

ANÁLISE DE QUALIDADE DE MODELOS BIM: UMA PROPOSTA DE FLUXO DE TRABALHO COM FERRAMENTAS INTEROPERÁVEIS

JOÃO PEDRO CAMPELO DE OLIVEIRA E SILVA

Dissertação submetida para satisfação parcial dos requisitos do grau de
MESTRE EM ENGENHARIA CIVIL — ESPECIALIZAÇÃO EM CONSTRUÇÕES

Orientador: Professor Doutor João Pedro da Silva Poças Martins

JUNHO DE 2025

MESTRADO EM ENGENHARIA CIVIL 2024/2025

DEPARTAMENTO DE ENGENHARIA CIVIL

Tel. +351-22-508 1901

✉ m.ec@fe.up.pt

Editado por

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Rua Dr. Roberto Frias

4200-465 PORTO

Portugal

Tel. +351-22-508 1400

✉ feup@fe.up.pt

🌐 <http://www.fe.up.pt>

Reproduções parciais deste documento serão autorizadas na condição que seja mencionado o Autor e feita referência a *Mestrado em Engenharia Civil - 2024/2025 - Departamento de Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, 2025*.

As opiniões e informações incluídas neste documento representam unicamente o ponto de vista do respetivo Autor, não podendo o Editor aceitar qualquer responsabilidade legal ou outra em relação a erros ou omissões que possam existir.

Este documento foi produzido a partir de versão eletrónica fornecida pelo respetivo Autor.

Aos meus Pais

“Juízo.”

Anderson Silva Jr.

AGRADECIMENTOS

Agradeço ao meu orientador, Professor João Pedro da Silva Poças Martins, por toda a orientação, pela paciência e pela forma como soube equilibrar exigência e liberdade ao longo deste processo.

Ao meu supervisor na GEG, Joel Soares, pela forma generosa como me recebeu na equipe, pelos desafios propostos e pela constante partilha de conhecimento. Obrigado por ter acreditado que eu podia contribuir com um tema tão atual e por ter alimentado essa investigação desde o primeiro dia.

À empresa GEG, pelo acolhimento, pelos materiais fornecidos e pela oportunidade de crescer num ambiente tão rico em experiências.

Ao Gustavo, Guilherme, Sónia, Henrique e todos os Joãos do departamento de BIM. Obrigado pelo interesse sincero na minha tese, pela ajuda nos momentos certos e por terem sido companhia nos almoços e ao longo de todo o percurso. Aprendi muito com vocês, e fico feliz por ter feito parte dessa equipe.

Aos colegas que me acompanharam ao longo destes anos de faculdade. Cada partilha, risada, ou desabafo teve o seu peso. E um agradecimento especial àqueles que enfrentaram a etapa da dissertação comigo: pelas conversas, pelas fofocas técnicas, pelo apoio mútuo e pela sensação de que, mesmo no caos, estávamos juntos.

À Bela, que entrou na minha vida no segundo ano da faculdade e nunca mais saiu. Parceira em tudo, no amor, na rotina, na saudade de casa. Vivemos o desafio de viver longe juntos, e foi com você que entendi o que é sentir-se em casa, mesmo longe. Obrigado por estar do meu lado nos dias mais difíceis e por me amar com tanta força. Você é meu porto-seguro e nossa caminhada só está começando.

À minha irmã, com quem divido não só o sangue, mas agora também espaço nessa indústria. Você é uma inspiração silenciosa, exemplo de esforço e determinação, e eu te amo mais do que jamais vou conseguir escrever aqui. Que nós nunca paremos de aprender um com o outro. E que você fique rica logo.

À minha mãe, por um amor que nunca falha e por uma força que vai muito além do que se vê. Mesmo nos dias mais difíceis, nunca soltamos a mão um do outro. Crescemos juntos e nos reconstruímos juntos. Obrigado por ter me mostrado que vulnerabilidade também é coragem, e por me ensinar, com cada conversa e cada silêncio, o que é amar de verdade. Você é parte do que me fez chegar até aqui e, acima de tudo, é parte do que me faz continuar.

