de dígitos ou caracteres.

O Quadro 5 mostra os tipos numéricos para banco de dados do PostgreSQL, bem como o respectivo espaço de armazenamento e valores possíveis.

Quadro 5 - Tipos numéricos do PostgreSQL

Tipo de dado	Tamanho	Descrição	Intervalo
<i>smallint</i>	2 bytes	Inteiro (menor intervalo)	-32768 a +32767
<i>integer</i>	4 bytes	Inteiro (típico)	-2147483648 a +2147483647
<i>bigint</i>	8 bytes	Inteiro (maior intervalo)	-9223372036854775808 a +9223372036854775807
<i>decimal/numeric</i>	variável	Real de precisão definida, exato	até 131072 dígitos antes da casa decimal; até 16383 dígitos depois da casa decimal
<i>real</i>	4 bytes	Precisão variável, inexato	decimal com precisão de 6 dígitos
<i>double precision</i>	8 bytes	Precisão variável, inexato	decimal com precisão de 15 dígitos
<i>smallserial</i>	2 bytes	Inteiro autoincrementado (menor intervalo)	1 a 32767
<i>serial</i>	4 bytes	Inteiro autoincrementado (típico)	1 a 2147483647
<i>bigserial</i>	8 bytes	Inteiro autoincrementado (maior intervalo)	1 a 9223372036854775807

Fonte: PostgreSQL (2021).

O Quadro 6 apresenta as características dos tipos utilizados para armazenar dados textuais para n caracteres no PostgreSQL.

Quadro 6 - Tipos de dados textuais

Tipo de dado	Descrição
<i>character varying(n), varchar(n)</i>	Comprimento variável para o limite de n caracteres
<i>character(n), char(n)</i>	Comprimento fixo, espaços em branco completados
<i>text</i>	Variável, sem limite de comprimento

Fonte: PostgreSQL (2021).

Os tipos temporais podem ser do tipo: DATE para armazenar somente datas; TIME para armazenar o tempo (hora, minuto e segundo) do dia; TIMESTAMP para armazenar a data e tempo; e INTERVAL para armazenar um intervalo de tempo.

Os tipos temporais podem também ser armazenados com a informação de fuso horário através da cláusula WITH TIME ZONE. Além disso, os dados de tempo aceitam o parâmetro de tempo (p) que especifica o número de casas decimais para os valores de segundos (podendo variar de 0 a 6).

O Quadro 7 apresenta os tipos de dados temporais do PostgreSQL e suas características.

Quadro 7 - Tipos de dados temporais

Tipo de dado	Tamanho	Descrição	Valor Máximo	Valor Mínimo	Resolução
<i>timestamp [(p)] [without time zone]</i>	8 bytes	Data e hora (sem fuso horário)	4713 BC	294276 AD	1 micro-segundos / 14 dígitos
<i>timestamp [(p)] with time zone</i>	8 bytes	Data e hora, com fuso horário	4713 BC	294276 AD	1 micro-segundos / 14 dígitos
<i>date</i>	4 bytes	Apenas data	4713 BC	5874897 AD	1 dia
<i>time [(p)] [without time zone]</i>	8 bytes	Hora do dia (sem fuso horário)	00:00:00	24:00:00	1 micro-segundos / 14 dígitos
<i>time [(p)] with time zone</i>	12 bytes	Hora do dia, com fuso horário	00:00:00+1559	24:00:00-1559	1 micro-segundos / 14 dígitos
<i>interval [fields] [(p)]</i>	16 bytes	Intervalo de tempo	-178000000 years	178000000 years	1 micro-segundos / 14 dígitos

Fonte: PostgreSQL (2021).

O tipo de dado booleano é o de menor tamanho (apenas 1 byte), o qual pode ser utilizado para armazenar um atributo cujo estado seja verdadeiro, falso ou desconhecido. Em SQL são utilizadas as palavras reservadas TRUE, FALSE ou NULL, sendo para “true” também seja aceita as *strings* ‘yes’ e ‘on’ ou o número 1; e para “false”, seja aceito ‘no’ e ‘off’ ou o número 0.

Conhecendo-se os principais tipos de dados utilizados em SIG, os quais foram apresentados até esse momento, é possível ter melhor compreensão dos possíveis domínios que podem ser exigidos para cada caso.

Para dados numéricos do tipo inteiro, o domínio pode ser um subconjunto dos valores possíveis para esse tipo de dado, com a finalidade de atender requisitos lógicos para um determinado atributo. Neste caso, o domínio pode ser de intervalo (*range*) ou código (*codelist*).

Como exemplo de domínio de intervalo, pode-se citar a o atributo de contagem do número de vias da classe “trecho rodoviário”. Nessa situação, esperam-se valores entre 1 e 5, pois um número menor que 1 é inaceitável e maiores que 5 são pouco prováveis de acontecer.

O tipo inteiro também é bastante utilizado para codificar uma lista de atributos, através de mapa de valores (*codelists*). Nessa situação, valores fora do *codelist* ferem a lógica para o domínio do atributo.

A Figura 14a apresenta um exemplo de *codelist* para o atributo “terreno” da classe “Veg_cultivada” da ET-EDGV3.0 e a Figura14b é o resultado de sua implementação no PostgreSQL.

Figura 14 - Exemplo de Codelist

3.11 Condicao_Terreno <<codeList>>

Nome/Valor	Descrição
Condicao_Terreno	Indica a condição do terreno, onde ocorre a vegetação cultivada.
Desconhecida	Valor desconhecido.
Seco	Terreno seco.
Irrigado	Terreno irrigado.
Inundado	Terreno inundado.

(a)

	code [PK] smallint	code_name text
1	0	Desconhecido
2	1	Seco
3	2	Irrigado
4	3	Inundado

(b)

Fonte: CONCAR (2017).

Os dados numéricos do tipo ponto flutuante são empregados para descrever grandezas contínuas, sendo o seu domínio a um subconjunto dos números reais. Neste caso, o mais comum é utilizar o domínio do tipo intervalo (*range domain*).

Um exemplo de aplicação de domínio para dados numéricos do tipo ponto flutuante, é padronizar intervalo de valores possíveis para o atributo elevação da classe que indique a altimetria. Considerando-se que terreno está acima do nível médio dos mares, o valor de altitude no Brasil pode variar entre 0 e 2.994 metros e, nestas condições, qualquer valor fora desse intervalo será uma inconsistência de domínio.

Para dados do tipo textual, o domínio pode ser definido como um subconjunto aceito como correto para uma *string* composta por um determinado número de caracteres (*n*), seja ela uma palavra ou sentença. Este tipo de verificação ainda é um desafio que vai além das funcionalidades dos SGBD.

Por exemplo, um atributo textual pode ser verificado se está contido no subconjunto de palavras que atendem as normas ortográficas de um determinado idioma.

Na mesma linha de raciocínio, outros problemas de digitação podem ser identificados e evitados, por exemplo: espaços duplos; espaço no início ou no fim da *string*; e letras maiúsculas ou minúsculas de forma inadequada.

Muito embora seja de implementação complexa em SIG, domínios para atributos textuais compostos por frases ou expressões também podem ser logicamente considerados para a verificação automática de sintaxe, considerando as normas gramaticais do idioma utilizado no banco de dados.

Os dados temporais vêm ganhando ênfase nas ferramentas de SIG, não apenas pelo registro da data-hora dos fenômenos, mas também devido às demandas associadas ao controle de produção como, por exemplo, em plataformas *Building Information Modeling* (BIM) (Ariza-López, 2019; França, 2018b).

O domínio para dados temporais pode estar associado a um intervalo de datas, dias da semana ou, até mesmo, faixas de horário específicas.

Os dados booleanos aceitam apenas três tipos de valores (TRUE, FALSE e NULL), no entanto, para esse tipo de dado, pode-se exigir como domínio somente as opções TRUE ou FALSE, obrigando-se o preenchimento do atributo.

Não somente no tipo booleano, mas também em todos os outros tipos de dados a cláusula NOT NULL pode ser utilizada como restrição, evitando-se um atributo nulo para o objeto instanciado.

Outro aspecto que pode ser compreendido como consistência de domínio, é a definição do regiões (2D ou 3D) para as coordenadas da geometria das feições.

Em um projeto, costuma-se estabelecer a área de trabalho (ou mapeamento), sendo ela muitas vezes definida por um polígono ou retângulo envolvente (*bounding box*).

A validação da consistência de domínio para as geometrias de feições consiste em verificar se as coordenadas da geometria se encontram espacialmente dentro da área de mapeamento. As coordenadas podem estar em um espaço bidimensional (2D) ou tridimensional (3D), quando possuir a coordenada Z.

De forma geral, o Quadro 8 apresenta os principais tipos de domínio para os tipos de dados utilizado em SIG elencados neste trabalho.

Quadro 8 - Resumo dos principais tipos de domínios

Tipo de dado	Tipo de domínio	Descrição	Exemplo
Numérico - inteiro	Código (Codelist)	Valor específico associado a um código.	x in (0, 1, 3, 7)
	Intervalo (Range)	Espaço definido por valores mínimo e/ou máximo.	$1 < x < 10$ $y > 0$
Numérico – ponto flutuante	Intervalo (Range)	Espaço definido por valores mínimo e/ou máximo.	$0 < x < 3.14$ $y > 0.01$
Textual	Ortografia	Atributo textual que atende a norma ortográfica	'eucalipito'
	Digitação	Observância a regras de digitação.	' vila São João '
	Sintaxe	Disposição lógica entre palavras	'ruas isolada'
Temporal	Intervalo de tempo	Faixa contínua de tempo para data ou hora definida por valores máximo e/ou mínimo	1999 a 2021 08:00 às 17:00
	Sazonal	Períodos descontínuos na linha de tempo	Verão Segunda a Quinta
Booleano	Não nulo (not null)	Atributo de preenchimento obrigatório	True False
Geometria	Região 2D	Geometria (G) dentro de um Espaço 2D definido (S2D)	G within S2D
	Região 3D e 4D	Idem ao anterior, acrescentando a definição de limites para as coordenadas Z e M	G within S2D e $\min < Z < \max$ e/ou $\min < M < \max$

Fonte: O autor (2021).

Em todos os casos, rotinas de restrição de integridade (*constraints*) podem ser implementadas diretamente no banco de dados de forma forçar que as regras sejam obedecidas e garantir a qualidade dos dados (Borges et al., 2002).

Na prática, grande parte dos domínios citados no Quadro 8 é pouco explorada em SIG, surgindo várias possibilidades de pesquisa e implementação.

No que se refere ao controle de qualidade, a documentação desses critérios se torna indispensável nas especificações dos produtos cartográficos, considerando-os como regras a serem observadas na validação da consistência de domínio.

3.3 CONSISTÊNCIA DE FORMATO

A consistência de formato mede o grau de conformidade de como os dados são armazenados na estrutura física do conjunto de dados (ISO, 2013).

A consistência de formato está relacionada ao arquivo final do processo de produção dos dados geoespaciais. Assim, nesta avaliação, são verificados erros ou conflitos na estrutura física do arquivo fornecido no formato indicado na especificação do produto.

Baseadas em IBGE (2020) e DSG (2016), as seguintes situações podem ser citadas como inconsistência de formato:

- Arquivo diferente da especificado (extensão do arquivo), por exemplo, Shapefile ao invés de Geopackage;
- Arquivo corrompido, quando há falha na transmissão dos dados, impedindo-os de ser totalmente ou parcialmente inteligíveis;
- Sistema de Referência de Coordenadas (SRC) diferente da especificada, por exemplo, WGS84 ao invés de SIRGAS2000;
- Codificação dos dados diferente da especificada, por exemplo, Latin1 (ISO-8859-1) ao invés de UTF-8.

Por cerca de 30 anos o Shapefile, formato desenvolvido pela Esri, foi utilizado para a disponibilização de dados geoespaciais vetoriais na internet, sendo até hoje o mais conhecido e distribuído. Este formato consiste em 3 arquivos obrigatórios: os dados de geometria (*.shp*); a tabela de atributos (*.dbf*); o arquivo de índice que relaciona a geometria da feição ao atributo (*.shx*). O arquivo (*.prj*), embora não seja obrigatório, é essencial para definir o sistema de referência de coordenadas.

Apesar da boa aceitação nos últimos anos, o formato Shapefile possui uma série de limitações que dificultam o armazenamento dos dados (Esri, 2020), a citar:

- Formato composto por vários arquivos (.shp, .shx, .dbf, .prj, etc.);
- Nome do campo do atributo com, no máximo, 10 caracteres;
- Número máximo de 255 caracteres para atributos textuais;
- Quantidade limitada de tipos de dados (inteiro, ponto flutuante, data e texto);
- Limitação do tamanho do arquivo a 2 GB;
- Não guarda o tipo de codificação dos caracteres ASCII.

Segundo Rashidan & Musliman (2015), o formato Geopackage, lançado em 2014, tem o intuito de contornar problemas de interoperabilidade entre diferentes plataformas e APIs, de forma a facilitar o acesso e gerenciamento dos dados, por exemplo em dispositivos móveis para SIG, além de reduzir redundâncias nos dados, aprimorar a capacidade de armazenamento e facilitar a conversão de dados.

O formato Geopackage é um padrão aberto desenvolvido pela OGC, não-proprietário e independente de plataforma, que serve como repositório de vários tipos de dados geoespaciais, facilitando a distribuição e aumentando a interoperabilidade entre plataformas, aplicações e serviços web (OGC, 2018; Rashidan & Musliman, 2015).

Diferentemente do Shapefile, que armazena apenas uma camada vetorial, o Geopackage funciona como um container, podendo armazenar dentro de um único arquivo diversas camadas vetoriais e tabelas, além de dados matriciais (raster). Outras vantagens podem ser citadas:

- Padrão aberto da OGC;
- Ampla implementação (GDAL, QGIS, Esri, R, Python, etc.);
- Menor espaço em disco quando comparado com banco de dados convencionais;
- Maior velocidade de consultas e renderização;
- Sem limitação do tamanho de arquivo.

Portanto, observa-se a tendência de que o arquivo Shapefile seja substituído pelo Geopackage para a disponibilização de dados geoespaciais na internet nos próximos anos, tendo em vista a redução de possibilidades de inconsistências de formato. Dentre os diversos motivos, pode ser citado o simples fato de o

GeoPacake armazenar todo o conjunto de dados em um único arquivo *.gpkg*, o que não acontece com o Shapefile que se torna inválido se os arquivos *.shp*, *.dbf* e *.shx* não corresponderem.

Para grandes projetos de mapeamento cartográfico, como é o caso do Projeto de Mapeamento do Estado da Bahia (Passos & França, 2018), as entidades contratantes têm optado que, além da entrega do CDGV final no formato Shapefile, o banco de dados instanciado, na mesma estrutura do banco semente, seja fornecido, buscando-se facilitar o controle e evitar a conversão de dados.

A disponibilização de banco de dados PostGIS pode ser feita através de arquivos de *backup*, sejam eles no formato *.backup* ou *.sql*. Nessa disponibilização, deve-se atentar para a compatibilidade da versão do software PostgreSQL e na codificação ASCII, a qual está relacionada ao sistema operacional da máquina onde se encontra o servidor de banco de dados.

3.4 CONSISTÊNCIA TOPOLÓGICA

A consistência topológica verifica a coerência das características topológicas explicitamente estabelecidas a um conjunto de dados (ISO, 2013).

O termo topologia remonta ao século XVIII, mais precisamente 1.736, onde Leonhard Euler, o fundador das bases da topologia e teoria dos grafos, solucionou o problema das sete pontes de Königsberg (Figura 15), atual Kalingrado, exclave russo no mar Báltico. Euler provou, com base na topologia, que não era possível atravessar todas as pontes sem ter que cruzar uma delas pelo menos duas vezes (Shields, 2012; Romanholi & Queiroz Filho, 2018).

também conhecida como estrutura espaguete (ou *spaghetti data structure*) (Maraş et al.,2010), sendo representada com as três formas geométricas (ponto, linha e polígono) de acordo com a escala de mapeamento.

As feições que são representadas pela geometria ponto são elementos zero-dimensionais (0D), sendo cada um deles definido por um par de coordenadas (x,y). As feições do tipo linha são uni-dimensionais (1D), sendo cada elemento definido por uma sequência de coordenadas (x,y). Já as feições do tipo polígono são bi-dimensionais (2D), sendo definidas por formas fechadas e compostas por linhas que começam e terminam no mesmo ponto.

Maraş et al. (2010) citam alguns problemas relacionados a vetorização em estruturas não-topológicas que dificultam a análise espacial dos dados:

- a) Feições do tipo ponto podem não estar no ponto de interseção de linhas, por exemplo, ponte na interseção de rio e rodovia;
- b) Relacionamento de vizinhança entre feições pode não estar claro;
- c) Pontos de contato não coincidem, por exemplo, rio que não coincide com a borda de lago;
- d) Como a vizinhança entre dois polígonos é representada duas vezes, pode não haver total coincidência em todos os vértices, ocasionando problemas de sobreposição (*overlaps*) ou lacuna (*gaps*).
- e) Navegação não é possível quando o conceito de rede e direcionamento não está presente nas feições do tipo linha.

Siejka et al. (2013) apontam que o modelo vetorial simples (não-topológico) tem duas desvantagens básicas. Uma delas é a redundância de dados em situações quando um ponto pertence a dois ou mais objetos. Nesses casos, as coordenadas dos pontos são salvas em cada um desses objetos. A outra desvantagem significativa é que os relacionamentos espaciais entre os objetos só podem ser detectados através de métodos mais complexos de geometria analítica, ocasionando um maior custo computacional.

3.4.1.2 Estrutura Topológica

A topologia possibilita examinar características que vão além das informações geométricas das feições. Segundo Maraş et al. (2010), o objetivo do conhecimento topológico em SIG é aumentar as oportunidades de análises espaciais, de forma a

representar relações espaciais como vizinhança, coincidência, direção e ligações.

Na topologia, um nó (*node*) corresponde ao ponto; o arco (*edge*) é um elemento correspondente a linha; e o polígono (*face*). O arco é um conjunto de pares de coordenadas que começa com um nó e termina com um nó. O polígono, por sua vez, é um espaço bi-dimensional delimitado por arcos.

Há duas abordagens: topologia arco-nó, baseada em grafos e bastante utilizada para representar redes; e a arco-nó-polígono, que considera a área dos polígonos (Casanova et al., 2005). Tanto na abordagem arco-nó quanto arco-nó-polígono, a topologia pode ser compreendida como o relacionamento entre seus elementos (nós, arcos e polígonos), sendo um meio de definir relacionamentos espaciais, em adição aos relacionamentos métricos geometricamente identificáveis (Maraş et al., 2010).

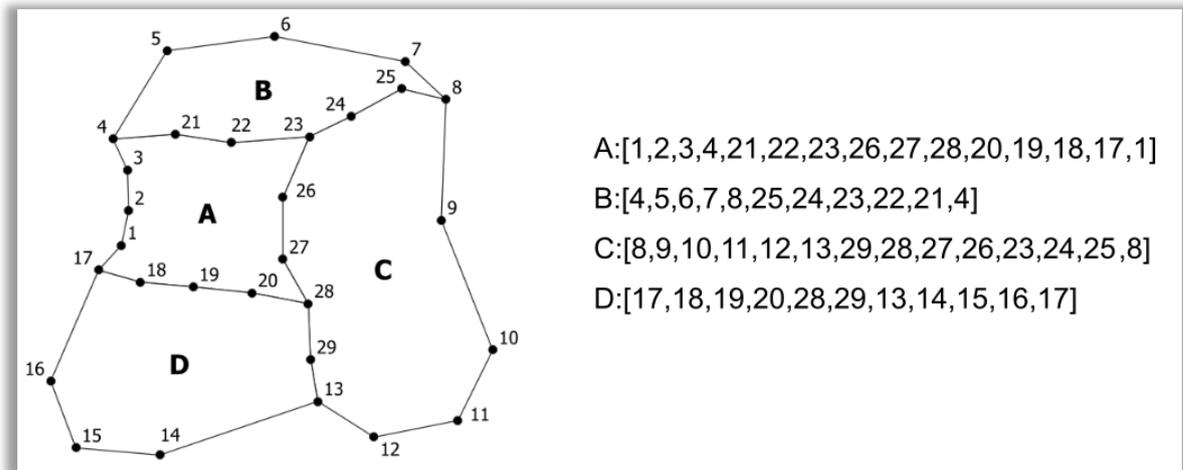
Referente à topologia arco-nó-polígono, para que um banco de dados geoespaciais tenha uma estrutura topológica, os seguintes elementos devem ser determinados e armazenados:

- a) Arcos que definem as fronteiras de cada polígono (na tabela topologia de polígono);
- b) Relação de vizinhança entre polígonos (na tabela topologia de arco);
- c) Conexão nos pontos de interseção (na tabela topologia de nó); e
- d) Pontos inicial e final dos arcos (na tabela arco-coordenadas).

3.4.1.3 Comparação entre estruturas não-topológicas e topológicas

A Figura 16 apresenta um exemplo de estrutura não-topológica (espaguete), verificando-se, à esquerda, os polígonos A, B, C e D e, à direita o identificador dos pontos utilizados para a construção de cada polígono. Observa-se que cada polígono é representado por um laço (*loop*), representado por uma sequência de pontos.

Figura 16 - Exemplo de estrutura não-topológica (espaguete)



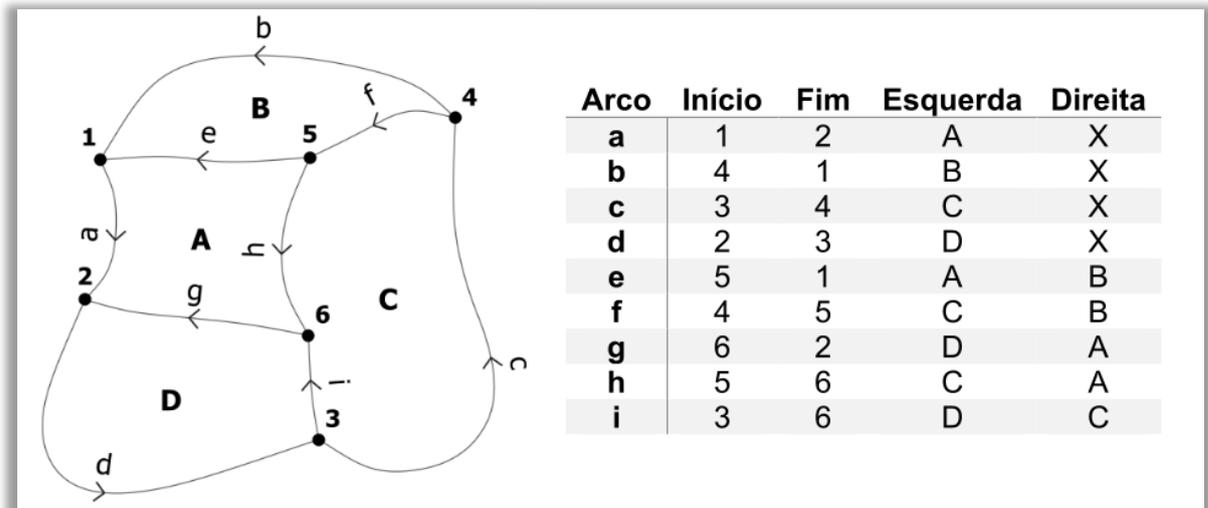
Fonte: Adaptado de Cao (2014).

Na estrutura não-topológica, não há representação explícita das relações topológicas entre as feições como, por exemplo, de adjacência. Além disso, verifica-se a redundância de dados para os pontos compartilhados por dois ou mais polígonos.

A Figura 17 é um exemplo de estrutura topológica, onde as seguintes propriedades são observadas:

- cada arco tem exatamente um nó inicial e final;
- cada nó pode ser inicial ou final (ou ambos) de pelo menos um arco direcional;
- cada face é cercada por um ou mais arcos;
- Arcos podem se interceptar apenas nos seus nós; e
- Cada arco direcionado deve ter apenas uma face à direita ou à esquerda.

Figura 17 - Exemplo de estrutura topológica



Fonte: Adaptado de Cao (2014).

Nos bancos de dados, as propriedades topológicas podem ser armazenadas e utilizadas para diversas consultas como a conexão de linhas, pontos de rede e feições que se encontram à direita ou esquerda de feições lineares.

Entretanto, a existência de dados topológicos, em adição aos dados não-topológicos (espaguete) e seus atributos, pode causar um aumento considerável no volume do banco de dados.

Quando a digitalização de feições do tipo polígono é realizada através da vetorização de linhas (França et al., 2018), obedecendo-se a construção topológica para os arcos em comum, é possível citar as seguintes vantagens:

- Ganho de tempo (evitando-se a vetorização duplicada e fronteiras comuns);
- Garantia que os polígonos compartilham a mesma fronteira;
- Fronteiras serão representadas sem pontos repetidos;
- Rápido processamento dos dados; e
- Redução de custos.

3.4.2 Relacionamento Topológico com uso da matriz DE-9IM

No contexto de SIG, a topologia é compreendida como o posicionamento relativo entre feições, onde a consistência topológica está relacionada à verificação da adequação dos relacionamentos espaciais de suas geometrias às situações identificadas no mundo real.

A despeito de como os dados vetoriais estão estruturados, o entendimento dos relacionamentos entre as geometrias é essencial para a execução de análises espaciais. Estes relacionamentos são definidos através de operadores topológicos (Egenhofer et al., 1993).

Os diversos tipos de relacionamentos espaciais entre feições podem ser determinados através da matriz das 9 interseções dimensionalmente estendidas, do inglês, *Dimensionally-Extended 9-Intersection Matrix* (DE-9IM) (Clementini & Di Felice, 1995).

A DE-9IM é aplicada a geometrias de feições armazenadas em estruturas vetoriais não-topológicas. O resultado da DE-9IM foi estabelecido pela OGC como base para a implementação das funções de relacionamento espacial nos SGBD (Lizardo & Davis Jr., 2017).

A matriz DE-9IM é resultante das interseções do interior (I), fronteira (B) e exterior (E) de duas geometrias analisadas. O interior, fronteira e exterior para as geometrias ponto, linha e polígono são:

- I(ponto) = ponto;
- I(linha) = todos os pontos da linha, excluindo os pontos inicial e final no caso de linha aberta;
- I(polígono) = os pontos dentro da fronteira;
- B(ponto) = \emptyset
- B(linha) = os pontos inicial e final de linha aberta ou \emptyset para linhas fechadas;
- B(polígono) = anéis (linhas) que compõe o polígono;
- E(pontos, linhas ou polígonos) = o que não é interior ou fronteira.

A Figura 18 apresenta a configuração da matriz M resultante das interseções entre interior, fronteira e exterior de uma geometria A em relação a geometria B.