Ao meu pai, que mesmo a milhares de quilômetros, nunca deixou de estar por perto. Foram muitas as tentativas de vir me ver nesses anos — barradas por pandemias, doenças, diagnósticos duros e a vida que insiste em testar os fortes. E você foi forte. Enfrentou tudo com coragem, dignidade e, mesmo à distância, nunca deixou de me acompanhar. Obrigado por me mostrar o que é lutar, recomeçar e seguir em frente mesmo quando parece impossível. Hoje nos entendemos mais do que nunca, e a cada conversa sinto o quanto você se orgulha de mim. Saiba que esse orgulho é recíproco, pai. Você é parte de tudo isso.

Por fim, agradeço a todas as pessoas que, de alguma forma, cruzaram este caminho comigo — pelas palavras, pelos silêncios, pelas oportunidades e pela presença. Esta tese é fruto de muitas mãos, corações e histórias que me rodeiam.

RESUMO

A ausência de um processo estruturado e automatizado para a verificação da qualidade informacional de modelos BIM continua a ser um problema recorrente em muitas organizações. A verificação é frequentemente realizada de forma manual, fragmentada e pouco rastreável, dificultando o controle de requisitos, a padronização de dados e a confiabilidade das informações ao longo do ciclo de vida dos projetos.

Embora a verificação automática da informação já seja tema de pesquisa desde a década de 1970, o setor da Arquitetura, Engenharia e Construção (AEC) ainda enfrenta limitações significativas ligadas à fragmentação de ferramentas, à baixa interoperabilidade e à forte dependência de soluções proprietárias. Nesse cenário, os padrões abertos emergem como uma alternativa sólida e durável para a sistematização do controle de qualidade em ambientes BIM.

Para responder a esse desafio, esta dissertação propõe um fluxo de trabalho automatizado baseado em *Information Delivery Specification* (IDS), desenvolvido e testado em parceria com uma empresa de engenharia. A abordagem abrange todas as etapas da verificação: desde a definição de requisitos informacionais, passando pela sua formalização em ficheiros IDS, até a execução das verificações e a gestão estruturada das não conformidades. O fluxo foi aplicado a um projeto real, envolvendo cinco modelos disciplinares, permitindo validar sua viabilidade técnica e sua adaptabilidade a diferentes contextos de modelação.

Os resultados obtidos demonstram que a proposta é tecnicamente viável, mesmo com ferramentas acessíveis e *open source*, e que oferece ganhos significativos em rastreabilidade, clareza de requisitos e eficiência do processo. A dissertação contribui não apenas com uma solução prática e replicável, mas também com uma base metodológica atualizada, alinhada às exigências crescentes por padronização, interoperabilidade e integração digital no setor da construção.

PALAVRAS-CHAVE: BIM, *Automated Compliance Checking*, *OpenBIM*, *Information Delivery Specification*, Bonsai.

ABSTRACT

The lack of a structured and automated process for verifying the information quality of BIM models remains a recurring issue in many organizations. Verification is often performed manually, in a fragmented and poorly traceable manner, making it difficult to control requirements, standardize data, and ensure the reliability of information throughout the project lifecycle.

Although Automated Compliance Checking has been a topic of research since the 1970s, the Architecture, Engineering and Construction (AEC) sector still faces significant limitations related to fragmented tools, low interoperability, and strong dependence on proprietary solutions. In this context, open standards are emerging as a solid and long-term alternative for systematizing quality control in BIM environments.

To address this challenge, this dissertation proposes an automated workflow based on Information Delivery Specification (IDS), developed and tested in partnership with an engineering company. The approach covers all stages of the verification process — from defining information requirements and formalizing them in IDS files to performing automated checks and managing identified nonconformities. The workflow was applied to a real project involving five disciplinary models, allowing the assessment of its technical feasibility and adaptability to different modeling contexts.