Figura 18 - Matriz DE-9IM

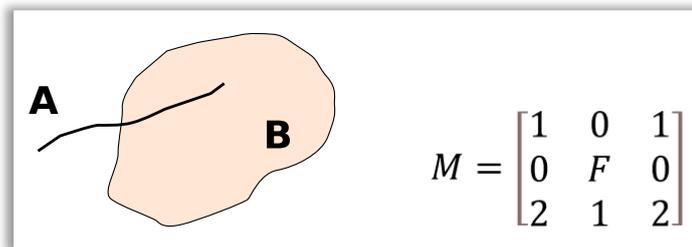
$$M = \begin{bmatrix} I(A) \cap I(B) & I(A) \cap B(B) & I(A) \cap E(B) \\ B(A) \cap I(B) & B(A) \cap B(B) & B(A) \cap E(B) \\ E(A) \cap I(B) & E(A) \cap B(B) & E(A) \cap E(B) \end{bmatrix}$$

Fonte: Adaptado de Clementini & Di Felice (1995).

A DE-9IM pode ser aplicada a objetos espaciais de diferentes dimensões (Borrmann & Rank, 2009). No caso de uma geometria A interceptar outra geometria B, é retornado o valor 0, 1 ou 2, de acordo com a dimensão da interseção (ponto, linha e polígono, respectivamente).

A Figura 19 é um exemplo de resultado da matriz DE-9IM para uma configuração entre uma geometria A do tipo linha e outra geometria B do tipo polígono. Quando não há interseção, o resultado é “F” (para *false*); já no caso de contrário (*true*), é retornado a dimensão da interseção.

Figura 19 - Exemplo de resultado da matriz DE-9IM



Fonte: O autor (2021).

Os operadores topológicos, também denominados de predicados topológicos (Borrmann & Rank, 2009), retornam o valor booleano “*True*” se o resultado de M for classificado nos possíveis esquemas de relações espaciais, verificando-se situações como: intercepta, toca, cruza, contém, dentro de, disjunto, igual, sobrepõe, entre outros predicados espaciais.

O Quadro 9 apresenta o padrão de valores aceitos da matriz DE-9IM para os principais predicados topológicos. Quando o caractere “T” é utilizado, significa que pode ser aceito qualquer dimensão (0, 1 ou 2), Já quando é utilizado o caractere asterisco (*), significa que pode ser qualquer valor “T” ou “F”.

Retornando atenção à situação da Figura 19, e comparando com as possibilidades do Quadro 9, fica evidente que a geometria A intercepta a geometria B, bem como A cruza B.

Quadro 9 - Padrão de matriz DE-9IM para os principais predicados topológicos

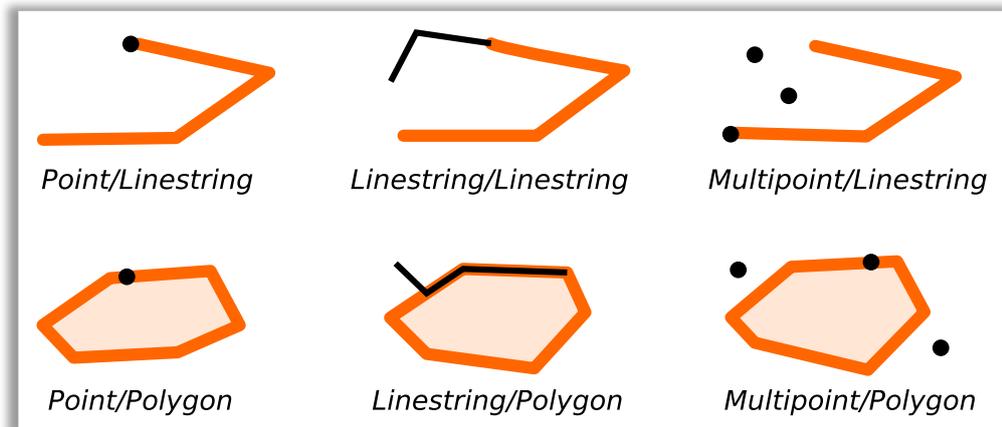
Predicado Topológico	Padrão de Matriz
A.Equals(B)	$\begin{bmatrix} T & * & F \\ * & * & F \\ F & F & * \end{bmatrix}$
A.Disjoint(B)	$\begin{bmatrix} F & F & * \\ F & F & * \\ * & * & * \end{bmatrix}$
A.Intersects(B)	$\begin{bmatrix} T & * & * \\ * & * & * \\ * & * & * \end{bmatrix}$ or $\begin{bmatrix} * & T & * \\ * & * & * \\ * & * & * \end{bmatrix}$ or $\begin{bmatrix} * & * & * \\ T & * & * \\ * & * & * \end{bmatrix}$ or $\begin{bmatrix} * & * & * \\ * & T & * \\ * & * & * \end{bmatrix}$
A.Touches(B)	$\begin{bmatrix} F & T & * \\ * & * & * \\ * & * & * \end{bmatrix}$ or $\begin{bmatrix} F & * & * \\ * & T & * \\ * & * & * \end{bmatrix}$ or $\begin{bmatrix} F & * & * \\ T & * & * \\ * & * & * \end{bmatrix}$
A.Crosses(B)	$\begin{bmatrix} T & * & T \\ * & * & * \\ * & * & * \end{bmatrix}$ or $\begin{bmatrix} T & * & * \\ * & * & * \\ T & * & * \end{bmatrix}$ or $\begin{bmatrix} 0 & * & * \\ * & * & * \\ * & * & * \end{bmatrix}$
A.Overlaps(B)	$\begin{bmatrix} T & * & T \\ * & * & * \\ T & * & * \end{bmatrix}$ or $\begin{bmatrix} 1 & * & T \\ * & * & * \\ T & * & * \end{bmatrix}$
A.Within(B)	$\begin{bmatrix} T & * & F \\ * & * & F \\ * & * & * \end{bmatrix}$
A.Contains(B)	$\begin{bmatrix} T & * & * \\ * & * & * \\ F & F & * \end{bmatrix}$

Fonte: Strobl (2017).

Os operadores topológicos também podem ser utilizados com geometrias multipartes. Algumas possibilidades de operadores de acordo com os tipos de geometria são apresentadas nas Figuras 19 a 25. Em todos os casos um tipo de geometria A, na cor preta, é relacionada a uma segunda geometria B, na cor laranja.

- Toca (*touches*): testa se uma geometria A toca uma geometria B, ou seja, checa se as geometrias possuem pelo menos um ponto em comum, desde que seus interiores não se interceptem (Figura 20).

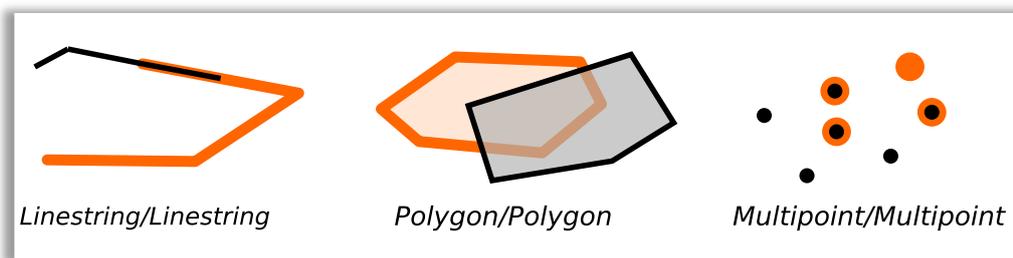
Figura 20 - Operador “Toca – touches”



Fonte: Adaptado de Esri (2010).

- Sobrepõe (*overlaps*): testa se uma geometria A sobrepõe outra geometria B de mesma dimensão. As geometrias possuem alguns, mas não todos os pontos em comum (Figura 21).

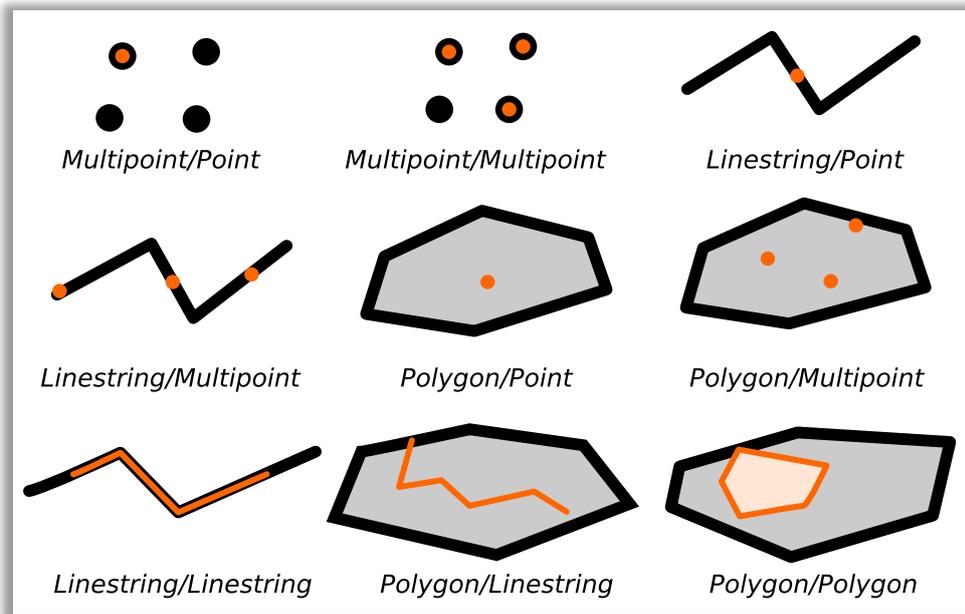
Figura 21 - Operador “Sobrepõe – overlaps”



Fonte: Adaptado de Esri (2010).

- Contém (*contains*): testa se uma geometria A contém outra geometria B, ou seja, todos os pontos da geometria B são ponto da geometria A e os interiores das duas geometrias têm pelo menos um ponto em comum (Figura 22).

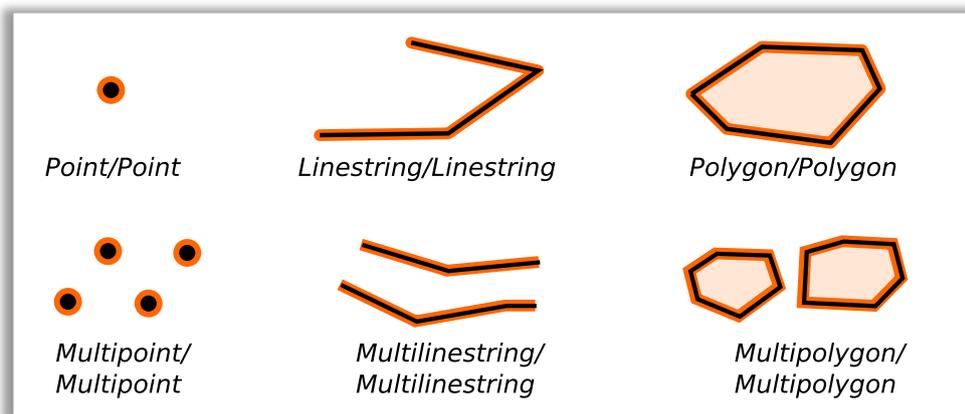
Figura 22 - Operador “Contém – contains”



Fonte: Adaptado de Esri (2010).

- Igual (*equals*): testa se uma geometria é igual a outra, ou seja, os interiores se interceptam e nenhuma parte do interior ou fronteira de uma geometria intercepta o exterior da outra (Figura 23).

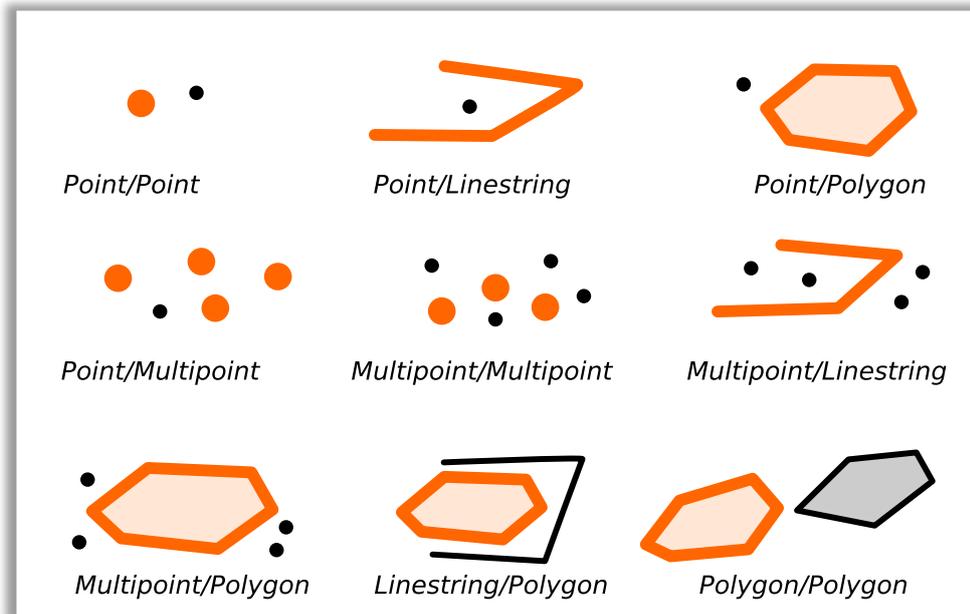
Figura 23 - Operador “Igual – equals”



Fonte: Adaptado de Esri (2010).

- Disjunto (*disjoint*): testa se uma geometria é disjunta da outra, ou seja, as duas geometrias não têm pontos em comum (Figura 24).

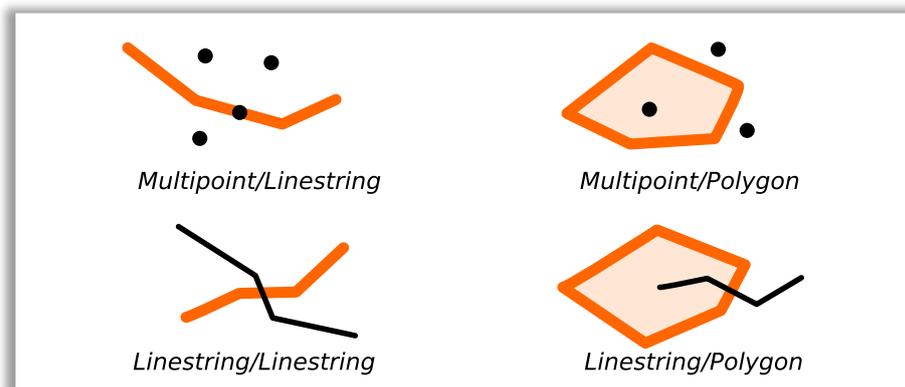
Figura 24 - Operador “Disjunto – disjoint”



Fonte: Adaptado de Esri (2010).

- Cruza (*crosses*): testa se uma geometria cruza outra geometria específica. As geometrias têm alguns, mas, não todos os pontos interiores em comum (Figura 25).

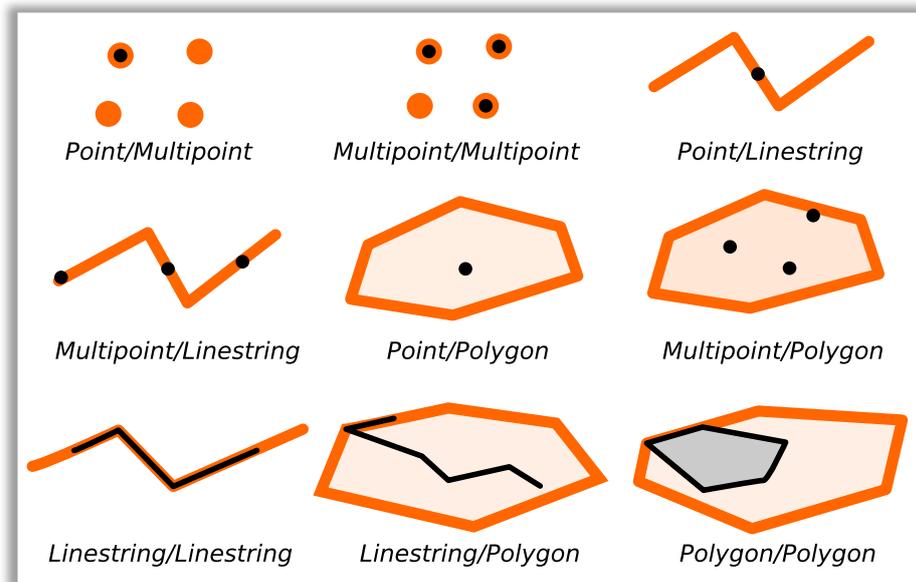
Figura 25 - Operador “Cruza – crosses”



Fonte: Adaptado de Esri (2010).

- Dentro de (*within*): testa se uma geometria A está dentro da geometria B. Cada ponto da geometria A corresponde a um ponto da geometria B e os interiores de ambas possuem pelo menos um ponto em comum (Figura 26). É o equivalente a “B contém A”.

Figura 26 - Operador “Dentro de – within”



Fonte: Adaptado de Esri (2010).

3.4.3 Validação Topológica

No processo de validação topológica é comum se deparar com geometrias inválidas ou, até mesmo, geometrias construídas com configuração inadequada à representação cartográfica da feição geográfica para uma escala de mapeamento. Essas geometrias impossibilitam, ou induzem ao erro, a determinação de relacionamentos espaciais, bem como qualquer outra análise de geoprocessamento ou medições geométricas como área ou perímetro.

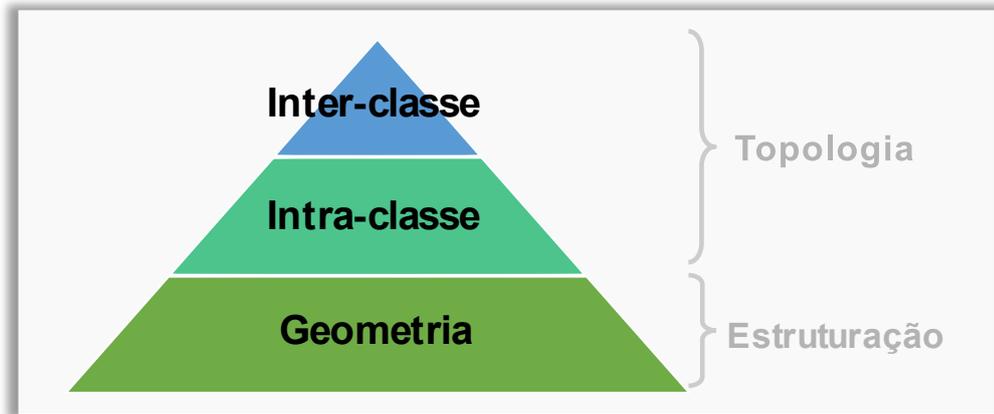
Para melhor entendimento do processo de validação topológica, considerando-se os tipos de inconsistências e o fluxo de verificação, foram definidos neste trabalho os seguintes níveis de validação:

1. Validação da geometria: objetiva checar validade da geometria de acordo com as *Simple Feature specifications* (SFS) da OGC e outras regras de construção da geometria exigidas pelo modelo.
2. Validação intraclasses (interna): objetiva identificar inconsistências topológicas entre feições de uma mesma classe (IBGE, 2017).
3. Validação interclasses (externa): objetiva identificar inconsistências topológicas entre feições de classes diferentes (IBGE, 2017).

A Figura 27 representa o processo de validação de um conjunto de dados por meio de uma pirâmide, tendo como base a estruturação da geometria, seguida pelo

correto relacionamento topológico entre as feições da mesma classe até o topo, onde se verificam relacionamentos topológicos entre feições de classes diferentes.

Figura 27 - Níveis de validação topológica



Fonte: O autor (2021).

Os dados geoespaciais devem ser construídos (ou estruturados) de acordo com o modelo de dados e inconsistências nas geometrias (seja de forma ou posição relativa dos vértices) devem ser verificadas de forma a buscar representar fielmente uma feição do mundo real.

As inconsistências topológicas estão relacionadas ao posicionamento relativo entre as feições dentro da mesma classe (intraclasse) ou em classes diferentes (interclasse), de forma a atenderem o significado de relacionamento espacial dos objetos. Essa verificação pode ser feita através da criação de estruturas topológicas, mas o mais comum é que especialistas utilizarem métodos analíticos baseados na matriz DE-9IM.

No caso deste trabalho, o processo de validação dos dados é realizado em estruturas não-topológicas, entretanto buscando-se analisar a topologia entre as feições através de operadores topológicos, baseados na matriz DE-9IM.

3.4.3.1 Validação da Geometria

A validação da geometria consiste em verificar se a geometria das feições está de acordo com os padrões da OGC para feições simples, bem como outros critérios estabelecidos pelo modelo de dados.

Na validação da geometria podem ser realizadas as seguintes verificações:

- 1) Geometria vazia;
- 2) Geometria não-simples;
- 3) Geometria inválida;
- 4) Geometria limpa;
- 5) Geometria multiparte;
- 6) Geometria com ângulos pequenos;
- 7) Geometria fora das medidas; e
- 8) Fechamento de anel linear.

3.4.3.1.1 *Geometria Vazia*

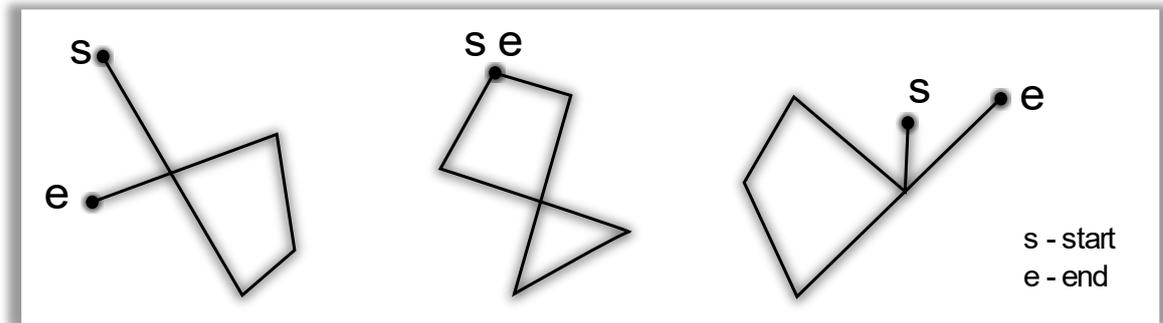
Uma geometria vazia é o nome dado quando um objeto é criado no banco de dados sem armazenar a respectiva geometria. No PostGIS, a função `ST_IsEmpty` pode ser utilizada para identificar os registros com geometria sem valor (NULL). Para evitar esse tipo de problema, a restrição NOT NULL pode ser aplicada para a coluna geometria na definição de cada tabela (camada).

3.4.3.1.2 *Geometria não-simples*

De acordo com as especificações do OGC, uma geometria é dita “simples” quando em seu interior não possui pontos anômalos, como auto-intersecção ou auto-tangência (Herring, 2011).

Casos de geometrias não-simples podem ser observados em geometrias do tipo Multipoint, Linestring, MultiLinestring. Geometrias do tipo Polygon e MultiPolygon, por definição devem ter seus anéis interiores e exteriores como simples. A Figura 28 apresenta alguns exemplos de linhas classificadas como “não-simples”.

Figura 28 - Exemplo de linhas não-simples



Fonte: Herring (2011).

3.4.3.1.3 Geometria inválida

O termo “geometria válida” é aplicado a polígonos. Andrade (2018) considera que para geometrias do tipo linha ser válido, basta apenas ser simples.

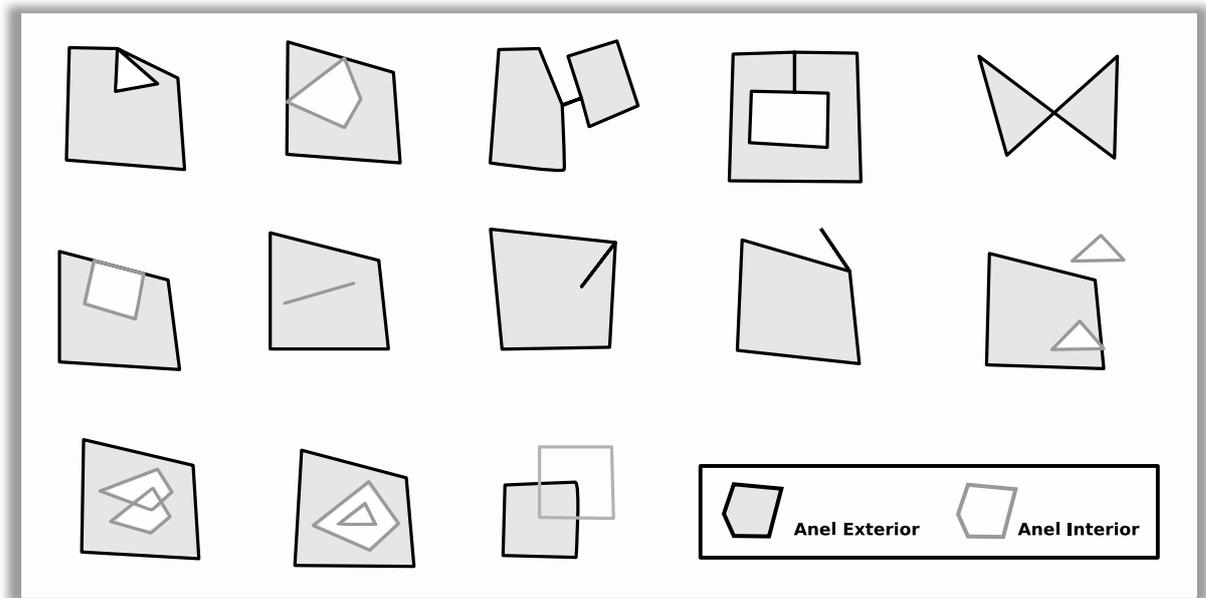
Uma geometria é dita inválida quando não atende a definição de polígono da SFS da OGC: uma superfície planar definida por 1 fronteira exterior e 0 ou mais fronteiras interiores, onde cada fronteira interior define um buraco no polígono. Ledoux et al. (2012) apresentam as seguintes regras para definir um polígono válido:

- a) Os anéis (LinearRing) que definem as fronteiras exterior e interior devem ser simples e fechados.
- b) Os anéis não podem se cruzar, mas podem se interceptar em apenas um ponto de tangência.
- c) Polígonos não podem ter linhas de corte ou filete (externo ou interno);
- d) O interior de cada polígono deve ser um conjunto de pontos conectados;
- e) Os anéis internos não podem estar fora do anel exterior e nem dentro de outros anéis interiores.

Geometrias inválidas impossibilitam diversas operações de geoprocessamento, além de falsear medições de área e perímetro.

Existe um número não exaustivo de possibilidades de geometrias inválidas. A Figura 29 apresenta alguns exemplos de casos que ferem as regras que definem um polígono válido.

Figura 29 - Exemplo de geometria inválidas



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

3.4.3.1.4 Geometria Limpa

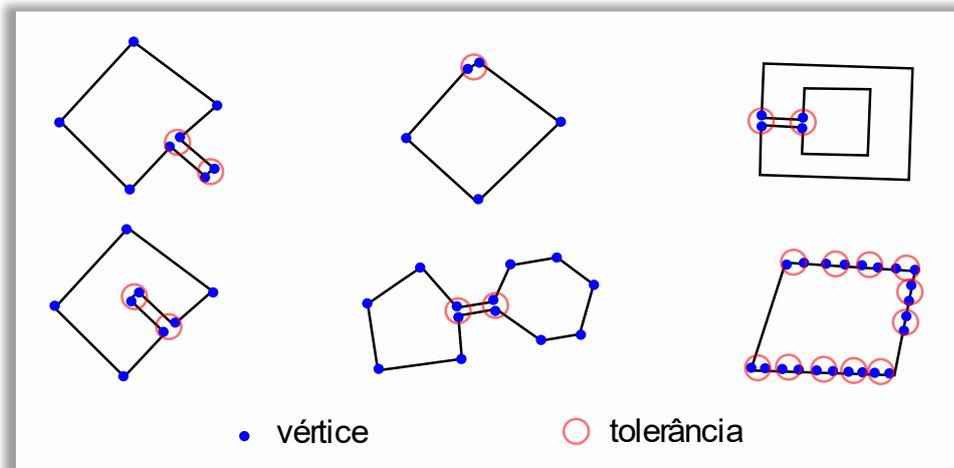
De acordo com Oosterom et al. (2005), uma geometria é dita “limpa” quando nenhum ponto de uma geometria está dentro de uma tolerância (ϵ) de qualquer outro ponto ou segmento (que não o contém).

O conceito de geometria limpa pode ser aplicado tanto para linhas quanto polígonos. Alguns problemas causadores de geometria não-limpa (ou poluída) são:

- Vértices próximos a arestas;
- Efeito ponta (ou molhe);
- Efeito ponte (interna ou externa);
- Pontos excessivos; e
- Pontos duplicados (repetidos).

A Figura 30 apresenta alguns exemplos de geometrias “não-limpa”, considerando-se uma tolerância (ϵ).

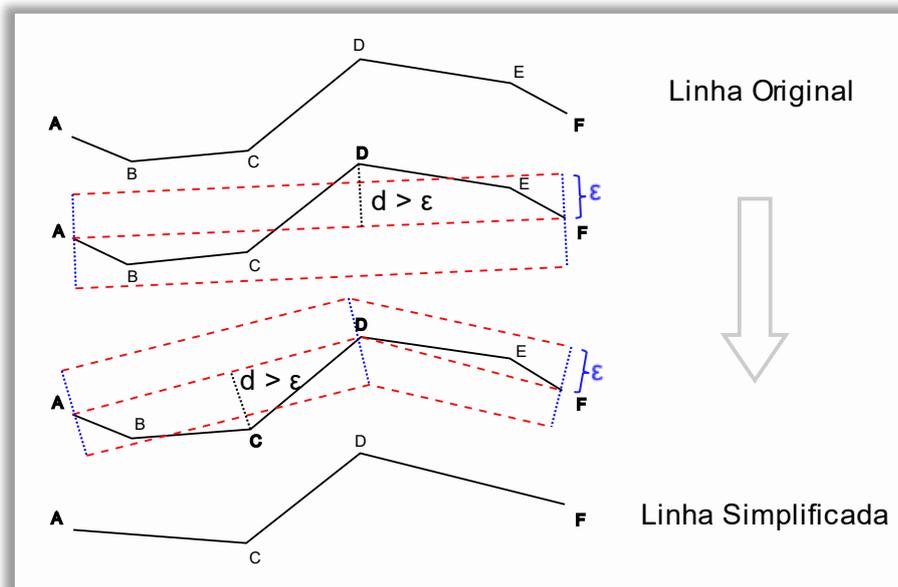
Figura 30 - Exemplo de geometrias não-limpas



Fonte: O autor (2021).

Segundo Andrade (2018), uma solução para a remoção de vértices em excesso é a utilização de algoritmos de simplificação da geometria, como o de Douglas-Peucker (Figura 31).

Figura 31 - Simplificação por Douglas-Peucker



Fonte: Adaptado de Crespo et al. (2014).

Nos casos em que a simplificação de Douglas-Peucker não eliminar os vértices dentro da tolerância, um vértice médio pode ser calculado, substituindo os vértices anteriores e eliminando segmentos pequenos.

O conceito da geometria limpa está relacionado aos princípios de generalização cartográfica, sendo a tolerância determinada conforme a escala de mapeamento.

O procedimento de limpeza das geometrias tem vários efeitos positivos podendo ser citados a redução de espaço para armazenamento dos dados e o ganho de eficiência no processamento computacional.

3.4.3.1.5 Geometria multiparte

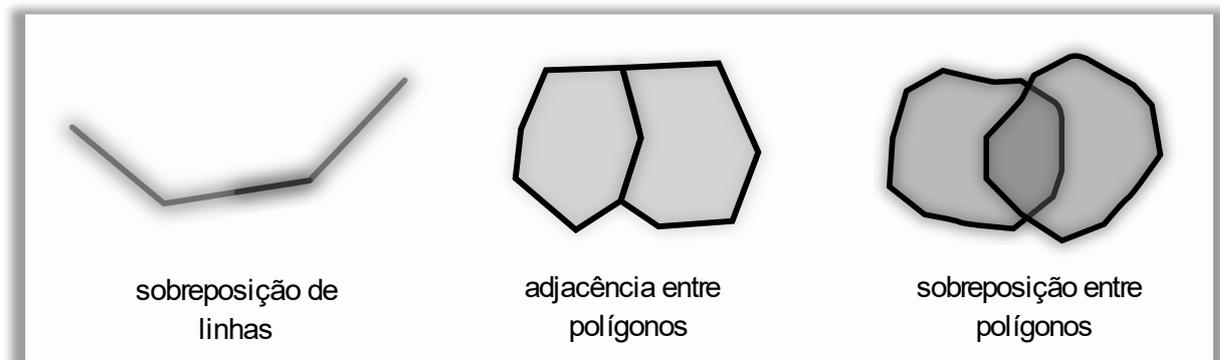
Uma geometria multiparte é, de fato, uma coleção de geometrias simples (partes-simples). Se ela contém apenas um tipo de geometria simples, ela pode ser classificada em *Multipoint*, *Multilinestring* e *Multipolygon* (Herring, 2011; Wadembere & Ogao, 2014).

Um exemplo de aplicação de geometria multiparte é utilizado para representar o estado do Havaí, nos Estados Unidos da América, onde várias ilhas representadas por polígonos compõe um único objeto do tipo multipolígono.

A utilização de geometrias multiparte deve estar previsto no modelo do banco de dados para representar feições descontínuas em um único objeto.

Regras topológicas entre as partes simples de uma geometria multiparte devem ser verificadas. A Figura 32 apresenta alguns exemplos de inconsistências entre as partes simples em geometrias do tipo multilinha e multipolígono.

Figura 32 - Exemplo de inconsistências em geometrias multipartes



Fonte: O autor (2021).

Andrade (2018) explica que a utilização desnecessária de geometrias multipartes pode levar a lentidão na execução de consultas espaciais, considerando-se que, em geral, as estruturas de indexação espacial são baseadas no conceito de mínimo retângulo envolvente (*bounding box*), fazendo que a primeira fase de uma consulta espacial (determinação dos candidatos) seja rápida, entretanto, a segunda fase da consulta (filtragem) fique mais demorada, perdendo performance computacional, devido à complexidade das geometrias.

Assim, para geometrias multiparte, é computacionalmente mais vantajoso desagregar geometrias multipartes em partes-simples, além de recriar a indexação espacial para que, ao invés de considerar um grande *bounding box*, sejam criados retângulos menores para as partes simples (Andrade, 2018; Laggner & Orthen 2014).

3.4.3.1.6 Geometria com ângulos pequenos

A incidência de ângulos pequenos pode estar relacionada a problemas na aquisição das geometrias do tipo linha e polígono, sendo também causado por processos automáticos de geoprocessamento.

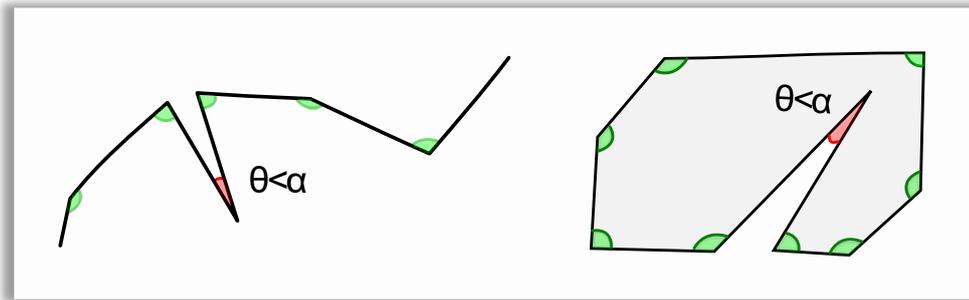
Em geral, as feições geográficas se configuram como formas “suaves”, onde as “pontas” são mais raras de serem observadas no mundo real.

Andrade (2018) aponta que os ângulos pequenos estão relacionados ao conceito de geometria limpa, mas como nem sempre o ângulo pequeno é detectado observando-se apenas uma distância de tolerância, o critério de verificação dos ângulos também se torna necessário.

A verificação consiste em calcular o menor ângulo entre segmentos adjacentes da sequência de pontos que compõe as geometrias do tipo linhas, bem como para os anéis que definem as geometrias do tipo polígono.

A Figura 33 apresenta exemplos de geometrias com ângulos menores que o tolerado ($\theta < \alpha$). Tal parâmetro pode ser especificado no modelo de dados, de acordo com cada classe.

Figura 33 - Exemplo de geometrias com ângulos pequenos



Fonte: O autor (2021).

3.4.3.1.7 Geometria fora das medidas

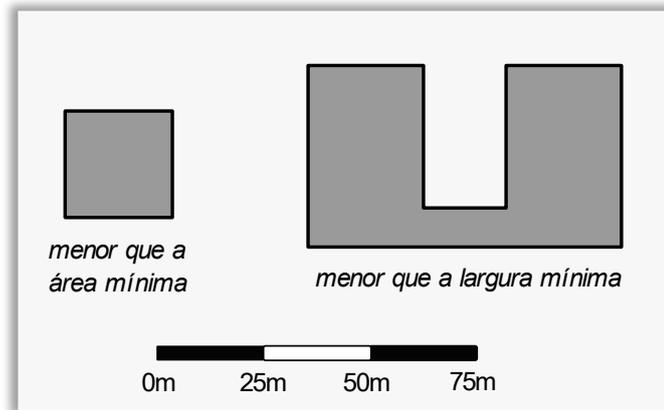
As geometrias das feições são utilizadas para representar por planimetria a sua localização espacial. Logicamente, espera-se que essas geometrias tenham medidas mínimas e máximas toleráveis em relação ao que é observado na natureza.

Feições do tipo linha e polígono podem ter suas medições (comprimento, área, perímetro etc.) comparadas com parâmetros esperados (máximos e mínimos) para cada tipo classe de feições.

Na prática, as medições são comparadas com valores mínimos estabelecidos para uma escala de mapeamento, a fim de identificar geometrias que possuem comprimento ou área abaixo de um valor limite e que não estão adequadas àquela escala.

Uma outra medida que deve ser verificada para polígonos é a largura mínima, isto é, a menor distância entre segmentos oposto de um polígono. A Figura 34 apresenta situações de polígonos que não atendem às medidas mínimas, considerando-se como área e largura mínimas de 1.250m² e 25m, respectivamente.

Figura 34 - Exemplo de geometrias fora das medidas mínimas



Fonte: O autor (2021).

Feições com geometrias que não atendem às medidas especificadas, podem ser generalizadas para uma classe do tipo ponto, quando houver. Outras soluções são: a remodelagem, dissolução com geometria adjacente ou, até mesmo, a eliminação.

Vale ressaltar que a verificação das medidas das geometrias para a identificação de inconsistências relacionadas aos valores esperados para determinados objetos do mundo real, ainda é pouco aplicada na produção cartográfica.

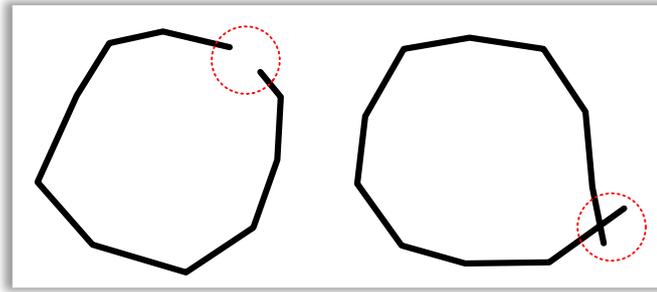
Observa-se também o papel fundamental da avaliação dessa medida para verificação do nível de generalização de um conjunto de dados. Segundo Shen & Ai (2017), a generalização é a questão-chave em aplicações que dependem da escala, como consultas espaciais em diferentes resoluções, integração de infraestruturas e rápida transmissão de dados.

3.4.3.1.8 Fechamento de anel linear

Um anel linear (*LinearRing*) é um tipo linha (*Linestring*) onde o último ponto coincide com o primeiro. O anel linear é a estrutura utilizada para representar isolinhas.

O fechamento de anel linear consiste basicamente em verificar se os pontos inicial e final são iguais. A Figura 35 apresenta alguns exemplos de erro de fechamento de anel linear.

Figura 35 - Exemplos de erro de fechamento de anel linear



Fonte: O autor (2021).

A exceção é admitida quando os pontos inicial e final de uma linha tocarem a borda da área de mapeamento (moldura), onde a linha é descontinuada.

3.4.3.2 Validação intraclasses (interna)

A validação intraclasses corresponde a verificação de regras topológicas entre as feições de uma mesma classe. Dependendo da modelagem dos dados, as seguintes checagens podem ser realizadas:

1. Geometrias duplicadas (repetidas);
2. Sobreposição entre geometrias;
3. Cruzamento entre linhas;
4. Lacunas entre polígonos;
5. Feição retalhada;
6. Ponta de linha solta;
7. Ausência de ponto de conectividade; e
8. Estrutura de rede.

3.4.3.2.1 Feições com geometrias duplicadas (repetidas)

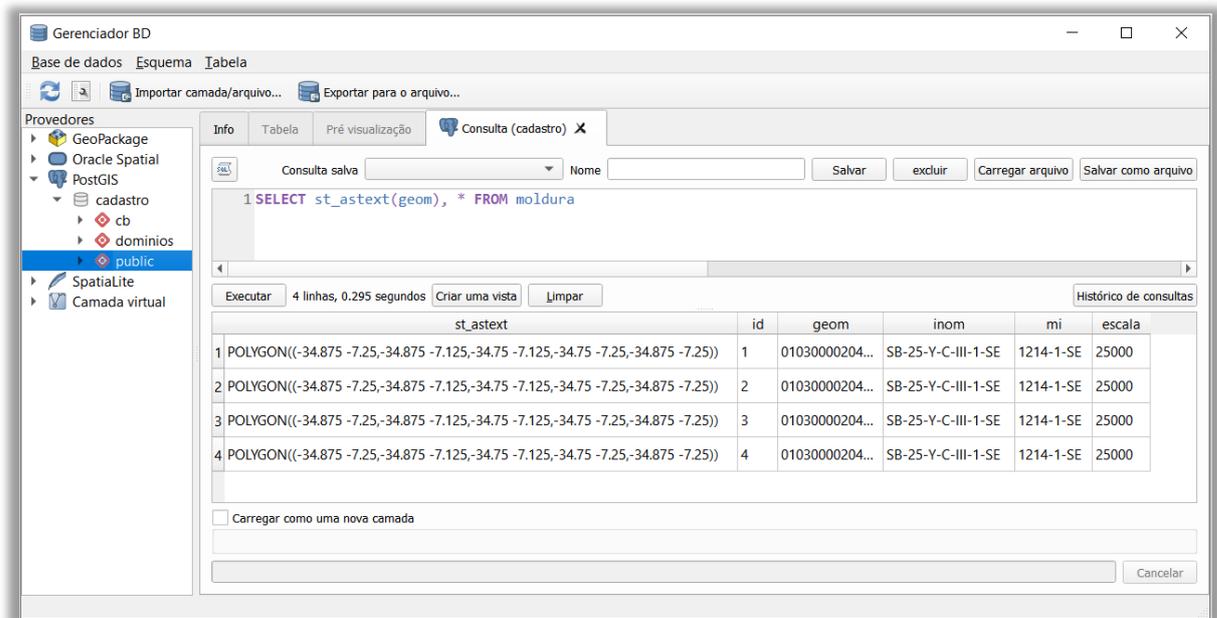
Geometrias exatamente iguais para diferentes objetos (registros no banco de dados) configuram uma inconsistência quando essas geometrias são utilizadas para representar uma mesma feição em um determinado momento.

Quando os atributos dessa feição são iguais para todos os objetos duplicados, os registros repetidos podem ser eliminados por processo automático.

Já no caso de existirem atributos diferentes, o operador necessita selecionar quais atributos devem permanecer.

A Figura 36 apresenta um exemplo de objetos repetidos que se referem a uma mesma feição. No caso da figura, a mesma feição foi inserida 4 vezes no banco de dados.

Figura 36 - Exemplos de geometrias duplicadas (repetidas)



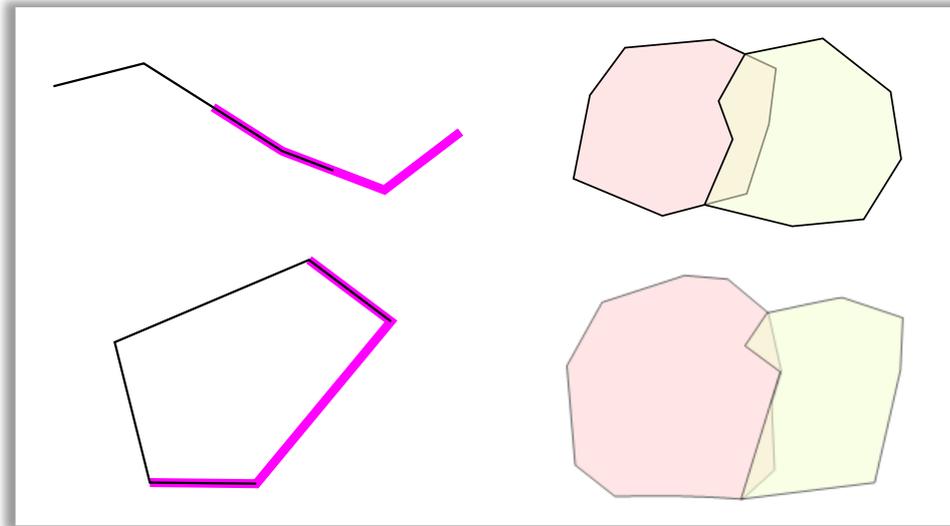
Fonte: O autor (2021).

3.4.3.2 Sobreposição entre geometrias

A sobreposição ocorre quando os interiores de duas geometrias distintas se interceptam e essa interseção tem a mesma dimensão das geometrias originais.

A Figura 37 apresenta situações de sobreposição para linhas e polígonos de uma mesma classe.

Figura 37 - Exemplos de sobreposição entre geometrias de uma mesma classe



Fonte: O autor (2021).

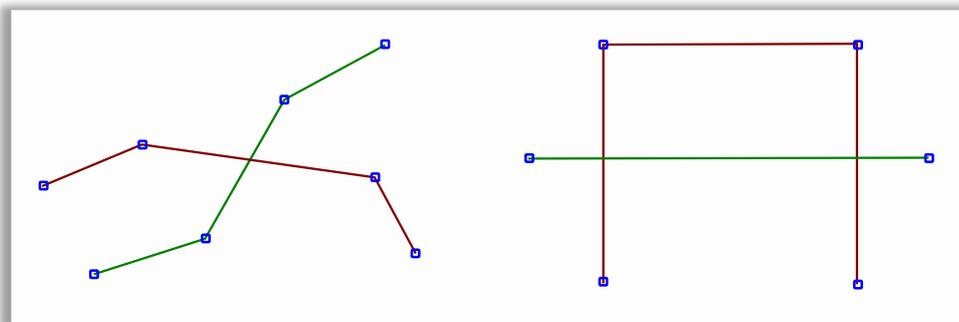
As geometrias com sobreposição devem ser consertadas de tal forma que as interseções sejam recortadas, eliminando-se todas as sobreposições. Ferramentas de geoprocessamento para determinar interseções e realizar diferenças entre geometrias podem ser utilizadas para este fim.

3.4.3.2.3 Cruzamento entre Linhas

O cruzamento de linhas pode ser identificado pelo operador topológico *cruza*, retornando *True* se o interior das linhas se interceptarem.

O cruzamento entre linhas pode ocorrer em um ou mais pontos de interseção. A Figura 38 mostra dois exemplos de cruzamento entre linhas.

Figura 38 - Exemplos de cruzamentos entre linhas



Fonte: O autor (2021).

Linhas que se cruzam devem ser seccionadas nos pontos de interseção, mas em casos específicos de geometrias que estejam representadas no espaço tridimensional (3D), o cruzamento entre linhas pode ser aceito, quando previsto na modelagem dos dados.

3.4.3.2.4 Lacunas entre polígonos

A verificação de lacunas (ou vazios) entre polígonos é bastante utilizada para classes que representam a cobertura terrestre (França et al., 2018), bem como municípios de um estado, lotes de uma quadra, ou qualquer outro conjunto de feições que compõe uma agregação espacial que, de acordo com a modelagem de dados, deve ter área completa, não aceitando vazios dentro dela.

A lacuna (ou em inglês, *gap*) geralmente é ocasionada por processos manuais de aquisição vetorial. Algumas vezes, esse tipo de inconsistência passa despercebida quando visualizada em escalas menores. A Figura 39 apresenta alguns exemplos de lacunas entre polígonos.

Figura 39 - Exemplos de lacunas entre polígonos



Fonte: O autor (2021).

As lacunas podem ser identificadas automaticamente através da operação da diferença da camada de polígonos por uma feição de referência, que pode ser o

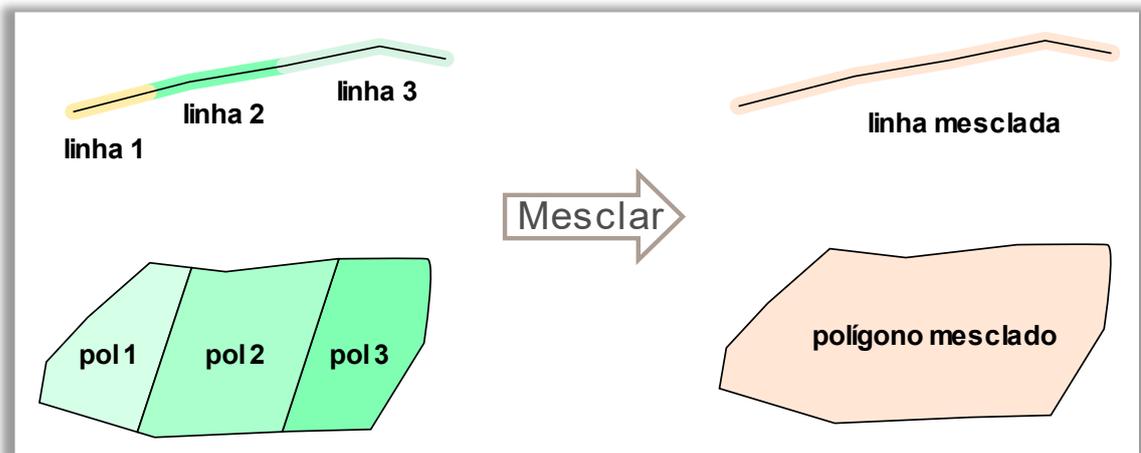
polígono de enquadramento da área de mapeamento (moldura) ou o seu próprio envoltório côncavo.

3.4.3.2.5 Feição retalhada

Feição retalhada é o nome dado a um grupo de feições adjacentes que possuem o mesmo conjunto de atributos, representando um único objeto.

A Figura 40 exemplifica dois casos de feição retalhada para linha e polígono. Em ambos, deve ser aplicado o processo de mesclar (ou dissolver) geometrias.

Figura 40 - Exemplos de feições retalhadas

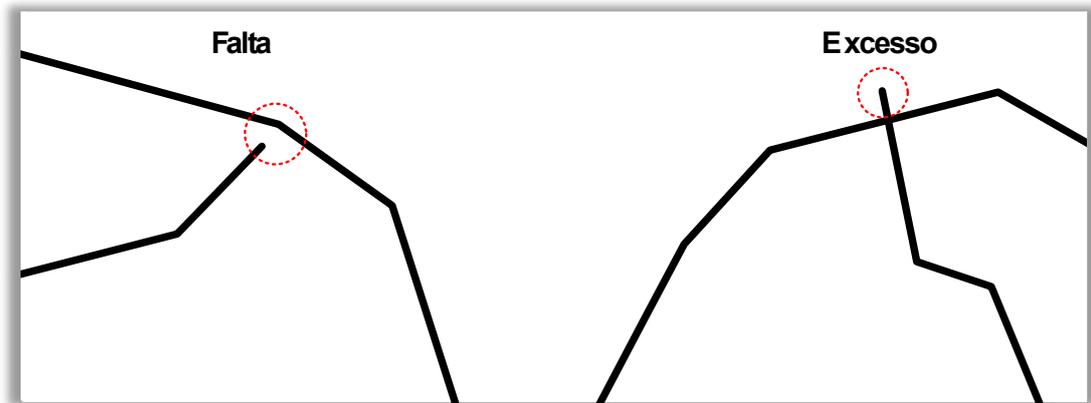


Fonte: O autor (2021).

3.4.3.2.6 Ponta de linha solta

Uma ponta de linha solta ocorre quando a extremidade de uma linha não se conecta à geometria de outra linha. As linhas soltas podem ser ocasionadas por “falta” ou “excesso” no comprimento da linha (Figura 41).

Figura 41 - Exemplos de ponta de linha solta



Fonte: O autor (2021).

Quando a linha solta é ocasionada por “excesso”, a inconsistência de cruzamento entre linhas também é identificada, o que não ocorre com linhas com “falta”.

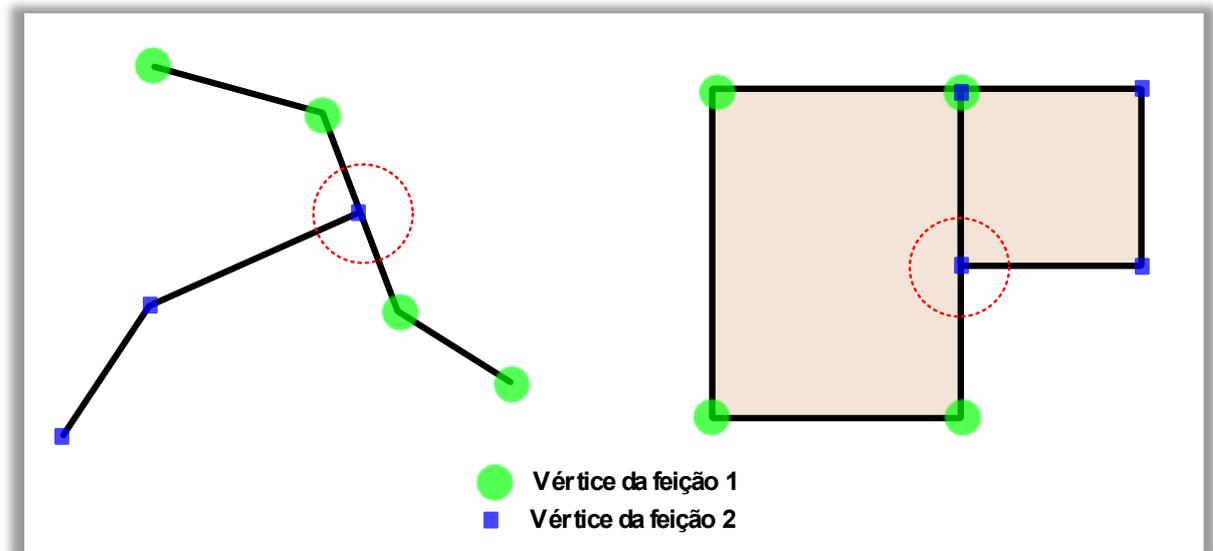
Para se evitar linhas soltas, é necessário, antes do processo de aquisição vetorial, a correta configuração da aderência entre feições (para vértices e/ou segmentos).

3.4.3.2.7 Ausência de ponto de conectividade

O ponto de conectividade entre duas geometrias corresponde ao vértice de interseção que deve constar em ambas. Nos casos em que esse ponto não é vértice de uma das geometrias, ou seja, pertencendo somente ao segmento, ocorre a inconsistência chamada de “ausência de ponto de conectividade”.

A Figura 42 apresenta casos de ausência de ponto de conectividade para linhas e polígonos.

Figura 42 - Exemplos de ausência de ponto de conectividade



Fonte: O autor (2021).

A importância do ponto de conectividade está evidenciada na garantia da conectividade entre as geometrias, mesmo quando elas são reprojctadas para um SRC diferente daquele no qual foram construídas, seja em coordenadas projetadas ou geodésicas (geográficas).

3.4.3.2.8 Estrutura de rede

Estrutura de rede é a denominação dada a uma classe de feições que tem como base o direcionamento de linhas para representar um fluxo unidirecional ou bidirecional.

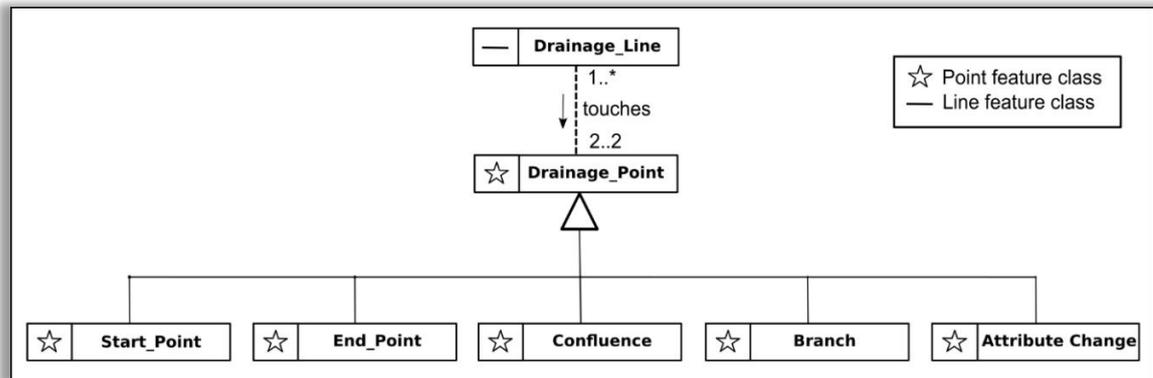
Alguns exemplos de aplicação de estrutura de rede são os trechos de drenagem fluvial, linhas de saneamento básico, linhas de transmissão de energia e telecomunicações cabeadas (fibra óptica).

França et al. (2020) analisaram a utilização de linhas unidirecionais para representar o fluxo da água em bacias hidrográficas, o qual é regido pela lei da gravidade. Em tal situação, os vértices das linhas correspondentes aos trechos de drenagem devem ser vetorizados e armazenados de tal forma que se iniciem de um ponto mais elevado (montante) e terminem em um ponto de menor cota (jusante).

Os pontos inicial e final das linhas de uma rede de drenagem definem os pontos de drenagem (Figura 43). Os pontos de drenagem, para um fluxo

unidirecional podem ser classificados em: ponto inicial; ponto final; confluência; ramificação; e ponto de mudança de atributo.

Figura 43 - Tipos de pontos de rede

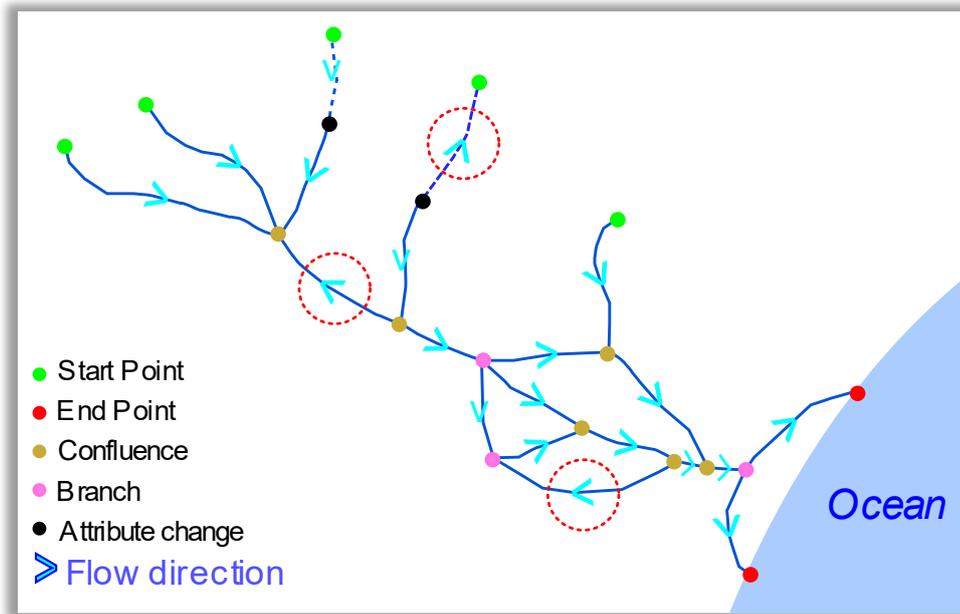


Fonte: França (2018).

- Ponto inicial (*start point*): ponto onde uma rede de drenagem começa, como uma fonte d'água, vertedouro ou algo similar;
- Ponto final (*end point*): ponto onde uma rede de drenagem termina, como uma foz, desembocadura, sumidouro ou algo similar;
- Confluência (*confluence*): ponto onde duas ou mais linhas de drenagem convergem, resultando em apenas uma única linha;
- Ramificação (*branch*): ponto onde uma linha de drenagem se divide e duas ou mais linhas;
- Mudança de atributo (*attribute change*): ponto onde ocorre mudança de características (atributos) da linha.

A Figura 44 apresenta exemplos de inconsistências na estrutura da rede de drenagem devido a erros no direcionamento das linhas.

Figura 44 - Exemplos de inconsistência na rede de drenagem



Fonte: O autor (2021).

Outro problema comum na rede de drenagem é o laço (*loop*), um tipo de inconsistência que fere a lógica da rede de drenagem, considerando-se que a água segue na direção do seu fluxo e não pode retornar para o mesmo ponto.

Portanto, para análises consistentes baseada na rede, é essencial garantir a correta orientação de todas as linhas, bem como a conectividade entre elas (França et al., 2020).

3.4.3.3 Validação interclasses (externa)

Na validação interclasses são verificadas regras topológicas entre as feições de classes distintas. Para que a execução desta validação seja efetiva, é essencial que a validação intraclasse e das geometrias sejam executadas preliminarmente.

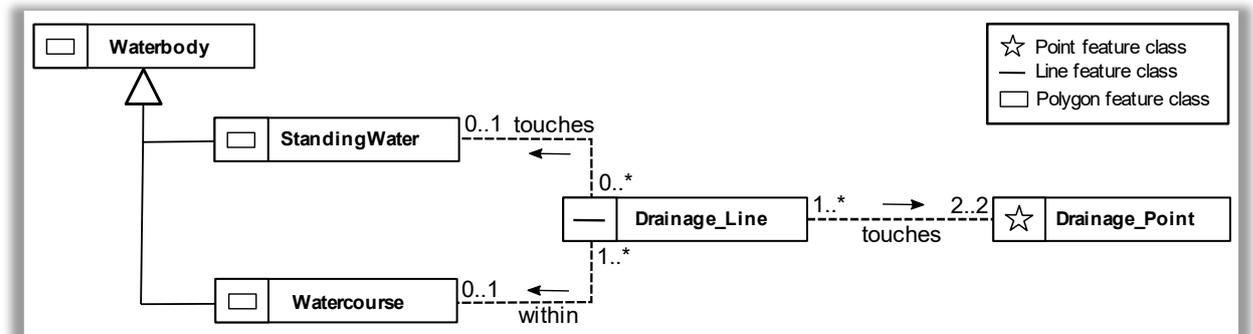
Os relacionamentos espaciais entre as feições de classes distintas podem ser indicados nos diagramas do modelo conceitual pelo seu nome convencional (por exemplo, *contains*, *touches* etc.) ou, em casos mais específicos, por restrições baseadas diretamente nos resultados da matriz DE-9IM (Lizardo & Davis Jr., 2017).

Os relacionamentos espaciais são caracterizados pela cardinalidade. A cardinalidade representa o número de instâncias de uma classe que pode estar associadas a instâncias de outra classe (Borges et al., 2005).

A notação de cardinalidade adotada pelo modelo OMT-G é baseada na empregada pela UML (Davis Jr. & Laender, 2005), onde são indicados o valor mínimo e máximo, separados por dois pontos consecutivos. O asterisco é utilizado quando se deseja simbolizar o valor máximo como “muitos”, por exemplo em: 1..*, onde lê-se: “um a muitos”.

A Figura 45 apresenta um diagrama de modelo conceitual em OMT-G para representar algumas classes de uma bacia hidrográfica. Neste diagrama é possível observar os principais relacionamentos espaciais entre as classes e suas cardinalidades.

Figura 45 - Relacionamentos espaciais entre classes e suas cardinalidades



Fonte: França et al. (2020).

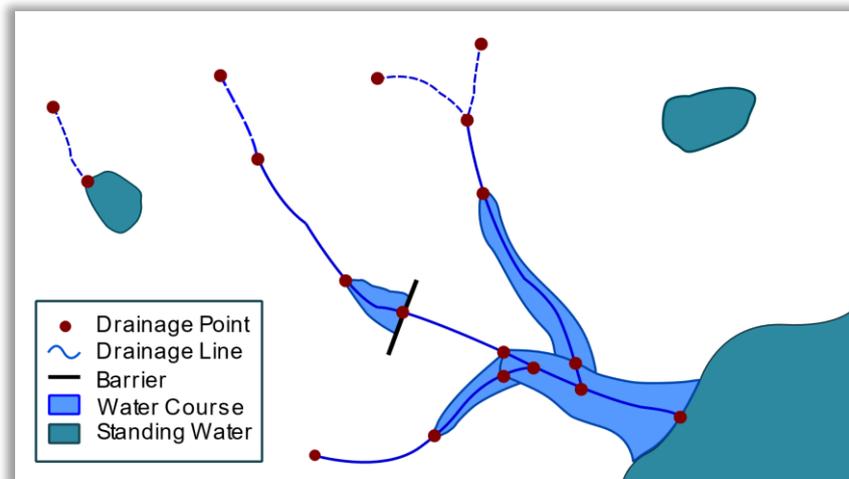
Da Figura, observa-se que as classes “Trecho de Massa d’água – *Watercourse*” e “Massa d’água – *StandingWater*” se generalizam em “Corpo d’água – *Waterbody*”. O Quadro 10 apresenta a descrição das classes apresentadas na Figura 45. A Figura 46 ilustra alguns casos dessas classes.

Quadro 10 - Descrição das classes hidrográficas

Classe	Geometria	Descrição
Ponto de Drenagem <i>Drainage_Point</i>	Ponto	Ponto de conectividade entre linhas de drenagem definidos pelos seus pontos inicial e final.
Trecho de Drenagem <i>Drainage_Line</i>	Linha	Representação linear do fluxo principal de um corpo d'água.
Corpo d'água <i>Waterbody</i>	Polígono	Superfície com acumulação significativa de água para uma determinada escala. Ela pode ser classificada em “Massa d'água – <i>Standing Water</i> ” (lagos, lagoas, oceanos etc.) e “Trecho de massa d'água – <i>Water Course</i> ” (rios, córregos, canais etc.), de acordo com a existência de fluxo de água.

Fonte: França et al. (2020).

Figura 46 - Configuração dos relacionamentos entre classes hidrográficas



Fonte: O autor (2021).

Embora o modelo conceitual torne explícito os principais relacionamentos espaciais, tornando-os mais compreensíveis, outras regras podem estar implicitamente definidas como parte do significado das feições, podendo ser deduzidas do modelo (Lizard & Davis Jr., 2017).

Do diagrama da Figura 45, por exemplo, está definida a seguinte relação: um trecho de drenagem (*drainage_line*) deve estar dentro de 0 ou 1 trecho de massa d'água (*watercourse*). No entanto a mesma relação, no sentido oposto, pode ser

interpretada, ou seja, um trecho de massa d'água deve conter 1 ou muitos trechos de drenagem.

Muitas regras topológicas podem ser estabelecidas de forma a garantir a integridade dos dados, podendo o número de regras não ser exaustivo (França et al., 2020).

O Quadro 11 lista, baseadas nas classes apresentadas na Figura 45, regras topológicas que devem (ou não) ocorrer, considerando-se determinada cardinalidade.

Quadro 11 - Exemplos de regras para validação interclasse

Regra	Classe 1	Deve	Topologia	Classe 2	Cardinalidade
01	<i>Drainage Line</i>	<i>yes</i>	<i>touches</i>	<i>Standing Water</i>	0..1
02	<i>Drainage Line</i>	<i>yes</i>	<i>Within</i>	<i>Watercourse</i>	0..1
03	<i>Drainage Line</i>	<i>yes</i>	<i>touches</i>	<i>Drainage Point</i>	2..2
04	<i>Watercourse</i>	<i>yes</i>	<i>contains</i>	<i>Drainage Line</i>	1..*
05	<i>Drainage Point</i>	<i>yes</i>	<i>touches</i>	<i>Drainage Line</i>	1..*
06	<i>Drainage Point</i>	<i>no</i>	<i>Within</i>	<i>Standing Water</i>	1..*
07	<i>Drainage Line</i>	<i>no</i>	<i>Crosses</i>	<i>Standing Water</i>	1..*

Fonte: França et al. (2020).

Observa-se que as regras 04 e 05 correspondem respectivamente às regras 02 e 03 no sentido oposto. Já as regras 06 e 07 são exemplos de regras para checar situações que não devem ocorrer, sendo regras implícitas baseadas no entendimento do mundo real.

França et al. (2020), afirmam que as regras explícitas no modelo conceitual podem não ser efetivas para identificar inconsistências topológicas entre classes, apontando a importância de realizar a verificação das regras no sentido contrário, bem como outras situações implícitas que não devem ocorrer na aquisição dos dados.

Portanto, as regras topológicas que não estão descritas no esquema conceitual devem constar no dicionário de dados, de forma a possibilitar critérios de checagem e a garantia da integridade dos dados. A documentação dessas regras

nas especificações técnicas do conjunto de dados, principalmente quando requeridas em contratos, eliminará qualquer subjetividade no processo de avaliação da qualidade.

Em termos práticos, as inconsistências na validação interclasse podem ser identificadas por consultas em SQL. A Figura 47 mostra um exemplo de consulta SQL do PostGIS, que pode ser aplicada à regra 04 do Quadro 11, para identificar os polígonos que não contém pelo menos uma linha.

Figura 47 - Consulta SQL para identificar inconsistências topológicas entre classes

```
1 SELECT
2 V.id AS id,
3 V.geom AS geom,
4 'polygon should be related to at least 1 line' AS erro_msg
5 FROM(
6     SELECT
7         C.id AS id,
8         C.geom AS geom,
9         sum(C.relation::int) AS n_elems
10    FROM(
11        SELECT
12            T1.id AS id,
13            T1.geom AS geom,
14            st_contains(T1.geom, T2.geom) AS relation -- analyse spatial relation
15        FROM polygon_class T1, line_class T2
16    ) C
17    GROUP BY C.id, C.geom
18 ) V
19 WHERE n_elems = 0 -- analyse cardinality
```

Fonte: O autor (2021).

4 CONSISTÊNCIA SEMÂNTICA

Além dos elementos de qualidade já abordado anteriormente e previstos na ISO 19.157, verifica-se a necessidade de um tipo específico de inspeção quando existem regras lógicas entre os atributos, bem como entre atributo e propriedade topológica da feição.

Devido a esta lacuna nas especificações de controle de qualidade de dados geoespaciais, este trabalho propõe a abordagem do elemento de qualidade denominado de “consistência semântica”, que visa avaliar as seguintes situações:

- Regras entre atributos de uma feição; e
- Regra entre atributo e topologia de uma feição.

A interpretação da Consistência Semântica difere dos elementos de qualidade da Acurácia Temática da ISO (2013). Para os dois elementos de qualidade relacionados a atributos: Acurácia de Atributos Quantitativos (DQ_QuantitativeAttributeAccuracy) e Correção de Atributos Qualitativos (DQ_NonQuantitativeAttributeCorrectness), o método de validação emprega análises estatísticas, comparando-se os dados com valores coletados no campo ou outras fontes de referência, como apresentado por Mõisja et al. (2018).

Por outro lado, a Consistência Semântica verifica regras lógicas entre dois atributos ou entre atributo e topologia de uma feição, utilizando-se de processos automáticos (rotinas de lógica matemática), sem a necessidade de valores de referência.

A Consistência Semântica também difere da Consistência de Domínio (DQ_DomainConsistency) da ISO (2013), pois, neste caso, é verificada a aderência às restrições de valores definidos no domínio de cada atributo, ou seja, o domínio corresponde a uma regra aplicada individualmente aos valores de entrada de um atributo, sem estudo condicional entre dois ou mais atributos e, muito menos, com propriedades topológicas da feição.

O termo “semântica” não é novidade na ciência da computação, já sendo empregado por Schewe & Thalheim (2008) no sentido de capturar o “significado” dos dados com os relacionamentos inerentes a um determinado entendimento organizacional.

Anteriormente, Elmasri & Navathe (2000) apresentaram o conceito de “restrições de integridade semântica”, as quais deveriam ser especificadas para os

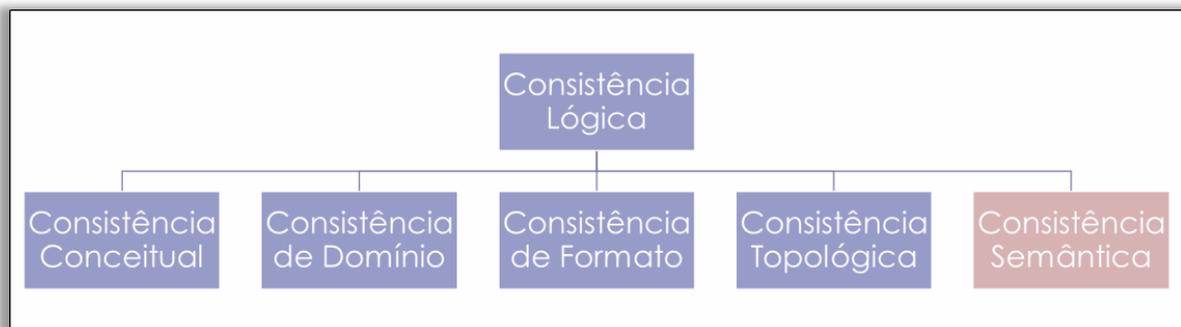
atributos e implementadas em banco de dados relacionais através de mecanismos como gatilhos (*triggers*).

Servigne et al. (2000) utilizou o termo “consistência topo-semântica” com um subconjunto da consistência lógica, entretanto, nesse caso a semântica esteve mais direcionada aos relacionamentos topológicos entre dois objetos, sem envolver regras entre os atributos das feições.

Neste trabalho, portanto, a Consistência Semântica tem um significado mais abrangente, pois se refere aos relacionamentos entre dois atributos ou entre atributo e topologia de uma feição.

A Figura 48 esquematiza o elemento de qualidade proposto dentro da categoria consistência lógica.

Figura 48 – Elemento de qualidade consistência semântica



Fonte: O autor (2021).

4.1 REGRAS ENTRE ATRIBUTOS

Esta situação é verificada quando existe uma regra condicional (se, então) entre dois ou mais atributos. Ou seja, quando um atributo satisfaz uma condição A, então outro atributo deve satisfazer uma condição B.

Um exemplo de regra entre atributos é: se uma feição da classe massa d’água tem o atributo *tipo_massa_dagua* igual a “oceano” ou “enseada”, então o atributo *salinidade* de ter o valor correspondente a “salgada”.

Outro exemplo é: se um atributo *tipo* for preenchido como “outro”, então o atributo *descricao* deve ser preenchido (ou seja, não nulo).

Regras bicondicionais do tipo “se e somente se” também podem ser utilizados, desde que se verifique as duas condições (ida e volta).

4.2 REGRAS ENTRE ATRIBUTOS E TOPOLOGIA

Nos casos em que os atributos estão associados às características topológicas (da própria feição ou de outra a que estiver relacionada), seus valores devem ser confrontados com valores extraídos a partir de consultas espaciais.

Um exemplo prático de regra é: uma feição da classe **parcela** deve ter o atributo *codigo_setor* coerente com uma feição da classe setor, ou seja, “dentro” do respectivo polígono de setor que o contém.

Outro exemplo é a classe **ponto_de_drenagem** que possui o atributo *relacionado* para dizer com qual feição de outra classe ele se relaciona (por exemplo, eclusa, barragem, comporta, queda d’água, lago, lagoa, foz, represa etc.).

Atributos que armazenam informações de uma estrutura de rede, como é o caso da codificação de Pfafstetter, embutindo a informação topológica de uma rede de drenagem como atributo em cada feição (Romanholi e Queiroz Filho, 2018), podem ter a hierarquização dos trechos de drenagem de uma bacia hidrográfica avaliados nesse elemento de qualidade.

Tanto para o relacionamento entre atributos quanto entre atributo e topologia, o problema da consistência semântica está em mapear as regras para caracterização dos estados das instâncias do banco de dados. Exemplos de regras para o mapeamento topográfico estão detalhadas no capítulo de metodologia deste trabalho, onde as medidas de qualidade constam nos Apêndices B e C.

4.3 LIDANDO COM EXCEÇÕES

As restrições de integridade definem regras que os dados devem seguir. Elas são pensadas de forma que todos os objetos de uma classe de feições sigam essas mesmas regras. Isso significa que é esperado que as feições do mundo real tenham características pré-definidas. Tal hipótese, de fato, é bastante inflexível (Servigne et al., 2000).

No processo de validação, as exceções são apontadas por intermédio de consultas espaciais, onde o operador pode intervir, caso julgue necessário. Não é raro ter que lidar com exceções em um conjunto de dados geográficos. Podem ser citados os seguintes exemplos:

- Edificação não pode estar dentro de massa d'água, no entanto, em algumas regiões na Amazônia é bastante comum observar diversas construções flutuantes.
- Ponte deve se relacionar com rodovia ou ferrovia (cruza ou sobrepõe), mas podem existir raras situações de pontes abandonadas ou em construção sem qualquer relacionamento com algum meio de transporte, daí a necessidade de que o atributo *situacao_fisica* seja preenchido, por exemplo, como “Abandonada” ou “Destruída”.
- Edificação (polígono) deve estar dentro de uma parcela (lote), no entanto existem casos de edificação construídas em mais de uma parcela ou parcialmente fora do lote ao qual está relacionado.

Apesar da possibilidade de existir diversos tipos de exceções, isso não significa que se deva deixar de impor regras para os dados geoespaciais, no entanto, é necessário um meio de lidar com essas exceções.

Para os casos gerais, as regras de integridade topológica devem ser detectadas e corrigidas (Siejka et al., 2013). Nos casos de exceção, não há motivo de correção, entretanto essas situações devem ser reportadas no relatório de qualidade e armazenadas em uma base de exceções (Servigne et al., 2000).

Servigne et al. (2000) apresentaram duas formas de lidar com exceções. A primeira é definir cada restrição com um atributo de exceção e reportar cada caso assim que detectado. A base de exceções é então construída pouco a pouco, adicionando uma lista de casos para cada restrição.

A segunda forma de lidar com exceções é utilizar uma lista prévia de exceções já conhecidas, e classificar os casos que violam as regras do modelo durante processo de validação da consistência topológica e semântica.

5 AVALIAÇÃO DA CONSISTÊNCIA LÓGICA

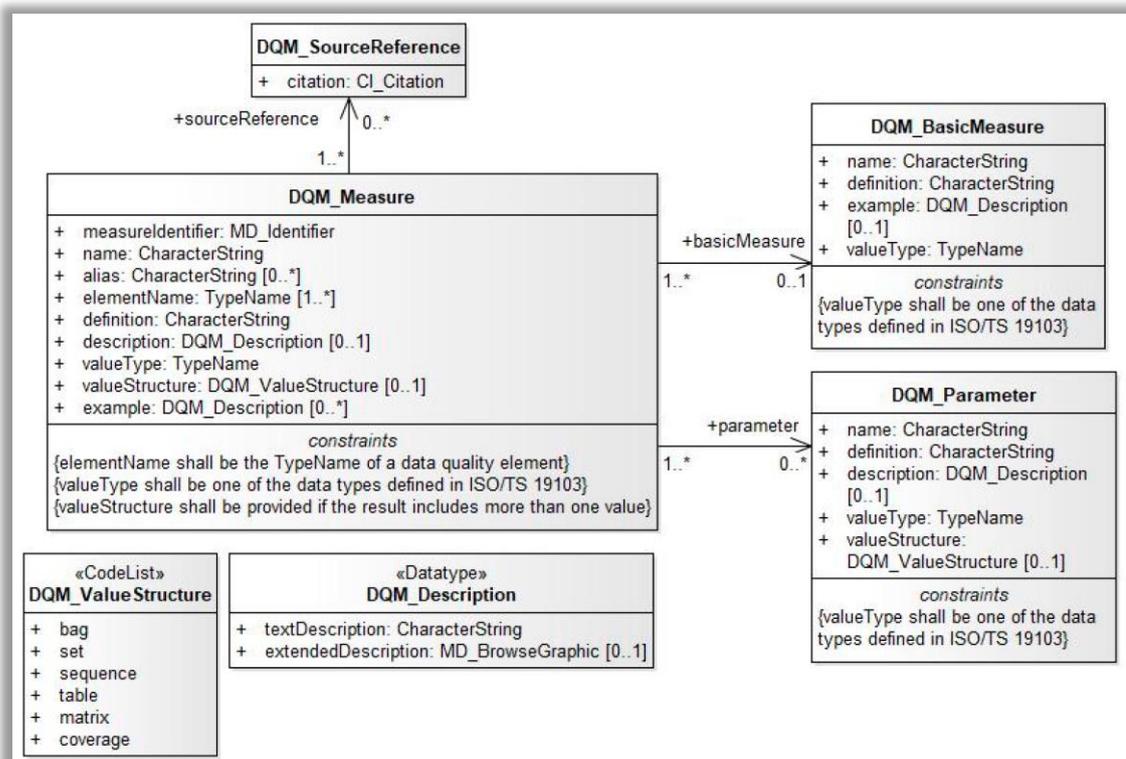
A mensuração da qualidade dos dados geoespaciais possibilita que o usuário de SIG possa avaliar a adequação desses dados ao seu propósito, verificando o potencial de cumprir determinados requisitos necessários para uma finalidade pretendida (Servigne et al., 2006).

A avaliação dos elementos de qualidade é baseada nas medidas de qualidade. Tais medidas são utilizadas para guiar o processo de avaliação de cada tipo de produto.

5.1 MEDIDAS DE QUALIDADE PARA A CONSISTÊNCIA LÓGICA

A ISO 19157 (2013) padroniza que cada medida de qualidade deva ser apresentada por meio de descritores (componentes que descrevem uma medida de qualidade). Esses descritores são apresentados na Figura 49 e Quadro 12, por meio dos atributos da classe DQM_Measure e classes associadas.

Figura 49 - Descritores da medida de qualidade de dados geoespaciais



Quadro 12 - Descrição dos atributos da classe DQM_Measure

Atributo	Descrição
<i>measureIdentifier</i>	Valor que identifica unicamente a medida.
<i>name</i>	Nome da medida de qualidade aplicada aos dados especificados pelo escopo.
<i>alias</i>	Nome alternativo, abreviatura ou nome curto para a mesma medida de qualidade.
<i>elementName</i>	Nome do elemento de qualidade ao qual se refere a medida. Mais de um elemento pode ser usado.
<i>definition</i>	Definição do método para calcular o valor da medida de qualidade.
<i>description</i>	Todas as fórmulas e/ou esquemas necessários para obter o resultado ao aplicar a medida.
<i>valueType</i>	Tipo de valor para informar o resultado da medida. Exemplos: texto, número real.
<i>valueStructure</i>	Possível estrutura para o valor da medida. Isso ocorre em resultados complexos, como uma lista ou matriz.
<i>basicMeasure</i>	Nome da medida básica da qual se deriva a medida descrita.
<i>example</i>	Exemplo de uso da referida medida de qualidade.
<i>parameter</i>	Variável auxiliar utilizada pela medida de qualidade incluindo nome, definição e descrição. Mais de um parâmetro pode ser usado.
<i>sourceReference</i>	Referência externa caso a medida tenha sido documentada em outra fonte.

Fonte: DSG (2016).

O Quadro 13 apresenta e define os principais tipos de medidas básicas de qualidade da classe DQM_BasicMeasure.

Quadro 13 - Medidas Básicas de Qualidade (*basicMeasure*)

Medida básica	Definição	Tipo de valor	Exemplo
Indicador de erro	Indica que um elemento não está correto ou é falso.	Booleano	<i>False</i>
Indicador de acerto	Indica que um elemento está correto ou é verdadeiro.	Booleano	<i>True</i>
Contagem de erro	Número total de elementos errados em um conjunto de dados, definido no escopo do controle de qualidade dos dados.	Inteiro	11
Contagem de acerto	Número total de elementos corretos em um conjunto de dados, definido no escopo do controle de qualidade dos dados.	Inteiro	571
Taxa de erro	Quantidade de erros dividido pelo número de elementos que estão presentes no conjunto de dados, definidos no escopo.	Real	0,019

Taxa de acerto	Quantidade de acertos dividido pelo número de elementos que estão presentes no conjunto de dados, definidos no escopo.	Real	98,1%
----------------	--	------	-------

Fonte: DSG (2016).

Uma medida de qualidade DQM_Measure pode, na prática, ser representada por uma tabela de forma descritiva. O Quadro 14 apresenta um exemplo aplicado para a medida “conformidade de estrutura física”, explicitando as verificações necessárias para garantir a consistência de formato de um CDGV.

Quadro 14 - Medida de qualidade padronizada “conformidade de estrutura física”

Atributo	Descrição
<i>measureIdentier</i>	VCL100
<i>name</i>	Conformidade de estrutura física
<i>alias</i>	conf_estr_fisica
<i>elementName</i>	Consistência de formato
<i>definition</i>	Instrumento que indica se existe alguma inconsistência ou conflito na estrutura física de armazenamento dos dados.
<i>description</i>	Devem ser verificados erros ou conflitos na estrutura física do arquivo fornecido no formato indicado na especificação do produto. Itens a serem verificados: <ul style="list-style-type: none"> • Arquivo diferente da especificado (extensão do arquivo); • Arquivo corrompido (com falha na transmissão dos dados); • Sistema de Referência de Coordenadas (SRC) diferente da especificada; e • Codificação dos dados textuais diferente da especificada. Caso haja qualquer inconsistência nas etapas de verificação, o resultado da medida será <i>False</i> (não conforme), caso contrário a medida retorna <i>True</i> (conforme).
<i>basicMeasure</i>	Indicador de acerto
<i>valueType</i>	Booleano
<i>valueStructure</i>	Não é o caso
<i>example</i>	Exemplos de inconsistências: <ol style="list-style-type: none"> 1) Entrega de arquivo Shapefile ao invés de Geopackage; 2) Falha na leitura de algumas camadas pelo software; 3) Coordenadas em WGS84 ao invés de SIRGAS2000; e 4) Codificação em Latin1 (ISO-8859-1) ao invés de UTF-8.
<i>parameter</i>	Geopackage (.gpkg) SIRGAS2000

	UTF-8
<i>sourceReference</i>	-

Fonte: Adaptado de DSG (2016).

As medidas de qualidade aplicadas neste trabalho para a consistência topológica e semântica se encontram nos Apêndices A, B e C.

Neste trabalho, foi adotado o código do *measureIdentifier* com o prefixo VCL, de Validação da Consistência Lógica, seguido de um número relacionado a ordem em que as medidas de qualidade e seus subitens devem ser avaliados.

Essa codificação foi adotada para se garantir uma melhor padronização com o procedimento apresentado neste trabalho, tendo em vista seguir uma sequência mais adequada, atualizada e completa de validação da consistência lógica, se comparado com ISO (2013), DSG (2016) e IBGE (2020).

5.2 PROCEDIMENTO DE AVALIAÇÃO DE QUALIDADE

Na avaliação da consistência lógica de um CDGV, a estrutura física das classes e todas as feições (geometrias e atributos) devem ser inspecionadas por completo, considerando-se que computacionalmente é possível aplicar rotinas automáticas para a validação de regras lógicas.

O procedimento de avaliação deve seguir a sequência apresentada no Quadro 15, ressaltando que um elemento de qualidade pode possuir mais de uma medida, como é o caso da consistência topológica. Para cada medida de qualidade, o respectivo resultado de conformidade é apresentado na coluna à direita.

Quadro 15 - Sequência de Validação da Consistência Lógica (VCL)

Ordem	Elemento	Medida ID	Nome da Medida	Conformidade
1	Consistência de formato	VCL100	Conformidade de estrutura física	M = True (conforme)
2	Consistência conceitual	VCL200	Conformidade com o modelo de dados	M = True (conforme)
3	Consistência de domínio	VCL300	Valores de domínios conformes	M = True (conforme)
4	Consistência topológica	VCL410	Validação da Geometria	M = 0 (zero) inconsistências
5	Consistência topológica	VCL420	Validação intraclasse	M = 0 (zero)

				inconsistências
6	Consistência topológica	VCL430	Validação interclasse	M = 0* (zero) inconsistências
7	Consistência semântica	VCL510	Validação de regras entre atributos	M = 0* (zero) inconsistências
8	Consistência semântica	VCL520	Validação de regras entre atributo e topologia	M = 0* (zero) inconsistências

* Sujeito a casos de exceção.

Fonte: O Autor (2021).

5.3 RELATÓRIO DE QUALIDADE

O relatório de qualidade é a concretização do conceito de metaqualidade (do inglês, *metaquality*), o qual fornece informações sobre a avaliação da qualidade e seus resultados com objetivo de descrever a adequação do método de avaliação e a(s) medida(s) aplicadas (Møisja, 2018).

O resultado da avaliação de qualidade de um CDGV pode ser implementado como metadados em *Extensible Markup Language* (XML) ou documentado como relatório técnico de qualidade.

O relatório de qualidade por metadados é regido pela ISO 19115 (*Geographic Information – Metadata*), que em suas Parte1: *Fundamentals* (2014); Parte 2: *Extensions for acquisition and processing* (2019); e Parte 3: *XML Schema implementation for fundamental concepts* (2016), define o esquema de implementação XML para o modelo conceitual relacionado aos processos de controle de qualidade (Brodeur et al., 2020).

O Perfil de Metadados Geoespaciais do Brasil (MGB) (CONCAR, 2009) é o padrão de metadados adotado no Brasil, baseando-se na ISO 19115 e adequando-se as características de mapeamento no território nacional.

Uma das seções do Perfil MGB trata da qualidade dos dados geoespaciais, sendo compostos com elementos relacionados ao nível hierárquico, linhagem e o relatório de qualidade, propriamente dito.

A Figura 50 é um exemplo de metadados de qualidade na estrutura XML para a medida “conformidade de estrutura física” resultante da avaliação da consistência de formato.

Figura 50 - Extrato de relatório de Qualidade em XML

```

<report>
  <DQ_FormatConsistency>
    <nameOfMeasure>
      <gco:CharacterString>Conformidade de estrutura física</gco:CharacterString>
    </nameOfMeasure>
    <measureIdentification>
      <MD_Identifier>
        <code>
          <gco:CharacterString>VCL100</gco:CharacterString>
        </code>
      </MD_Identifier>
    </measureIdentification>
    <measureDescription>
      <gco:CharacterString>Deve ser verificado se existe algum erro de conflito na estrutura física do arquivo de acordo com o formato indicado no parâmetro.</gco:CharacterString>
    </measureDescription>
    <evaluationMethodType>
      <DQ_EvaluationMethodTypeCode codeListValue="directInternal">diretoInterno</DQ_EvaluationMethodTypeCode>
    </evaluationMethodType>
    <dateTime>
      <gco:DateTime>2020-12-17T15:48:34</gco:DateTime>
    </dateTime>
    <result>
      <DQ_ConformanceResult>
        <specification>
          <CI_Citation>

```

Fonte: Extraído de DSG (2016).

O relatório técnico de qualidade deve ser confeccionado para informar ao contratante do produto sobre as medidas de qualidade e os resultados dos procedimentos de avaliação.

Diferentemente do relatório por metadados, onde as informações em XML são destinadas, principalmente, a alimentar uma base de dados de qualidade do produto, o relatório técnico possibilita a documentação através de uma apresentação mais inteligível pelo ser humano, geralmente fornecido no formato PDF.

No relatório de qualidade da consistência lógica de um CDGV basicamente devem constar as seguintes informações:

- Identificação do produto (escopo);
- Elemento de Qualidade;
- Nome, ID e descrição da Medida de Qualidade;
- Data/hora da avaliação;
- Resultado da Medida; e
- Identificação das inconsistências observadas (quando houver).

A Figura 51 apresenta um modelo de relatório de qualidade para a medida VCL100 (Conformidade de estrutura física).

Figura 51 - Modelo de relatório de qualidade em PDF

RELATÓRIO TÉCNICO DE QUALIDADE Nº 001/2021

1. Introdução

Relatório de Controle de Qualidade para atender ao previsto no Contrato nº 001/2021, entre a Prefeitura de X e esta empresa para a validação dos produtos geoespaciais elaborados no Projeto Y de atualização da base contínua cadastral.

2. Referências

- ISO 19.157 (2013)
- ET-CQDG (2016)
- ET-EDGV (2017)

3. Identificação do produto

- a) Título: CDGV de base contínua cadastral, bloco A1;
- b) Data criação: 10/04/2021;
- c) Resumo: CDGV de base contínua cadastral baseada na EDGV 3.0 para o Projeto Y;
- d) Responsável: Empresa Z, website: www.empresaz.com.br;
- e) Escala do Produto: 1:1.000;
- f) Visualização do Produto:



4. Procedimento de avaliação de qualidade

<i>Escopo</i>	<i>Medida ID</i>	<i>Nome da Medida</i>	<i>Conformidade</i>
Todas as classes	VCL100	Conformidade de estrutura física	M = True (conforme)

5. Data/hora da avaliação

2021-06-10 08:31:33

6. Resultados da Medida de Qualidade

<i>Medida ID</i>	<i>Nome da Medida</i>	<i>Medida</i>	<i>Resultado</i>
VCL100	Conformidade de estrutura física	True (conforme)	Aprovado

7. Inconsistências

Não há.

8. Observações

Não há.

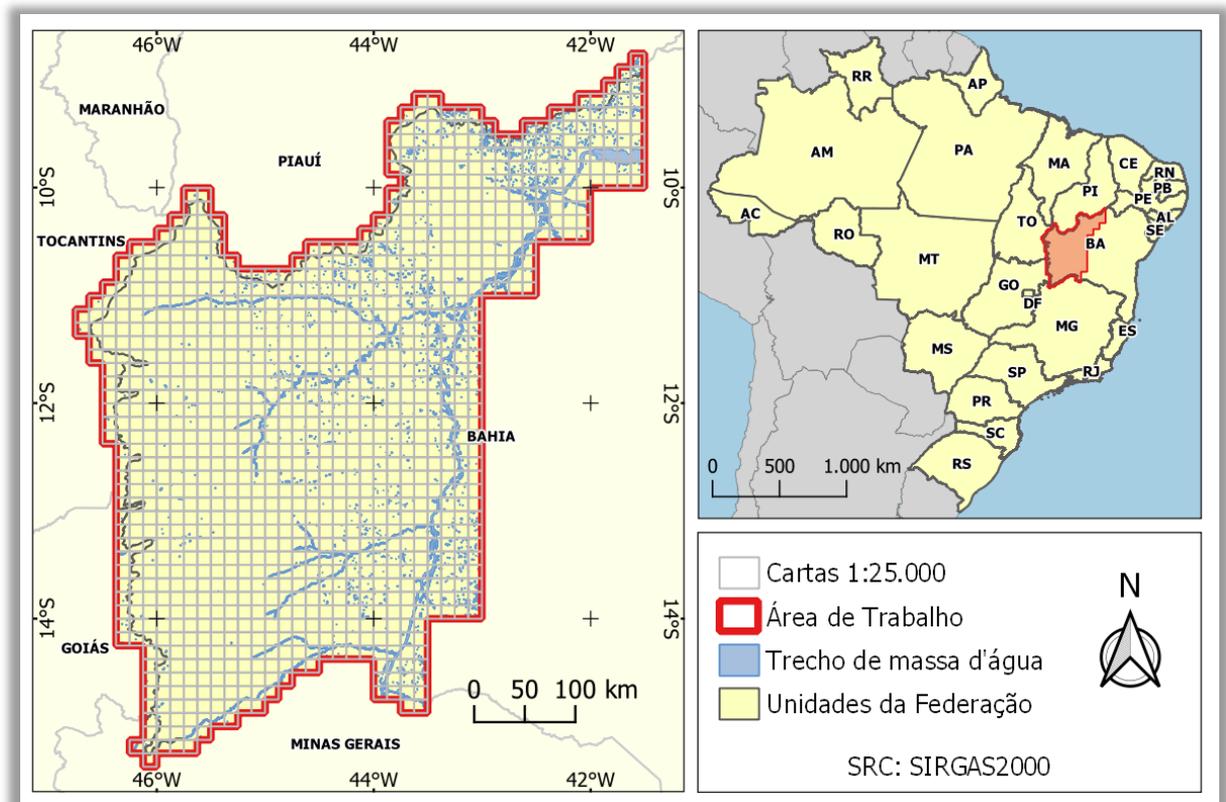
6 METODOLOGIA DA PESQUISA

Este capítulo é dedicado a descrever a metodologia de avaliação da consistência lógica de um Conjunto de Dados Geoespaciais Vetoriais (CDGV), aplicando-se a algumas possibilidades de verificações de inconsistências que ferem regras lógicas aplicadas ao mapeamento topográfico.

6.1 ÁREA DE ESTUDO

A área de estudo será o oeste do estado da Bahia, cujo mapeamento topográfico foi executado pelo Exército Brasileiro entre 2010 e 2019 no Projeto de Mapeamento do Estado da Bahia. A Figura 52 identifica cartograficamente a área de mapeada.

Figura 52 - Área de estudo



Fonte: O autor (2021).

Esta região tem o total de 209.923,5 Km² e corresponde a um total de 1.116 cartas topográficas na escala 1:25.000, seguindo o mapeamento sistemático

brasileiro (França e Ferreira da Silva, 2018; Passos e França, 2018; França et al., 2019).

6.2 DADOS UTILIZADOS

O material avaliado consiste no CDGV da área de estudo, no formato “.sql” correspondente ao backup de banco PostGIS, o qual deve seguir a modelagem das Especificações Técnicas para Estruturação de Dados Geoespaciais Vetoriais (ET-EDGV), versão 2.1.3.

Esses dados são públicos e disponibilizados tanto através do Banco de Dados Geográficos do Exército (BDGEx), assim como pelo geoportal da Superintendência de Estudos Econômicos e Sociais do estado da Bahia (SEI, 2021).

Os dados foram trabalhados utilizando o SGBD PostgreSQL, v. 10.11, com sua extensão espacial PostGIS, v.2.5.1, sendo os mapas elaborados com o software QGIS 3.16 e as avaliações, análises e relatórios feitos baseados em consultas em SQL implementadas para este trabalho, estando todo o código disponível nos Apêndices de D a F.

Do CDGV do oeste baiano foram selecionadas para a avaliação as seguintes classes de feições para avaliação:

- Ponto de drenagem (*Ponto_Drenagem*): ponto de conectividade entre dois trechos de drenagem ou mais;
- Trecho de drenagem (*Trecho_Drenagem*): corresponde a um corpo d'água, cuja geometria do tipo linha representa o fluxo d'água, permanente ou temporário, contido ou coincidente com um trecho de massa d'água capturado como linha, em função da escala de aquisição;
- Massa d'água (*Massa_Dagua*): corpo d'água representado por polígono, tais como oceano, baías, enseadas, meandros abandonados, lagos, lagoas, e os açudes que não possuam fluxo d'água; e
- Trecho de massa d'água (*Trecho_Massa_Dagua*): segmentos de cursos d'água representados por polígonos, que possuem fluxo d'água.

6.3 PROCEDIMENTO METODOLÓGICO

O método de avaliação direta é especializado em: inspeção completa, em que todos os itens do dado são checados; e em inspeção por amostra em que é selecionada uma amostra do universo para serem efetuados a avaliação da qualidade (Araújo, 2015).

O grau de interferência humana, nos processos de avaliação da qualidade, pode ser classificado como não-automatizado, semiautomatizado e automatizado (Mobasher, 2013).

Como todos os procedimentos de avaliação da qualidade da consistência lógica podem ser controlados por algoritmos que verificam computacionalmente regras lógicas, exigindo do analista a definição formal dos parâmetros para cada medida de qualidade que será verificada pela máquina de forma automatizada, pode-se considerar como método de avaliação a inspeção completa, de forma a identificar e reportar todas as inconsistências presentes nos dados.

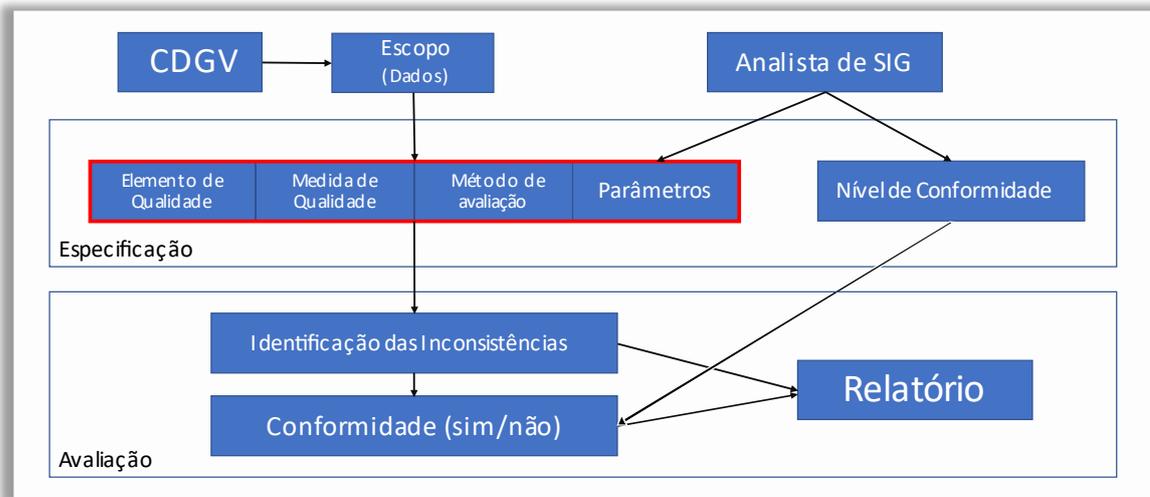
As medidas básicas de qualidade adequadas para a avaliação da consistência lógica são: o “indicador de acerto”, para consistência de formato, conceitual e de domínio; e a “quantidade de erros”, para as consistências topológica e semântica.

O nível de conformidade para a aprovação de um conjunto de dados deve ter como resultado o “indicador de acerto” igual a *True* (ou conforme), já no caso da medida básica de “quantidade de erros” o resultado deve ser zero (nenhuma inconsistência).

A Figura 53 apresenta, de forma geral, o fluxo para a avaliação da consistência lógica de um CDGV. Dentre as classes desse CDGV, são selecionadas aquelas que serão avaliadas definindo o escopo do processo.

O Analista de SIG especifica as medidas de qualidade e define os parâmetros para os procedimentos de avaliação para cada elemento de qualidade, seguindo o modelo apresentado na seção 4.1 (Quadro 14) e estabelecendo também o respectivo nível de conformidade.

Figura 53 - Fluxo de avaliação da consistência lógica para os dados de um CDGV



Fonte: O autor (2021).

O Quadro 16 apresenta os escopos e as medidas para a validação da consistência lógica, detalhando-se a consistência topológica e semântica.

Quadro 16 - Escopos e medidas de qualidade

Escopo	Medida ID	Nome da Medida
Todas as classes	VCL100	Conformidade de estrutura física
Todas as classes	VCL200	Conformidade com o modelo de dados
Todas as classes	VCL300	Valores de domínios conformes
Todas as classes	VCL411	Validação da Geometria: Geometria vazia
Classes do tipo linha e polígono	VCL412	Validação da Geometria: Geometria não-simples
Todas as classes	VCL413	Validação da Geometria: Geometria inválida
Classes do tipo linha e polígono	VCL414	Validação da Geometria: Geometria limpa
Todas as classes	VCL415	Validação da Geometria: Geometria multiparte
Classes do tipo linha e polígono	VCL416	Validação da Geometria: Geometria com ângulos pequenos
Todas as classes	VCL417	Validação da Geometria: Geometria fora das medidas
Todas as classes	VCL421	Validação intraclasses: Geometrias duplicadas (repetidas)
Classes do tipo linha e polígono	VCL422	Validação intraclasses: Sobreposição entre geometrias
Trecho de drenagem	VCL423	Validação intraclasses: Cruzamento entre linhas
Trecho de massa	VCL424	Validação intraclasses: Lacunas entre polígonos

d'água		
Classes do tipo linha e polígono	VCL425	Validação intraclasses: Feição retalhada
Trecho de drenagem	VCL426	Validação intraclasses: Ponta de linha solta
Trecho de drenagem	VCL427	Validação intraclasses: Ausência de ponto de conectividade
Trecho de drenagem	VCL428	Validação intraclasses: Estrutura de rede
Todas as classes	VCL430	Validação interclasses
Todas as classes	VCL510	Validação de regras entre atributos
Todas as classes	VCL520	Validação de regras entre atributo e topologia

Fonte: O autor (2021).

Neste trabalho, as três últimas medidas VCL430, VCL510 e VCL520 foram avaliadas, considerando-se que os dados já haviam passado por uma validação e correção na plataforma Gothic (Passos et al., 2017) e, em princípio, não possuem inconsistências de geometria e intraclasses.

Além disso, buscou-se destacar a utilização de consultas SQL e os operadores topológicos, os quais são baseados nas matrizes DE-9IM, para a validação do CDGV nessas medidas.

A medida VCL430 é detalhada no Apêndice A e na seção 3.4.1.3 deste trabalho, consistindo na verificação por consultas SQL (Apêndice E) das regras descritas no Quadro 17.

Quadro 17 - Regras topológicas interclasses verificadas

Regra	Classe 1	Deve	Topologia	Classe 2	Cardinalidade
01	hid_trecho_drenagem_l	sim	touches	hid_massa_dagua_a	0..1
02	hid_trecho_drenagem_l	sim	Within	hid_trecho_massa_dagua_a	0..1
03	hid_trecho_drenagem_l	sim	touches	hid_ponto_drenagem_p	2..2
04	hid_trecho_massa_dagua_a	sim	contains	hid_trecho_drenagem_l	1..*
05	hid_ponto_drenagem_p	sim	touches	hid_trecho_drenagem_l	1..*

06	hid_ponto_drenagem_p	não	Within	hid_massa_dagua_a	1..*
07	hid_trecho_drenagem_l	não	Crosses	hid_massa_dagua_a	1..*
08	hid_massa_dagua_a	não	intersects	hid_trecho_massa_dagua_a	1..*

Fonte: O Autor (2021).

A medida VCL510, que tem como objetivo avaliar a consistência semântica através de regras entre atributos, está detalhada no Apêndice B e seção 3.5.1 deste trabalho, consistindo na verificação por consultas SQL (Apêndice F) das regras descritas no Quadro 18.

Quadro 18 - Regras entre atributos verificadas

Regra	Classe	Atributo 1	Atributo 2	Condição	Se e somente se
01	hid_massa_dagua_a	tipoMassa Dagua	salinidade	Se atr1 ∈ {Oceano, Baía, Enseada}, então atr2 ∈ {Salgada} ou Se atr1 ∈ {Lago, Lagoa, Represa/Açude}, então atr2 ∈ {Doce}	Sim
02	hid_trecho_massa_dagua_a	tipoTrecho Massa	salinidade	Se atr1 ∈ {Laguna} então atr2 ∈ {Salgada} ou Se atr1 ∈ {Rio, Canal, Represa/Açude}, então atr2 ∈ {Doce}	Sim
03	hid_trecho_drenagem_l	regime	navegabilidade	Se atr1 ∈ {Seco}, então atr2 ∈ {Não navegável}	Não
04	hid_trecho_drenagem_l	dentroDe Poligono	coincideCom DentroDe	Se atr1 ∈ {Sim} então atr2 ∈ {Rio, Canal, Laguna, Represa/Açude, Corredeira, Eclusa, Terreno de inundação, Foz Marítima, Barragem}	Não

Fonte: O autor (2021).

A medida VCL520, que tem como objetivo avaliar a consistência semântica através de regras entre atributo e topologia, está detalhada no Apêndice C e seção 3.5.2 deste trabalho, consistindo na verificação por consultas SQL (Apêndice F) das regras descritas no Quadro 19.

Quadro 19 - Regras entre atributo e topologia verificadas

Regra	Classe 1	Atributo	Classe 2	Condição	Se e somente se
01	hid_ponto_drenagem_p	relacionado	hid_massa_dagua_a	Se atr \in {Lago, Lagoa, Meandro abandonado} então geometria1 toca geometria2	Sim
02	hid_trecho_drenagem_l	Coincide ComDentroDe	hid_trecho_massa_dagua_a	Se atr \in { Rio, Canal, Represa/açude, Laguna} então geometria1 deve estar dentro de geometria2	Sim

Fonte: O autor (2021).

7 RESULTADOS E DISCUSSÕES

O número de feições por classe avaliada do CDGV da área de estudo, na modelagem ET-EDGV v.2.1.3, é dado pela Tabela 1.

Tabela 1 - Número de feições por classe avaliada

Classe	Nome da classe no BD	Número de Feições
Ponto de drenagem	Ponto_Drenagem	17.051
Trecho de drenagem	Trecho_Drenagem	248.809
Massa d'água	Massa_Dagua	5.735
Trecho de massa d'água	Trecho_Massa_Dagua	3.654

Fonte: O autor (2021).

7.1 INCONSISTÊNCIAS TOPOLÓGICAS INTERCLASSES

A quantidade e percentual de inconsistências encontradas para as regras topológicas interclasses, definidas no Quadro 17, são dados na Tabela 2.

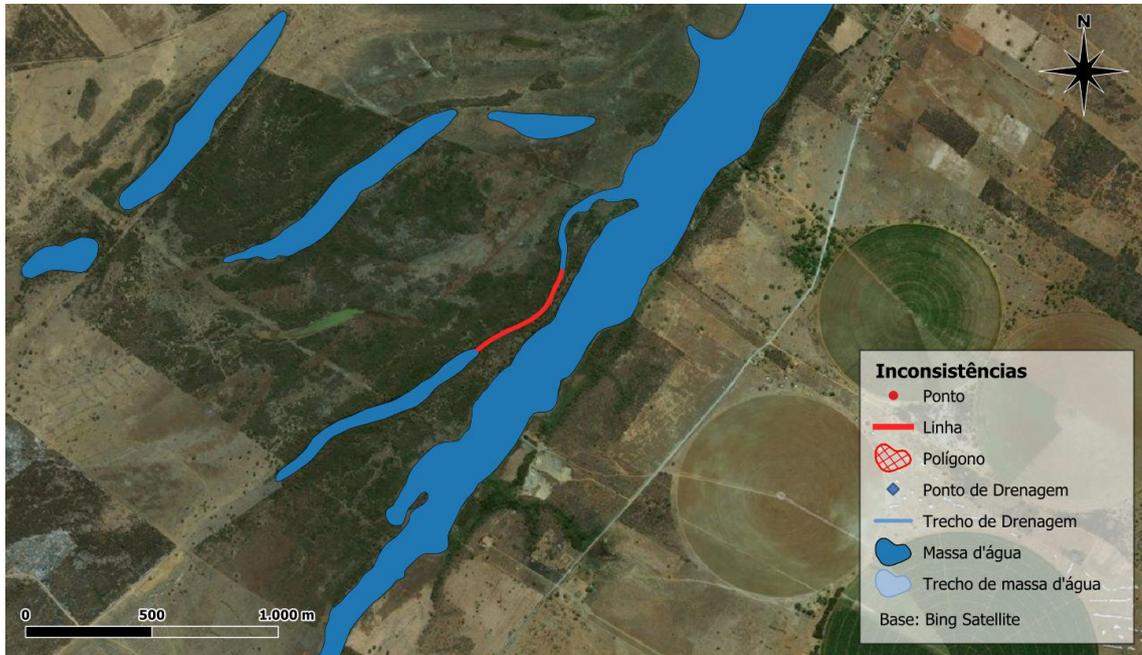
Tabela 2 - Total de inconsistências para as regras interclasses.

Regra	Descrição	# Inconsistências e Percentual
01	Trecho_Drenagem deve tocar 0 ou 1 feição da classe Massa_Dagua.	1 (0,00%)
02	Trecho_Drenagem deve estar dentro de 0 ou 1 feição da classe Trecho_Massa_Dagua.	0 (0,00%)
03	Trecho_Drenagem deve tocar exatamente 2 feições da classe Ponto_Drenagem.	244.705 (98,35%)
04	Trecho_Massa_Dagua deve conter pelo menos 1 feição da classe Trecho_Drenagem.	649 (17,76%)
05	Ponto_Drenagem deve tocar pelo menos 1 feição da classe Trecho_Drenagem.	3.452 (20,25%)
06	Ponto_Drenagem NÃO deve estar dentro de mais de 0 feição da classe Massa_Dagua.	0 (0,00%)
07	Trecho_Drenagem NÃO deve cruzar de mais de 0 feição da classe Massa_Dagua.	18 (0,01%)
08	Massa_Dagua NÃO deve interceptar mais de 0 feição da classe Trecho_Massa_Dagua.	5 (0,09%)

Fonte: O autor (2021).

Para a inconsistência da regra 01 foi encontrada uma única ocorrência (Figura 54), onde um trecho de drenagem toca em duas massas d'água, o que caracterizaria um fluxo de água entre elas, infringindo à regra do modelo conceitual.

Figura 54 - Inconsistência topológica - regra 01

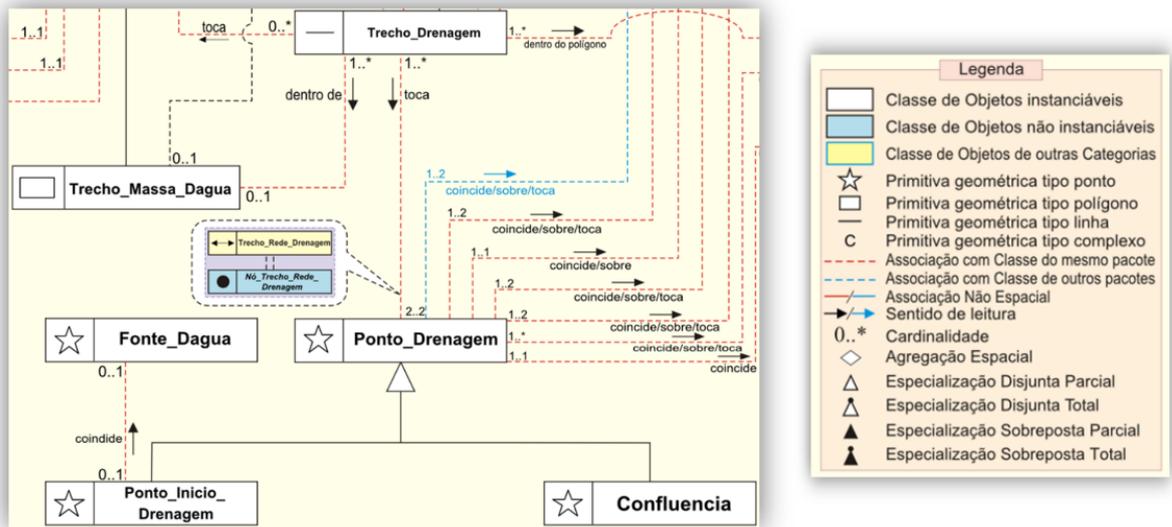


Fonte: O autor (2021).

Na validação da regra 02, nenhum caso de inconsistência foi encontrado, tendo em vista que somente linhas estivessem dentro de 2 ou mais polígonos dos trechos de massa d'água seriam identificadas.

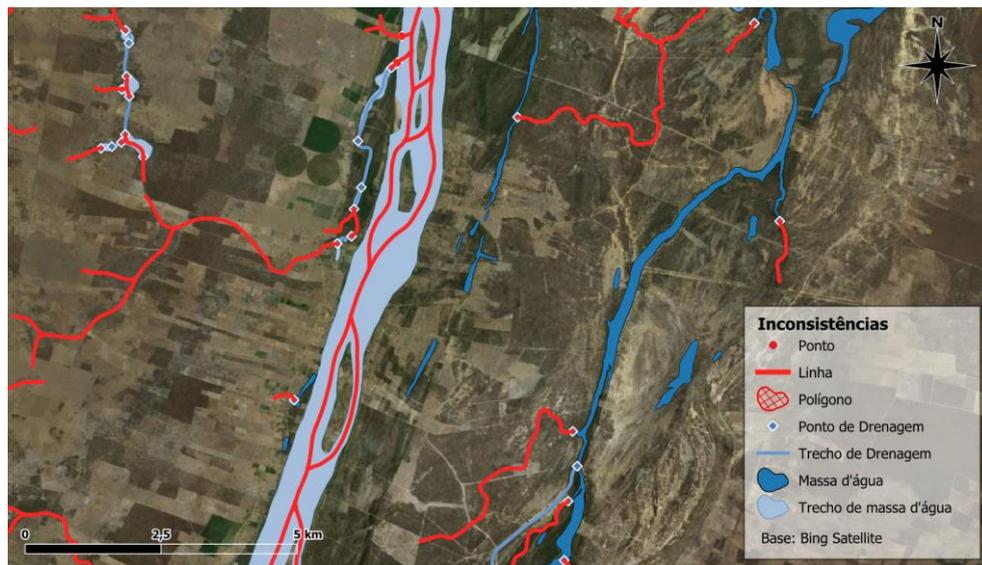
Para as inconsistências da regra 03, foi verificado que os pontos de “confluência” e “início de drenagem” não constam na classe “ponto de drenagem”, conforme previsto na ET-EDGV 2.1.3 (Figura 55). Esse problema no banco de dados fez com que mais de 98% dos trechos de drenagem estejam inconsistentes nesta regra (Figura 56).

Figura 55 - Relacionamento entre “Trecho de Drenagem” e “Ponto Drenagem” para a ET-EDGV 2.1.3



Fonte: CONCAR (2010).

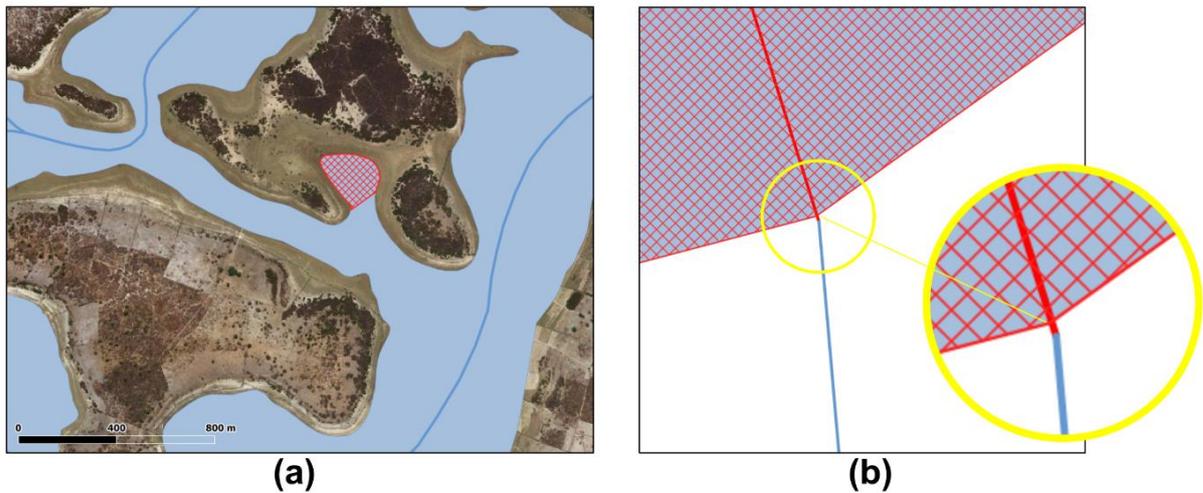
Figura 56 - Inconsistências topológicas - regra 03



Fonte: O autor (2021).

Na validação da regra 04, verificou-se raros casos de polígonos da classe “Trecho de Massa D’água” que realmente não sejam atravessados por um trecho de drenagem, como mostrado na Figura 57(a). No entanto, a maior causa das inconsistências está relacionada à falta de precisão das coordenadas das feições, fazendo com que a geometria linha não esteja completamente dentro do polígono, como mostrado na Figura 57(b).

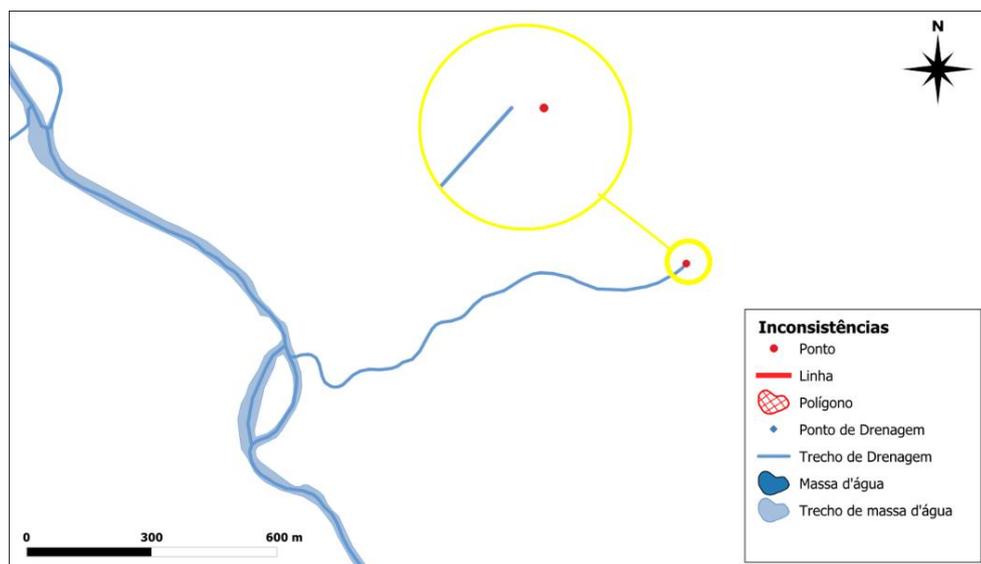
Figura 57 - Inconsistências topológicas – regra 04 (a) polígono que não contém linha; (b) polígono que não contém totalmente a linha por precisão das coordenadas



Fonte: O autor (2021).

Na validação da regra 05, não foi identificado ponto de drenagem “isolado”, ou seja, sem estar relacionado com uma linha da classe Trechos de Drenagem. Entretanto, o motivo das inconsistências encontradas pela consulta SQL está relacionado à precisão das coordenadas de alguns pontos, conforme verificado na Figura 58.

Figura 58 - Inconsistências topológicas - regra 05

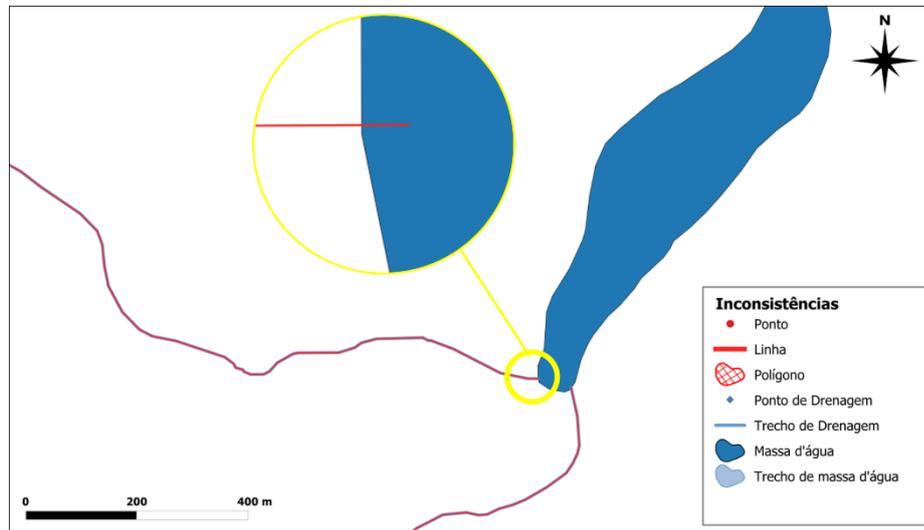


Fonte: O autor (2021).

A regra 06, verificou se algum Ponto de Drenagem está dentro de Massa d'água. Para esta regra não foi encontrado nenhuma inconsistência.

Já para a regra 07, que busca identificar linhas de drenagem que cruzam Massa d'água, todas as 18 inconsistências encontradas estão relacionadas à precisão das coordenadas, como mostrado na Figura 59.

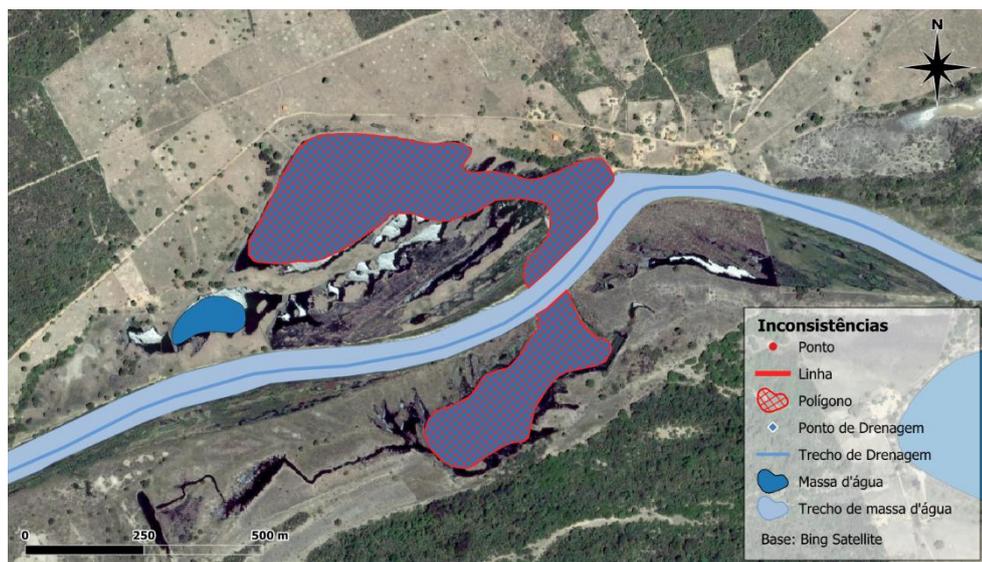
Figura 59 - Inconsistências topológicas - regra 07



Fonte: O autor (2021).

Na avaliação da última regra topológica interclasse, foram identificados cinco casos de interseção entre polígonos das classes Massa D'água e Trecho de Massa D'água, como mostrado na Figura 60.

Figura 60 - Inconsistências topológicas - regra 08



Fonte: O autor (2021).

7.2 INCONSISTÊNCIAS SEMÂNTICAS ENTRE ATRIBUTOS

A quantidade e percentual de inconsistências encontradas para as regras entre atributos, definidas no Quadro 18, são dados na Tabela 3.

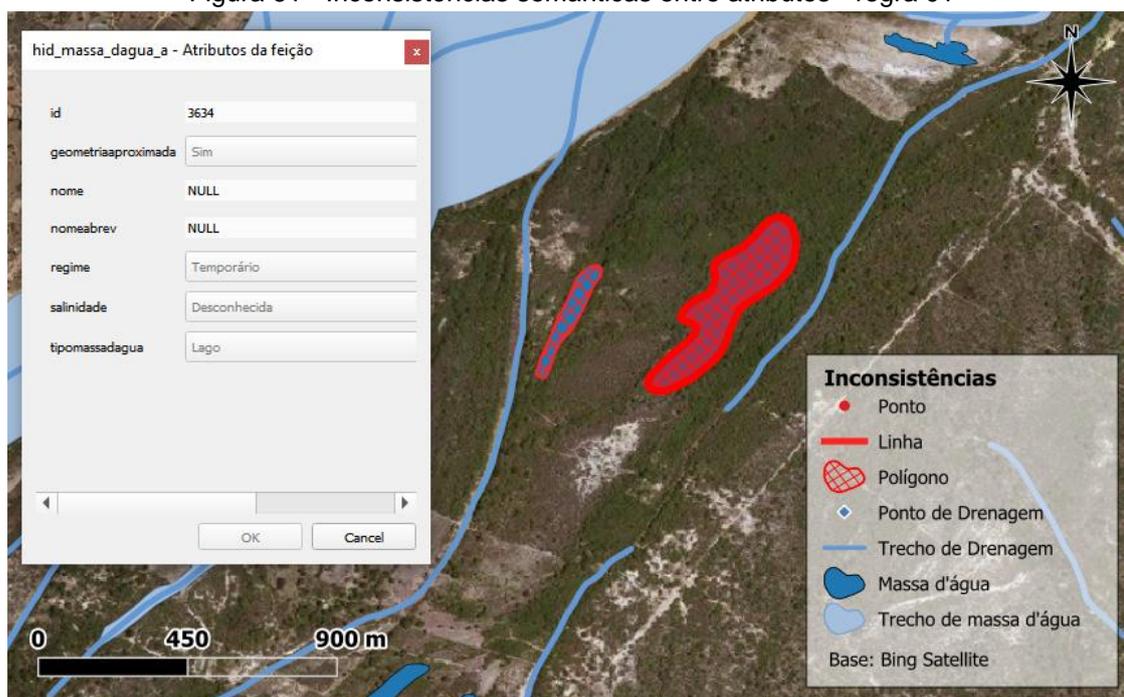
Tabela 3 - Total de inconsistências para as regras entre atributos

Regra	Descrição	Classe	# Inconsistências e Percentual
01	Se e somente se “tipoMassaDagua” \in {Oceano, Baía, Enseada}, então “salinidade” \in {Salgada} ou Se e somente se “tipoMassaDagua” \in {Lago, Lagoa, Represa/Açude}, então “salinidade” \in {Doce}.	Massa_Dagua	2 (0,03%)
02	Se e somente se “tipoTrechoMassa” \in {Laguna} então “salinidade” \in {Salgada} ou Se e somente se “tipoTrechoMassa” \in {Rio, Canal, Represa/Açude}, então “salinidade” \in {Doce}.	Trecho_Massa_Dagua	18 (0,49%)
03	Se “regime” \in {Seco}, então “navegabilidade” \in {Não navegável}.	Trecho_Drenagem	71 (0,03%)
04	Se “dentroDePoligono” \in {Sim} então “coincideComDentroDe” \in {Rio, Canal, Laguna, Represa/Açude, Corredeira, Eclusa, Terreno de inundação, Foz Marítima, Barragem}.	Trecho_Drenagem	2.940 (1,18%)

Fonte: O autor (2021).

Para a regra 01, foram encontradas apenas duas inconsistências (Figura 61) onde as feições do “tipoMassaDagua” igual a “Lago” tiveram o atributo “salinidade” definido com o valor “Desconhecido”, o que não é permitido.

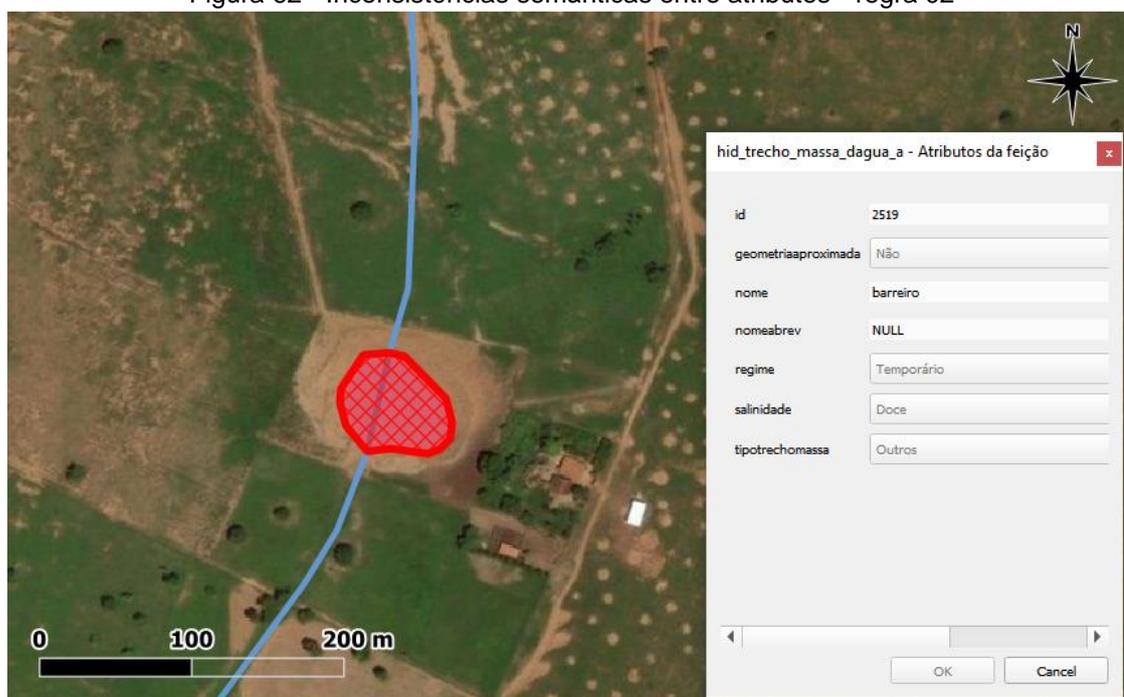
Figura 61 - Inconsistências semânticas entre atributos - regra 01



Fonte: O autor (2021).

Na validação da regra 02, foi verificado para todas as suas inconsistências que os trechos de massa d'água tem o atributo "tipoTrechoMassa" igual a "Outros", no entanto com atributo "salinidade" igual a "Doce" (Figura 62), o que também fere a regra.

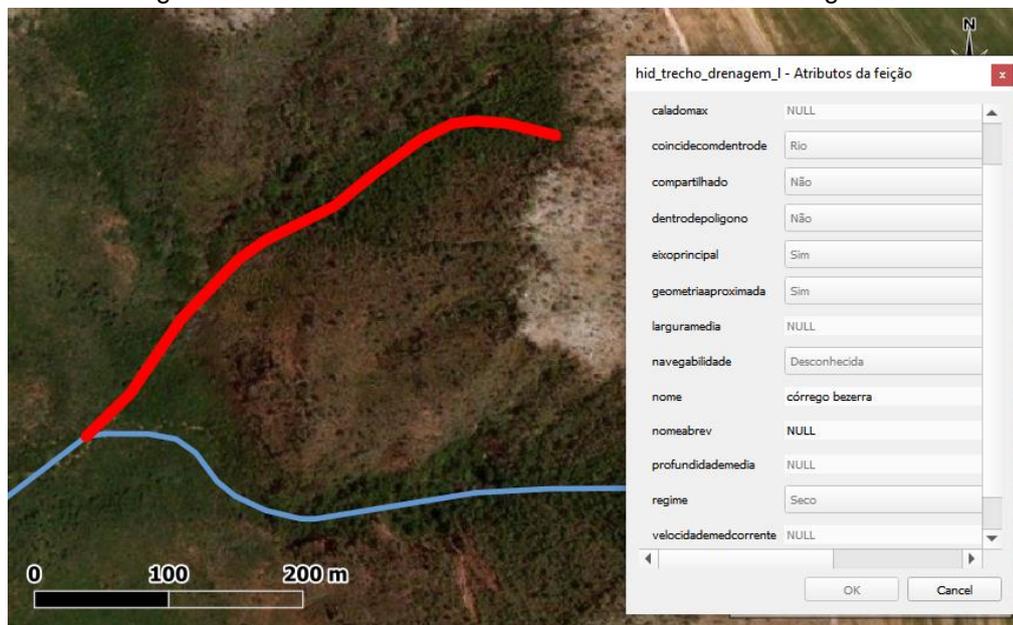
Figura 62 - Inconsistências semânticas entre atributos - regra 02



Fonte: O autor (2021).

Todas as inconsistências da regra 03 contam que o atributo “navegabilidade” do Trecho de Drenagem é “Desconhecido” quando o “regime” é igual a “Seco” (Figura 63), configurando-se em inconsistência lógica.

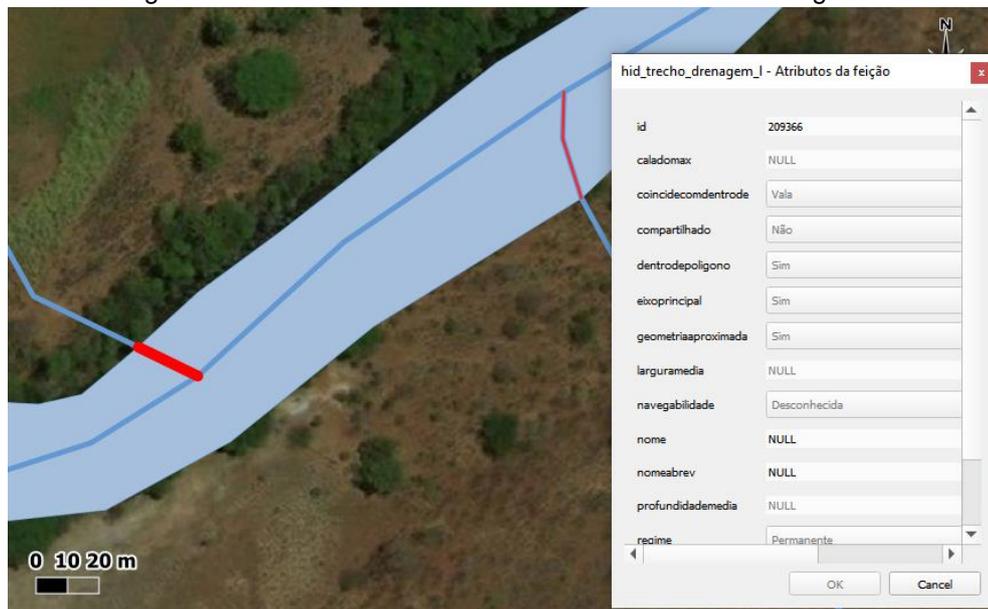
Figura 63 - Inconsistências semânticas entre atributos - regra 03



Fonte: O autor (2021).

As inconsistências da regra 04 contam que o atributo “dentroDePoligono” do Trecho de Drenagem é “Sim”, no entanto o atributo “coincideComDentroDe” é igual a “Vala” ou “Não aplicável”, como o exemplo da Figura 64.

Figura 64 - Inconsistências semânticas entre atributos - regra 04



Fonte: O autor (2021).

7.3 INCONSISTÊNCIAS SEMÂNTICAS ENTRE ATRIBUTO E TOPOLOGIA

A quantidade e percentual de inconsistências encontradas para as regras entre atributo e topologia, definidas no Quadro 19, são dados na Tabela 4.

Tabela 4 - Total de inconsistências para as regras entre atributo e topologia

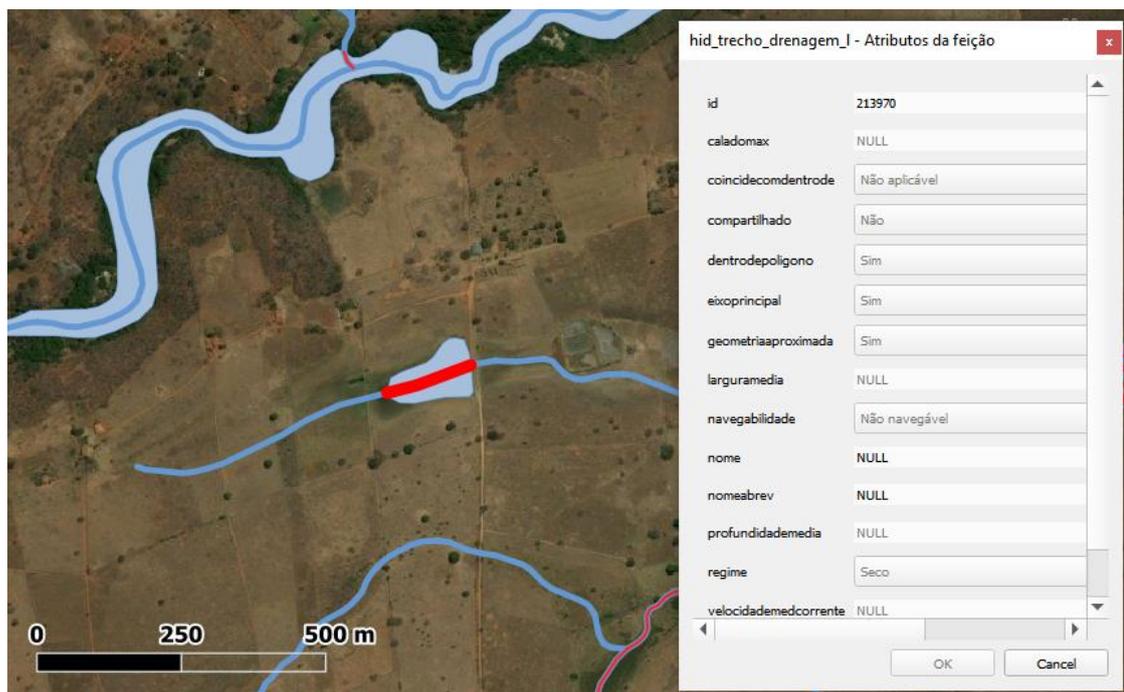
Regra	Descrição	Classe	# Inconsistências e Percentual
01	Se e somente se “relacionado” \in {Lago, Lagoa, Meandro abandonado} então geometria toca a geometria da classe “Massa_Dagua”.	Ponto_Drenagem	0 (0,00%)
02	Se e somente se “CoincideComDentroDe” \in { Rio, Canal, Represa/açude, Laguna} então geometria deve estar dentro de geometria da classe “Trecho_Massa_Dagua”.	Trecho_Drenagem	111.167 (44,68%)

Fonte: O autor (2021).

Da Tabela 4, observa-se que não existe inconsistências para a regra 01. Esse resultado pode ser explicado pelo fato de os Pontos de Drenagem serem gerados automaticamente a partir das classes Trecho e Drenagem, Massa d’água e Trecho de massa d’água.

O mesmo não acontece com a regra 02, onde o atributo “CoincideComDentroDe” é preenchido manualmente pelo operador. Neste caso uma grande quantidade das inconsistências está relacionada ao preenchimento do valor “Não aplicável” ou “Desconhecido” para as linhas que estão dentro de trecho de massa d’água, como o exemplo da Figura 65.

Figura 65 - Inconsistências semânticas entre atributo e topologia - regra 02



Fonte: O autor (2021).

8 CONCLUSÕES E RECOMENDAÇÕES

Alguns produtores de dados geoespaciais apresentam grande dificuldade de garantir a integridade e confiabilidade dos seus produtos devido à verificação inadequada das relações topológicas e semânticas, ocasionando retrabalhos com verificações repetidas e pouco eficientes.

O objetivo deste trabalho, portanto, foi de fornecer uma base teórica consistente para o entendimento da qualidade dos dados geográficos no que se refere a consistência lógica e propor novos conceitos para sistematizar os procedimentos de validação da consistência lógica de um CDGV.

As diversas regras de integridade especificadas e explanadas neste trabalho garantirão maior consistência dos dados, além de direcionar o controle de qualidade no mapeamento topográfico, permitindo diversas padronizações nos processos de validação e elaboração de relatório de qualidade de um CDGV, conforme apresentado na Figura 53, que mostra o fluxo de avaliação da qualidade para um escopo de um CDGV.

O Quadro 16, também desenvolvido neste trabalho, estabelece a sequência ideal para a avaliação das medidas de qualidade referentes a categoria consistência lógica, algo ainda ausente nas normativas nacionais. Vale ressaltar que esses procedimentos de avaliação podem ser adaptados às especificações dos produtos e necessidades do contratante.

Sobre a consistência semântica, um elemento qualidade proposto neste trabalho, constatou-se a importância desse elemento na realização de processos automáticos de validação de regras lógicas, sejam elas entre atributos ou entre atributo e topologia, tendo em vista que o controle da acurácia temática de atributos não garante a completa identificação e correção das inconsistências, por utilizar métodos de amostragem e um percentual estatístico de aceitação.

Sobre o exemplo de aplicação, este trabalho realizou a validação topológica interclasses e semântica em um CDGV previamente validado no Gothic, um sistema utilizado há mais de 10 anos pela DSG.

Dos resultados da consistência topológica, verificou-se uma grande quantidade de inconsistências relacionadas à precisão das coordenadas, e à implementação do modelo no banco de dados, como é o caso da classe Pontos de

Drenagem, além de uma menor quantidade de outros casos que, de fato, infringem as regras do modelo de dados.

Já na validação da consistência semântica entre atributos, as inconsistências constituem um pequeno percentual de cada regra. Diferente da validação semântica entre atributo e topologia, onde se identificou um elevado percentual de inconsistências por se tratar de um atributo preenchido manualmente na etapa de reambulação.

Os resultados obtidos nesse trabalho indicam a necessidade de modernização dos procedimentos adotados na produção cartográfica pelos órgãos e instituições responsáveis pelo mapeamento sistemático brasileiro, de forma a documentar e implementar todas as possíveis regras lógicas do modelo de dados para se alcançar uma completa integridade do CDGV produzido.

No que se refere às contribuições deste trabalho, além da compilação de uma base teórica abrangente e sistematização do fluxo dos procedimentos de avaliação da consistência lógica de um CDGV, destaca-se também o desenvolvimento de soluções para a verificação das inconsistências topológicas e semânticas com consultas SQL baseadas no software livre PostgreSQL e sua extensão PostGIS, sendo essas consultas compartilhadas através dos Apêndices D, E e F.

Uma proposta de trabalho futuro é o aprimoramento das consultas SQL desenvolvidas nesta pesquisa, de forma a viabilizar a customização dos parâmetros de validação. Assim, baseado na documentação das restrições semânticas, será possível automatizar as consultas a partir de um arquivo estruturado de configurações como, por exemplo, no formato *JavaScript Object Notation* (JSON), abrangendo novas classes com diferentes regras topológicas e semânticas.

Ainda sobre futuras pesquisas alinhadas com o tema deste trabalho, sugere-se também o estudo de técnicas que permitam a validação em tempo real das regras lógicas elencadas para o modelo de dados no exato momento da aquisição ou vetorização, seja na forma de construção de mecanismos de “alerta”, ou na construção de *constraints* que impedirão que o dado inconsistente seja armazenado no banco de dados.

REFERÊNCIAS

- ANDRADE, L. C. O. Integridade topológica em sistemas de bancos de dados espaciais. Dissertação de Mestrado. UNB. Brasília. 2018.
- ARAÚJO, Felipe Santos; DE SOUZA BIAS, Edilson; DE HOLANDA, Maristela Tertó. Proposta de um modelo conceitual de banco de dados geográficos para o cadastro territorial multifinalitário do Distrito Federal. *Revista Brasileira de Cartografia*, v. 67, n. 3, 2015.
- ARAÚJO, Felipe Santos. Modelagem de processos automatizados para controle de consistência lógica em banco de dados geográficos: uma aplicação para o Cadastro Territorial Multifinalitário do Distrito Federal. Dissertação de Mestrado. UnB. Brasília. 2015.
- ARIZA-LÓPEZ, Francisco Javier et al. Quality Control of “As Built” BIM Datasets Using the ISO 19157 Framework and a Multiple Hypothesis Testing Method Based on Proportions. *ISPRS International Journal of Geo-Information*, v. 8, n. 12, p. 569, 2019.
- BORGES, Karla AV; DAVIS, Clodoveu A.; LAENDER, Alberto HF. OMT-G: an object-oriented data model for geographic applications. *Geoinformatica*, v. 5, n. 3, p. 221-260, 2001.
- BORGES, Karla AV; DAVIS JR, Clodoveu A.; LAENDER, Alberto HF. Integrity constraints in spatial databases. In: *Database integrity: challenges and solutions*. IGI Global, 2002. p. 144-171.
- BORGES, Karla AV; DAVIS JR, Clodoveu A.; LAENDER, Alberto HF. Modelagem conceitual de dados geográficos. CASANOVA, et al. *Banco de Dados Geográfico. MundoGEO: Curitiba*, p. 83-136, 2005.
- BORRMANN, André; RANK, Ernst. Topological analysis of 3D building models using a spatial query language. ***Advanced Engineering Informatics***, v. 23, n. 4, p. 370-385, 2009.
- BRODEUR, Jean et al. Geographic Information Metadata—An Outlook from the International Standardization Perspective. *ISPRS International Journal of Geo-Information*, v. 8, n. 6, p. 280, 2019.
- CÂMARA, Gilberto. Representações computacionais do espaço geográfico. CASANOVA, Marco. CÂMARA, Gilberto. DAVIS, Clodoveu. VINHAS, Lúbia. QUEIROZ, Gilberto Ribeiro de. *Banco de Dados Geográficos. Curitiba*, 2005.
- CAO, Guofeng. Spatial analysis and modeling: Database Fundamentals. 06 aug. 2014, 12 dec. 2014. 99 p. Notas de Aula. Department of Geosciences. Texas Tech University.
- CASANOVA, M. A., Câmara, G., Davis Jr, C., Vinhas, L., & Queiroz, G. R. *Banco de dados geográficos. Editora Mundogeo. Curitiba-PR*. 2005.

CLEMENTINI, Eliseo; DI FELICE, Paolino. A comparison of methods for representing topological relationships. *Information sciences-applications*, v. 3, n. 3, p. 149-178, 1995.

COMISSÃO NACIONAL DE CARTOGRAFIA - CONCAR. Especificação Técnica para Estruturação de Dados Geoespaciais Vetoriais (ET-EDGV). Versão 3.0. Brasília, 2017.

COMISSÃO NACIONAL DE CARTOGRAFIA - CONCAR. Especificação Técnica para Estruturação de Dados Geoespaciais Vetoriais (ET-EDGV). Versão 2.1.3. Brasília, 2010.

CONCAR. Perfil de Metadados Geoespaciais do Brasil. 2009. Disponível em: http://www.concar.gov.br/arquivo/Perfil_MGB_Final_v1_homologado.pdf

CRESPO, R. G., Romero, W. L., Martínez, O. S., & Montenegro-Marín, C. E. Design and Modeling to Generalized Linear Elements in a Vector Formatted Cartographic. **International Journal of Advancements in Computing Technology**, v. 6, n. 3, p. 96, 2014.

DAVIS JR, Clodoveu A.; LAENDER, Alberto HF. Extensões ao modelo OMT-G para produção de esquemas dinâmicos e de apresentação. In: Anais do II Workshop Brasileiro de Geoinformática, São Paulo. 2000.

DAVIS JR, C. A.; BORGES, Karla Albuquerque de Vasconcelos; SOUZA, Ligiane Alves de ; CASANOVA, Marco Antônio ; LIMA JR, Paulo de Oliveira . O Open Geospatial Consortium. In: Casanova, M. A.; Câmara, G.; Davis Jr., C. A.; Vinhas, L.; Queiroz, G. R. (Org.). Bancos de Dados Geográficos. Curitiba (PR): Editora MundoGeo, 2005, v. único, p. 379-397.

DSG. Norma da Especificação Técnica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG) – 1ª Edição. 2016.

DSG. DSGTools. 2015. Disponível em: <https://plugins.qgis.org/plugins/DsgTools/>.

EGENHOFER, Max J. et al. A critical comparison of the 4-intersection and 9-intersection models for spatial relations: formal analysis. In: **AUTOCARTO-CONFERENCE-**. ASPRS AMERICAN SOCIETY FOR PHOTOGRAMMETRY AND, 1993. p. 1-1.

Elmasri, R. & Navathe, S. (2000) **Fundamentals of database systems**. 3rd Edition. Addison-Wesley, Reading, Massachusetts.

ESRI. ESRI Shapefile Technical Description: An ESRI White Paper—July 1998. Disponível em: <https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf>

ESRI. ArcGIS. Geoprocessing considerations for shapefile output. 2020. Disponível em: <https://desktop.arcgis.com/en/arcmap/latest/manage-data/shapefiles/geoprocessing-considerations-for-shapefile-output.htm>

ESRI. ArcGIS Resource Center ArcSDE SDK 10 C and Java API – Predicates. 2010. https://help.arcgis.com/en/geodatabase/10.0/sdk/arcsde/concepts/geometry/shapes/spatial_relations/predicates.htm

FRANÇA, L. L. S. Topological validation of drainage network with QGIS. Anais 7º Simpósio de Geotecnologias no Pantanal, Jardim, MS. Embrapa Informática Agropecuária/INPE. p. 262-273. 2018.

FRANÇA, L. L. S., Silva, T. A., Andrade, A.C.B.A.B., Alcântara, L.A. Vetorização de Cobertura Terrestre no QGIS. Simpósio Brasileiro de Ciências Geodésicas e Tecnologias Da Geoinformação, VII, Recife-PE, p.393-400. 2018.

FRANÇA, L.L.S., Emprego de Sistema de Informações Geográficas (SIG) no monitoramento e controle de obras, VII Simpósio Brasileiro de Ciências Geodésicas e Tecnologias da Geoinformação, Recife, 2018.

FRANÇA, Leandro Luiz Silva de; SILVA, Luiz Felipe Coutinho Ferreira da. Comparison between the Double Buffer Method and the Equivalent Rectangle Method for the quantification of discrepancies between linear features. **Boletim de Ciências Geodésicas**, v. 24, n. 3, p. 300-317, 2018.

FRANÇA, Leandro Luiz Silva de; PENHA, Alex de Lima Teodoro da; CARVALHO, João Alberto Batista de. Comparison between absolute and relative positional accuracy assessment-a case study applied to Digital Elevation Models. **Boletim de Ciências Geodésicas**, v. 25, n. 1, 2019.

FRANÇA, Leandro Luiz Silva de; Passos, Joel Borges dos ; Portugal, Jose Luiz . Topological validation: a study applied for hydrographic features of a watershed. Ciências Exatas e da Terra: Aprendizado, Integração e Necessidades do País. 1ed.: Atena Editora, 2020, p. 191-207.

FRANÇA, L. L. S. D., Seixas, A. D., Gama, L. F., & Moraes, J. N. D. Optimized determination of 3d coordinates in the survey of inaccessible points of buildings-example of application implemented in free software. **Boletim de Ciências Geodésicas**, v. 27, 2021.

HERRING, John et al. OpenGIS® Implementation Standard for Geographic information-Simple feature access-Part 1: Common architecture [Corrigendum]. 2011.

IBGE – Instituto Brasileiro de Geografia e Estatística. Avaliação da Qualidade de Dados Geoespaciais. Manuais Técnicos em Geociências nº 13. 2017.

IBGE – Instituto Brasileiro de Geografia e Estatística. Procedimentos de Inspeção de Qualidade de Bases Cartográficas Contínuas. Manual de Serviço. 2020.

INSPIRE. D2.8.I.8 Data Specification on Hydrography – Technical Guidelines. 2014.

ISO – International Organization for Standardization. 19.157:2013. Geographic Information – Services. Quality management systems-requirements (ISO 19.157:2013). 2013.

LAGGNER, Birgit; ORTHEN, Natascha. A new GIS toolbox for integrating massive heterogeneous GIS data for land use change analysis. **OSGeo Journal**, v. 13, n. 1, p. 78-89, 2014.

LEDOUX, H.; ARROYO OHORI, K.; MEIJERS, M. Automatically repairing invalid polygons with a constrained triangulation. Proceedings of the AGILE 2012 International Conference, p. 13–18, 2012. 12, 13.

LIZARDO, Luís Eduardo Oliveira; DAVIS, Clodoveu Augusto. OMT-G Designer: a Web tool for modeling geographic databases in OMT-G. In: International Conference on Conceptual Modeling. Springer, Cham, 2014. p. 228-233.

LIZARDO, Luís Eduardo Oliveira; DAVIS JR, Clodoveu Augusto. A PostGIS extension to support advanced spatial data types and integrity constraints. In: Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, 2017. p. 33.

LONGLEY, Paul A. et al. **Sistemas e ciência da informação geográfica**. Bookman Editora, 2013.

MARAŞ, Hakan Hadi et al. Topological error correction of GIS vector data. **International Journal of Physical Sciences**, v. 5, n. 5, p. 476-483, 2010.

MOBASHERI, AMIN. Exploring the Possibility of Semi-automated Quality Evaluation of Spatial Datasets in Spatial Data Infrastructure. **ITB Journal of Information and Communication Technology**, 2013, Vol 7(1), pp. 1-14.

MÕISJA, Kiira; UUEMAA, Evelyn; OJA, Tõnu. The implications of field worker characteristics and landscape heterogeneity for classification correctness and the completeness of topographical mapping. **ISPRS International Journal of Geo-Information**, v. 7, n. 6, p. 205, 2018.

OGC. Open Geospatial Consortium. 2020. Disponível em: <https://www.opengeospatial.org/>

OGC® , 2018. GeoPackage Encoding Standard. Disponível em: <https://www.geopackage.org/spec121/index.html> . Acesso em: 18 de dezembro de 2020.

PASSOS, J. B.; FRANÇA, L. L. Processo de reambulação no mapeamento topográfico. **Revista Brasileira de Geomática**, p. 119-138. 2018.

PASSOS, J. B., Carvalho, R. B., Penha, A. D. L. T., & França, L. L. S. Estruturação e validação de dados geográficos em ambiente orientado a objeto do Sistema Gothic. Anais do XVIII Simpósio Brasileiro de Sensoriamento Remoto -SBSR, Santos-SP. 2017.

PERCIVALL, George et al. Innovation in OGC: The Interoperability Program. *ISPRS International Journal of Geo-Information*, v. 4, n. 4, p. 2362-2378, 2015.

PostgreSQL Global Development Group. PostgreSQL 10.17 Documentation. Disponível em: <https://www.postgresql.org/docs/>. 2021.

RASHIDAN, Muhammad Hanis; MUSLIMAN, Ivin Amri. GeoPackage as future ubiquitous GIS data format: a review. *Jurnal Teknologi*, v. 73, n. 5, 2015.

ROMANHOLI, Mateus Pedrucci; DE QUEIROZ, Alfredo Pereira. Base hidrográfica otocodificada na escala 1: 25.000: Exemplo da bacia do Córrego Itapiranga (SP). *Caminhos de Geografia*, v. 19, n. 68, p. 46-60, 2018.

SCHEWE KD., Thalheim B. (2008) Semantics in Data and Knowledge Bases. SDKB 2008. Lecture Notes in Computer Science, vol 4925. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-88594-8_1

SERVIGNE, Sylvie et al. A methodology for spatial consistency improvement of geographic databases. *Geoinformatica*, v. 4, n. 1, p. 7-34, 2000.

SERVIGNE S. LESAGE N. LOBOUREL T. Spatial data quality components, standards and metadata. *Fundamentals of Spatial Data Quality. International Scientific and Technical Encyclopedia*. Wiley. 2006. ISBN 1905209568. Pp 179 – 208.

SHEN, Yilang; AI, Tinghua. A hierarchical approach for measuring the consistency of water areas between multiple representations of tile maps with different scales. *ISPRS International Journal of Geo-Information*, v. 6, n. 8, p. 240, 2017.

SHIELDS, Rob. Cultural topology: The seven bridges of Königsburg, 1736. *Theory, Culture & Society*, v. 29, n. 4-5, p. 43-57, 2012.

SIEJKA, M.; ŚLUSARSKI, M.; ZYGMUNT, M. Correction of topological errors in geospatial databases. *International Journal of Physical Sciences*, v. 8, n. 12, p. 498-507, 2013.

SINGH, Sunil Pratap; SINGH, Preetvanti. MODELING A GEO-SPATIAL DATABASE FOR MANAGING TRAVELERS'DEMAND. *International Journal of Database Management Systems*, v. 6, n. 2, p. 39, 2014.

STEMPLIUC, Sergio Murilo. Modelagem de restrições de integridade espaciais em aplicações de rede através do modelo UML-Geoframe. 2008. 74f. Dissertação (Mestrado em Ciência da Computação) – Curso de Pós-Graduação em Ciência da Computação, Universidade Federal de Viçosa, Viçosa.

STROBL C. (2017) Dimensionally Extended Nine-Intersection Model (DE-9IM). In: Shekhar S., Xiong H., Zhou X. (eds) *Encyclopedia of GIS*. Springer, Cham. https://doi.org/10.1007/978-3-319-17885-1_298

Superintendência de Estudos Econômicos e Sociais da Bahia – SEI. 2021. Geoserviços - Baseados em Vetores - Cartografia de Referência - Cartografia 1:25.000. Disponível em: https://www.sei.ba.gov.br/index.php?option=com_content&view=article&id=3346&Itemid=953.

VAN OOSTEROM, Peter; QUAK, Wilko; TIJSSEN, Theo. About invalid, valid and clean polygons. In: *Developments in Spatial Data Handling*. Springer, Berlin, Heidelberg, 2005. p. 1-16.

WADEMBERE, Ismail; OGAO, P. Validation of GIS vector data during geo-spatial alignment. **International Journal of Geoinformatics**, v. 10, n. 4, p. 17-25, 2014.

APÊNDICE A - MEDIDA DE QUALIDADE PARA VALIDAÇÃO INTERCLASSES

Atributo	Descrição																																																						
<i>measureIdentifier</i>	VCL430																																																						
<i>name</i>	Consistência de regras topológicas interclasses																																																						
<i>alias</i>	-																																																						
<i>elementName</i>	Consistência topológica																																																						
<i>definition</i>	Essa medida verifica se os dados são consistentes no que se refere às regras topológicas entre as feições de classes distintas, conforme modelo conceitual, verificando-se os relacionamentos espaciais e cardinalidade.																																																						
<i>description</i>	<p>Cada regra topológica do quadro abaixo deve ser verificada, levantando-se todas as inconsistências e exceções:</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Regra</th> <th>Classe 1</th> <th>Deve</th> <th>Topologia</th> <th>Classe 2</th> <th>Cardinalidade</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>hid_trecho_drenagem_l</td> <td>sim</td> <td>touches</td> <td>hid_massa_dagua_a</td> <td>0..1</td> </tr> <tr> <td>02</td> <td>hid_trecho_drenagem_l</td> <td>sim</td> <td>Within</td> <td>hid_trecho_massa_dagua_a</td> <td>0..1</td> </tr> <tr> <td>03</td> <td>hid_trecho_drenagem_l</td> <td>sim</td> <td>touches</td> <td>hid_ponto_drenagem_p</td> <td>2..2</td> </tr> <tr> <td>04</td> <td>hid_trecho_massa_dagua_a</td> <td>sim</td> <td>contains</td> <td>hid_trecho_drenagem_l</td> <td>1..*</td> </tr> <tr> <td>05</td> <td>hid_ponto_drenagem_p</td> <td>sim</td> <td>touches</td> <td>hid_trecho_drenagem_l</td> <td>1..*</td> </tr> <tr> <td>06</td> <td>hid_ponto_drenagem_p</td> <td>não</td> <td>Within</td> <td>hid_massa_dagua_a</td> <td>1..*</td> </tr> <tr> <td>07</td> <td>hid_trecho_drenagem_l</td> <td>não</td> <td>Crosses</td> <td>hid_massa_dagua_a</td> <td>1..*</td> </tr> <tr> <td>08</td> <td>hid_massa_dagua_a</td> <td>não</td> <td>intersects</td> <td>hid_trecho_massa_dagua_a</td> <td>1..*</td> </tr> </tbody> </table> <p>O resultado da medida será o número total de inconsistências, ou seja, o número de casos que ferem as regras topológicas que não sejam uma exceção justificada.</p>	Regra	Classe 1	Deve	Topologia	Classe 2	Cardinalidade	01	hid_trecho_drenagem_l	sim	touches	hid_massa_dagua_a	0..1	02	hid_trecho_drenagem_l	sim	Within	hid_trecho_massa_dagua_a	0..1	03	hid_trecho_drenagem_l	sim	touches	hid_ponto_drenagem_p	2..2	04	hid_trecho_massa_dagua_a	sim	contains	hid_trecho_drenagem_l	1..*	05	hid_ponto_drenagem_p	sim	touches	hid_trecho_drenagem_l	1..*	06	hid_ponto_drenagem_p	não	Within	hid_massa_dagua_a	1..*	07	hid_trecho_drenagem_l	não	Crosses	hid_massa_dagua_a	1..*	08	hid_massa_dagua_a	não	intersects	hid_trecho_massa_dagua_a	1..*
Regra	Classe 1	Deve	Topologia	Classe 2	Cardinalidade																																																		
01	hid_trecho_drenagem_l	sim	touches	hid_massa_dagua_a	0..1																																																		
02	hid_trecho_drenagem_l	sim	Within	hid_trecho_massa_dagua_a	0..1																																																		
03	hid_trecho_drenagem_l	sim	touches	hid_ponto_drenagem_p	2..2																																																		
04	hid_trecho_massa_dagua_a	sim	contains	hid_trecho_drenagem_l	1..*																																																		
05	hid_ponto_drenagem_p	sim	touches	hid_trecho_drenagem_l	1..*																																																		
06	hid_ponto_drenagem_p	não	Within	hid_massa_dagua_a	1..*																																																		
07	hid_trecho_drenagem_l	não	Crosses	hid_massa_dagua_a	1..*																																																		
08	hid_massa_dagua_a	não	intersects	hid_trecho_massa_dagua_a	1..*																																																		
<i>basicMeasure</i>	Contagem de erro																																																						
<i>valueType</i>	Inteiro																																																						
<i>valueStructure</i>	-																																																						
<i>example</i>	<p>Total de 12 inconsistências:</p> <p>02 inconsistências para regra 04 para as feições de ID 48 e 98;</p> <p>04 inconsistências para regra 05 para as feições de ID 9, 11, 31 e 88;</p> <p>06 inconsistências para regra 07 para as feições de ID 3, 8, 19, 22, 67 e 97.</p>																																																						

<i>parameter</i>	-
<i>sourceReference</i>	-

**APÊNDICE B - MEDIDA DE QUALIDADE PARA VALIDAÇÃO DE REGRAS
ENTRE ATRIBUTOS**

Atributo	Descrição																														
<i>measureIdentifier</i>	VCL510																														
<i>name</i>	Consistência de regras entre atributos																														
<i>alias</i>	-																														
<i>elementName</i>	Consistência semântica																														
<i>definition</i>	<p>Essa medida verifica regras condicionais (se, então) entre dois atributos. Ou seja, quando um atributo satisfaz uma condição A, então outro atributo deve satisfazer uma condição B.</p> <p>Regras bicondicionais do tipo “se e somente se” também podem ser utilizados, desde que se verifique as duas condições (ida e volta).</p>																														
<i>description</i>	<p>Cada regra entre atributos do quadro abaixo deve ser verificada, levantando-se todas as inconsistências e exceções:</p> <table border="1"> <thead> <tr> <th>Regra</th> <th>Classe</th> <th>Atributo 1</th> <th>Atributo 2</th> <th>Condição</th> <th>Se e somente se</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>hid_massa_dagua_a</td> <td>tipoMassa Dagua</td> <td>salinidade</td> <td>Se atr1 ∈ {Oceano, Baía, Enseada}, então atr2 ∈ {Salgada} ou Se atr1 ∈ {Lago, Lagoa, Represa/Açude}, então atr2 ∈ {Doce}</td> <td>Sim</td> </tr> <tr> <td>02</td> <td>hid_trecho_massa_dagua_a</td> <td>tipoTrecho Massa</td> <td>salinidade</td> <td>Se atr1 ∈ {Laguna} então atr2 ∈ {Salgada} ou Se atr1 ∈ {Rio, Canal, Represa/Açude}, então atr2 ∈ {Doce}</td> <td>Sim</td> </tr> <tr> <td>03</td> <td>hid_trecho_drenagem_l</td> <td>regime</td> <td>navegabilidade</td> <td>Se atr1 ∈ {Seco}, então atr2 ∈ {Não navegável}</td> <td>Não</td> </tr> <tr> <td>04</td> <td>hid_trecho_drenagem_l</td> <td>dentroDe Poligono</td> <td>coincideCom DentroDe</td> <td>Se atr1 ∈ {Sim} então atr2 ∈ {Rio, Canal, Laguna, Represa/Açude, Corredeira, Eclusa, Terreno de inundação, Foz Marítima, Barragem}</td> <td>Não</td> </tr> </tbody> </table>	Regra	Classe	Atributo 1	Atributo 2	Condição	Se e somente se	01	hid_massa_dagua_a	tipoMassa Dagua	salinidade	Se atr1 ∈ {Oceano, Baía, Enseada}, então atr2 ∈ {Salgada} ou Se atr1 ∈ {Lago, Lagoa, Represa/Açude}, então atr2 ∈ {Doce}	Sim	02	hid_trecho_massa_dagua_a	tipoTrecho Massa	salinidade	Se atr1 ∈ {Laguna} então atr2 ∈ {Salgada} ou Se atr1 ∈ {Rio, Canal, Represa/Açude}, então atr2 ∈ {Doce}	Sim	03	hid_trecho_drenagem_l	regime	navegabilidade	Se atr1 ∈ {Seco}, então atr2 ∈ {Não navegável}	Não	04	hid_trecho_drenagem_l	dentroDe Poligono	coincideCom DentroDe	Se atr1 ∈ {Sim} então atr2 ∈ {Rio, Canal, Laguna, Represa/Açude, Corredeira, Eclusa, Terreno de inundação, Foz Marítima, Barragem}	Não
Regra	Classe	Atributo 1	Atributo 2	Condição	Se e somente se																										
01	hid_massa_dagua_a	tipoMassa Dagua	salinidade	Se atr1 ∈ {Oceano, Baía, Enseada}, então atr2 ∈ {Salgada} ou Se atr1 ∈ {Lago, Lagoa, Represa/Açude}, então atr2 ∈ {Doce}	Sim																										
02	hid_trecho_massa_dagua_a	tipoTrecho Massa	salinidade	Se atr1 ∈ {Laguna} então atr2 ∈ {Salgada} ou Se atr1 ∈ {Rio, Canal, Represa/Açude}, então atr2 ∈ {Doce}	Sim																										
03	hid_trecho_drenagem_l	regime	navegabilidade	Se atr1 ∈ {Seco}, então atr2 ∈ {Não navegável}	Não																										
04	hid_trecho_drenagem_l	dentroDe Poligono	coincideCom DentroDe	Se atr1 ∈ {Sim} então atr2 ∈ {Rio, Canal, Laguna, Represa/Açude, Corredeira, Eclusa, Terreno de inundação, Foz Marítima, Barragem}	Não																										

	O resultado da medida será o número total de inconsistências, ou seja, o número de casos que ferem as regras entre atributos que não sejam uma exceção justificada.
<i>basicMeasure</i>	Contagem de erro
<i>valueType</i>	Inteiro
<i>valueStructure</i>	-
<i>example</i>	Total de 09 inconsistências: 02 inconsistências para regra 03 para as feições de ID 84 e 103; 07 inconsistências para regra 04 para as feições de ID 7, 18, 29, 32, 66 79 e 98. 01 exceção da regra 01 para a feição de ID 21, justificativa: a salinidade da lagoa foi comprovada em campo como sendo “salgada”.
<i>parameter</i>	-
<i>sourceReference</i>	-

**APÊNDICE C - MEDIDA DE QUALIDADE PARA VALIDAÇÃO DE REGRAS
ENTRE ATRIBUTO E TOPOLOGIA**

Atributo	Descrição																		
<i>measureIdentifier</i>	VCL520																		
<i>name</i>	Consistência de regras entre atributo e topologia																		
<i>alias</i>	-																		
<i>elementName</i>	Consistência semântica																		
<i>definition</i>	Essa medida verifica a coerência de atributos que estão associados às características topológicas da feição com outra a que estiver relacionada.																		
<i>description</i>	<p>Cada regra entre atributo e topologia do quadro abaixo deve ser verificada, levantando-se todas as inconsistências e exceções:</p> <table border="1"> <thead> <tr> <th>Regra</th> <th>Classe 1</th> <th>Atributo</th> <th>Classe 2</th> <th>Condição</th> <th>Se e somente se</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>hid_ponto_drenagem_p</td> <td>relacionado</td> <td>hid_massa_dagua_a</td> <td>Se $atr \in \{Lago, Lagoa, Meandro abandonado\}$ então geometria1 toca geometria2</td> <td>Sim</td> </tr> <tr> <td>02</td> <td>hid_trecho_drenagem_l</td> <td>Coincide ComDentroDe</td> <td>hid_trecho_massa_dagua_a</td> <td>Se $atr \in \{Rio, Canal, Represa/açude, Laguna\}$ então geometria1 dentro de geometria2</td> <td>Sim</td> </tr> </tbody> </table> <p>O resultado da medida será o número total de inconsistências, ou seja, o número de casos que ferem as regras entre atributo e topologia que não sejam uma exceção justificada.</p>	Regra	Classe 1	Atributo	Classe 2	Condição	Se e somente se	01	hid_ponto_drenagem_p	relacionado	hid_massa_dagua_a	Se $atr \in \{Lago, Lagoa, Meandro abandonado\}$ então geometria1 toca geometria2	Sim	02	hid_trecho_drenagem_l	Coincide ComDentroDe	hid_trecho_massa_dagua_a	Se $atr \in \{Rio, Canal, Represa/açude, Laguna\}$ então geometria1 dentro de geometria2	Sim
Regra	Classe 1	Atributo	Classe 2	Condição	Se e somente se														
01	hid_ponto_drenagem_p	relacionado	hid_massa_dagua_a	Se $atr \in \{Lago, Lagoa, Meandro abandonado\}$ então geometria1 toca geometria2	Sim														
02	hid_trecho_drenagem_l	Coincide ComDentroDe	hid_trecho_massa_dagua_a	Se $atr \in \{Rio, Canal, Represa/açude, Laguna\}$ então geometria1 dentro de geometria2	Sim														
<i>basicMeasure</i>	Contagem de erro																		
<i>valueType</i>	Inteiro																		
<i>valueStructure</i>	-																		
<i>example</i>	Total de 05 inconsistências: 02 inconsistências para regra 01 para as feições de ID 8 e 29; 03 inconsistências para regra 02 para as feições de ID 20, 33 e 54.																		
<i>parameter</i>	-																		
<i>sourceReference</i>	-																		

APÊNDICE D – SQL PARA INCONSISTÊNCIAS TOPOLÓGICAS

```

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
01      hid_trecho_drenagem_1   sim    touches    hid_massa_dagua_a   0..1
*/

CREATE TABLE tab01_ru101
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_trecho_drenagem_1 deve tocar 0 ou 1 feição da classe hid_massa_dagua_a' AS erro_msg
FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_touches(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_trecho_drenagem_1 T1, cb.hid_massa_dagua_a T2 -- classes
    ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems > 1; -- cardinalidade

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
02      hid_trecho_drenagem_1   sim    within     hid_trecho_massa_dagua_a   0..1
*/

CREATE TABLE tab01_ru102
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_trecho_drenagem_1 deve estar dentro de 0 ou 1 feição da classe
hid_trecho_massa_dagua_a' AS erro_msg
FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_within(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_trecho_drenagem_1 T1, cb.hid_trecho_massa_dagua_a T2 -- classes
    ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems > 1 -- cardinalidade

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
03      hid_trecho_drenagem_1   sim    within     hid_ponto_drenagem_p   2..2
*/

CREATE TABLE tab01_ru103
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_trecho_drenagem_1 deve tocar exatamente 2 feições da classe hid_ponto_drenagem_p' AS
erro_msg

```

```

FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_touches(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_trecho_drenagem_l T1, cb.hid_ponto_drenagem_p T2 -- classes
  ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems <> 2; -- cardinalidade

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
04      hid_trecho_massa_dagua_a   sim    contains    hid_trecho_drenagem_l   1..*
*/

CREATE TABLE tab01_rul04
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_trecho_massa_dagua_a deve conter pelo menos 1 feição da classe hid_trecho_drenagem_l'
AS erro_msg
FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_contains(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_trecho_massa_dagua_a T1, cb.hid_trecho_drenagem_l T2 -- classes
  ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems = 0; -- cardinalidade

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
05      hid_ponto_drenagem_p         sim    touches     hid_trecho_drenagem_l   1..*
*/

CREATE TABLE tab01_rul05
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_ponto_drenagem_p deve tocar pelo menos 1 feição da classe hid_trecho_drenagem_l' AS
erro_msg
FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_touches(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_ponto_drenagem_p T1, cb.hid_trecho_drenagem_l T2 -- classes
  ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems = 0; -- cardinalidade

```

```

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
06      hid_ponto_drenagem_p   não    within      hid_massa_dagua_a   1..*
*/

```

```

CREATE TABLE tab01_rul06
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_ponto_drenagem_p NÃO deve estar dentro de mais de 0 feição da classe
hid_massa_dagua_a' AS erro_msg
FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_within(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_ponto_drenagem_p T1, cb.hid_massa_dagua_a T2 -- classes
    ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems > 0; -- cardinalidade

```

```

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
07      hid_trecho_drenagem_l   não    crosses     hid_massa_dagua_a   1..*
*/

```

```

CREATE TABLE tab01_rul07
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_trecho_drenagem_l NÃO deve cruzar de mais de 0 feição da classe hid_massa_dagua_a' AS
erro_msg
FROM(
  SELECT
    C.id AS id,
    C.geom AS geom,
    sum(C.relation::int) AS n_elems
  FROM(
    SELECT
      T1.id AS id,
      T1.geom AS geom,
      st_crosses(T1.geom, T2.geom) AS relation -- relação topológica
    FROM cb.hid_trecho_drenagem_l T1, cb.hid_massa_dagua_a T2 -- classes
    ) C
  GROUP BY C.id, C.geom
) V
WHERE n_elems > 0; -- cardinalidade

```

```

/* Quadro 17
Regra   Classe 1           Deve   Topologia   Classe 2           Cardinalidade
08      hid_massa_dagua_a       não    intersects  hid_trecho_massa_dagua_a   1..*
*/

```

```

CREATE TABLE tab01_rul08
AS
SELECT
V.id AS id,
V.geom AS geom
--, 'hid_massa_dagua_a NÃO deve interceptar mais de 0 feição da classe
hid_trecho_massa_dagua_a' AS erro_msg
FROM(
  SELECT

```

```
C.id AS id,
C.geom AS geom,
sum(C.relation::int) AS n_elems
FROM(
  SELECT
    T1.id AS id,
    T1.geom AS geom,
    st_intersects(T1.geom, T2.geom) AS relation -- relação topológica
  FROM cb.hid_massa_dagua_a T1, cb.hid_trecho_massa_dagua_a T2 -- classes
) C
GROUP BY C.id, C.geom
) v
WHERE n_elems > 0; -- cardinalidade
```

APÊNDICE E – SQL PARA INCONSISTÊNCIAS SEMÂNTICAS ENTRE ATRIBUTOS

```

/* Quadro 18
Regra Classe Atributo 1 Atributo 2 Se e somente se
01 hid massa dagua a tipoMassaDagua salinidade sim

Condição:
Se atr1 ∈ {Oceano, Baía, Enseada}, então atr2 ∈ {Salgada}
ou
Se atr1 ∈ {Lago, Lagoa, Represa/Açude}, então atr2 ∈ {Doce}

0 "Desconhecida"
3 "Oceano"
4 "Baía"
5 "Enseada"
6 "Meandro Abandonado"
7 "Lago"
8 "Lagoa"
10 "Represa/Açude"
999 "A SER PREENCHIDO"

0 "Desconhecida"
1 "Doce"
2 "Salgada"
999 "A SER PREENCHIDO"
*/

--CREATE TABLE tab02_rul01 AS
SELECT
B.id AS id,
B.geom AS geom,
concat('tipoMassaDagua=', B.tipoMassaDagua, ' não é compatível com salinidade=', B.salinidade)
AS erro_msg
FROM(
SELECT
C.id AS id,
C.geom AS geom,
C.tipoMassaDagua AS tipoMassaDagua,
C.salinidade AS salinidade,
CASE
WHEN C.tipoMassaDagua IN (3,4,5) THEN C.salinidade IN (2)
WHEN C.salinidade IN (2) THEN C.tipoMassaDagua IN (3,4,5)
WHEN C.tipoMassaDagua IN (6,7,8,10) THEN C.salinidade IN (1)
WHEN C.salinidade IN (1) THEN C.tipoMassaDagua IN (6,7,8,10)
ELSE true
END AS relacao
FROM cb.hid_massa_dagua_a C
)B
WHERE NOT B.relacao;

/* Quadro 18
Regra Classe Atributo 1 Atributo 2 Se e somente se
02 hid_trecho_massa_dagua_a tipoTrechoMassa salinidade sim

Condição:
Se atr1 ∈ {Laguna} então atr2 ∈ {Salgada}
ou
Se atr1 ∈ {Rio, Canal, Represa/Açude}, então atr2 ∈ {Doce}

1 "Rio"
2 "Canal"
9 "Laguna"
10 "Represa/açude"
99 "Outros"
999 "A SER PREENCHIDO"

0 "Desconhecida"

```

```

1  "Doce"
2  "Salgada"
999 "A SER PREENCHIDO"
*/

-- IDA E VOLTAS
--CREATE TABLE tab02_rul02 AS
SELECT
B.id AS id,
B.geom AS geom,
concat('tipoTrechoMassa=', B.tipoTrechoMassa, ' não é compatível com salinidade=',
B.salinidade) AS erro_msg
FROM(
  SELECT
  C.id AS id,
  C.geom AS geom,
  C.tipoTrechoMassa AS tipoTrechoMassa,
  C.salinidade AS salinidade,
  CASE
  WHEN C.tipoTrechoMassa IN (9) THEN C.salinidade IN (2)
  WHEN C.salinidade IN (2) THEN C.tipoTrechoMassa IN (9)
  WHEN C.tipoTrechoMassa IN (1,2,10) THEN C.salinidade IN (1)
  WHEN C.salinidade IN (1) THEN C.tipoTrechoMassa IN (1,2,10)
  ELSE true
  END AS relacao
  FROM cb.hid_trecho_massa_dagua_a C
)B
WHERE NOT B.relacao;

/* Quadro 18
Regra   Classe                Atributo 1      Atributo 2      Se e somente se
03      hid_trecho_drenagem_1    regime          navegabilidade   não

Condição:
Se atr1 ∈ {Seco}, então atr2 ∈ {Não navegável}

0  "Desconhecido"
1  "Permanente"
2  "Permanente com grande variação"
3  "Temporário"
4  "Temporário com leito permanente"
5  "Seco"
6  "Sazonal"
999 "A SER PREENCHIDO"

0  "Desconhecida"
1  "Navegável"
2  "Não navegável"
999 "A SER PREENCHIDO"
*/

-- SOMENTE IDA
CREATE TABLE tab02_rul03 AS
SELECT
B.id AS id,
B.geom AS geom,
concat('regime=', B.regime, ' não é compatível com navegabilidade=', B.navegabilidade) AS
erro_msg
FROM(
  SELECT
  C.id AS id,
  C.geom AS geom,
  C.regime AS regime,
  C.navegabilidade AS navegabilidade,
  CASE
  WHEN C.regime IN (5) THEN C.navegabilidade IN (2)
  ELSE true
  END AS relacao
  FROM cb.hid_trecho_drenagem_1 C
)B
WHERE NOT B.relacao;

```

```

/* Quadro 18
Regra   Classe                Atributo 1                Atributo 2                Se e somente se
04      hid_trecho_drenagem_1 dentroDePoligono         coincideComDentroDe       sim

Condição:
Se atr1 ∈ {Sim} então atr2 ∈ {Rio, Canal, Laguna, Represa/Açude, Corredeira, Eclusa, Terreno
de inundação, Foz Marítima, Barragem}

1      "Sim"
2      "Não"
999   "A SER PREENCHIDO"

1      "Rio"
2      "Canal"
9      "Laguna"
10     "Represa/açude"
11     "Vala"
12     "Queda d'água"
13     "Corredeira"
14     "Eclusa"
15     "Terreno sujeito a inundação"
16     "Foz marítima"
19     "Barragem"
97     "Não aplicável"
999   "A SER PREENCHIDO"

*/

-- SOMENTE IDA
--CREATE TABLE tab02_rul04 AS
SELECT
B.id as id,
B.geom as geom,
concat('dentroDePoligono=', B.dentroDePoligono, ' não é compatível com coincideComDentroDe=',
B.coincideComDentroDe) AS erro_msg
FROM(
  SELECT
  C.id AS id,
  C.geom AS geom,
  C.dentroDePoligono AS dentroDePoligono,
  C.coincideComDentroDe AS coincideComDentroDe,
  CASE
  WHEN C.dentroDePoligono in (1) THEN C.coincideComDentroDe IN (1,2,9,10,13,14,15,16,19)
  ELSE true
  END AS relacao
  FROM cb.hid_trecho_drenagem_1 C
)B
WHERE NOT B.relacao;

```

APÊNDICE F – SQL PARA INCONSISTÊNCIAS SEMÂNTICAS ENTRE ATRIBUTO E TOPOLOGIA

```

/* Quadro 19
Regra   Classe 1           Atributo           Classe 2           Se e somente se
01      hid_ponto_drenagem_p  relacionado        hid_massa_dagua_a  sim

Condição:
Se atr ∈ {Lago, Lagoa, Meandro abandonado} então geometria1 toca geometria2

1  "Eclusa"
2  "Barragem"
3  "Comporta"
4  "Queda d'água"
5  "Corredeira"
6  "Foz marítima"
7  "Sumidouro"
8  "Meandro abandonado"
9  "Lago"
10 "Lagoa"
11 "Laguna"
12 "Represa/ açude"
13 "Entre trechos de drenagem"
16 "Vertedouro"
17 "Interrupção à Jusante"
18 "Interrupção à Montante"
999 "A SER PREENCHIDO"

*/

CREATE TABLE tab03_rul01 AS
SELECT
B.id AS id,
B.geom AS geom,
concat('relacionado=', B.relacionado, ' não é compatível com a sua topologia') AS erro_msg
FROM(
SELECT
C.id AS id,
C.geom AS geom,
C.relacionado AS relacionado,
CASE
WHEN C.relacionado IN (8,9,10) THEN (SELECT sum(st_touches(C.geom, D.geom)::int) FROM
cb.hid_massa_dagua_a D)
WHEN (SELECT sum(st_touches(C.geom, D.geom)::int) FROM cb.hid_massa_dagua_a D)>0 THEN
C.relacionado IN (8,9,10)::int
ELSE 1
END AS relacao
FROM cb.hid_ponto_drenagem_p C
)B
WHERE B.relacao = 0;

/* Quadro 19
Regra   Classe 1           Atributo           Classe 2           Se e somente
se
02      hid_trecho_drenagem_1  CoincideComDentroDe  hid_trecho_massa_dagua_a  Sim

Condição:
Se atr ∈ { Rio, Canal, Represa/açude, Laguna} então geometria1 dentro de geometria2

1  "Rio"
2  "Canal"
9  "Laguna"
10 "Represa/açude"
11 "Vala"
12 "Queda d'água"
13 "Corredeira"
14 "Eclusa"
15 "Terreno sujeito a inundação"

```

```

16 "Foz marítima"
19 "Barragem"
97 "Não aplicável"
999 "A SER PREENCHIDO"

*/

--CREATE TABLE tab03_rul02 AS
SELECT
B.id AS id,
B.geom AS geom,
concat('CoincideComDentroDe=', B.CoincideComDentroDe, ' não é compatível com a sua topologia')
AS erro_msg
FROM(
  SELECT
  C.id AS id,
  C.geom AS geom,
  C.CoincideComDentroDe AS CoincideComDentroDe,
  CASE
    WHEN C.CoincideComDentroDe IN (1,2,9,10) THEN (SELECT sum(st_within(C.geom,
D.geom)::int) FROM cb.hid_trecho_massa_dagua_a D)
    WHEN (SELECT sum(st_within(C.geom, D.geom)::int) FROM cb.hid_trecho_massa_dagua_a D)>0
  THEN C.CoincideComDentroDe IN (1,2,9,10)::int
    ELSE 1
  END AS relacao
  FROM cb.hid_trecho_drenagem_l C
)B
WHERE B.relacao = 0;

```