The results obtained demonstrate that the proposed solution is technically feasible, even when using accessible and open-source tools, and that it provides significant improvements in traceability, clarity of requirements, and overall process efficiency. This dissertation contributes not only with a practical and replicable solution but also with an updated methodological foundation aligned with the growing demands for standardization, interoperability, and digital integration in the construction sector.

KEYWORDS: BIM, Automated Compliance Checking, OpenBIM, Information Delivery Specification, Bonsai

ÍNDICE GERAL

AGRADECIMENTOS	I
RESUMO	III
ABSTRACT	V
ÍNDICE DE FIGURAS	X
ÍNDICE DE TABELAS.....	XIV
SÍMBOLOS, ACRÓNIMOS E ABREVIATURAS	XVI
1. INTRODUÇÃO	1
1.1. MOTIVAÇÃO	1
1.2. OBJETIVOS.....	1
1.3. ESTRUTURA DA DISSERTAÇÃO.....	2
2. ESTADO DA ARTE	5
2.1. QUALITY ASSURANCE (QA) E QUALITY CONTROL (QC) NO CONTEXTO BIM.....	5
2.1.1. DEFINIÇÃO DE QUALIDADE E SUA APLICAÇÃO NA INDÚSTRIA AEC	5
2.1.2. CONCEITOS DE QUALITY ASSURANCE (QA) E QUALITY CONTROL (QC)	6
2.1.3. MÉTODOS EXISTENTES DE MENSURAÇÃO DE QUALIDADE NOS MODELOS E PROCESSOS BIM.....	7
2.1.4. DESAFIOS E LIMITAÇÕES ATUAIS	8
2.2. O “I” DO BIM E SEUS REQUISITOS	8
2.2.1. REQUISITOS DE INFORMAÇÃO E SEUS MÉTODOS DE ESPECIFICAÇÃO	9
2.3. INTEROPERABILIDADE, OPENBIM E A IMPORTÂNCIA DOS PADRÕES ABERTOS	10
2.4. VERIFICAÇÃO AUTOMÁTICA DE CONFORMIDADE DE MODELOS BIM	12
2.4.1. INTRODUÇÃO À VERIFICAÇÃO AUTOMÁTICA DE CONFORMIDADE	12
2.4.2. ESTÁGIOS DA VERIFICAÇÃO AUTOMÁTICA.....	14
2.4.2.1. Interpretação da regra.....	15
2.4.2.2. Preparação do modelo.....	15
2.4.2.3. Execução de regras	16
2.4.2.4. Reportagem de regras	16
2.5. SISTEMAS DE VERIFICAÇÃO AUTOMÁTICA EXISTENTES	17
2.5.1. INICIATIVAS INSTITUCIONAIS.....	17
2.5.1.1. CORENET - Singapura	17
2.5.1.2. DesignCheck – Austrália	18
2.5.1.3. SMARTcodes – Estados Unidos	18

2.5.1.4. Statsbygg – Noruega	19
2.5.2. SISTEMAS PROPRIETÁRIOS	19
2.5.3. ABORDAGENS BASEADAS EM PADRÕES ABERTOS	23
2.5.3.1. Tecnologias Semânticas e Ontológicas.....	23
2.5.3.2. O fluxo OpenBIM da buildingSMART	24
2.5.4. EXPERIMENTOS ACADÉMICOS	26
2.5.5. ABORDAGENS RECENTES	30
2.6. SÍNTESE DO ESTADO DA ARTE	31
2.6.1. PRINCIPAIS LIMITAÇÕES	31
2.6.2. LACUNAS DE PESQUISA E OPORTUNIDADES	32
 3. METODOLOGIA	 33
3.1 INTRODUÇÃO	33
3.2 DESCRIÇÃO DA METODOLOGIA	33
3.3 FERRAMENTAS UTILIZADAS	35
3.1.1 USBIM IDS EDITOR.....	35
3.3.2 BLENDER E BONSAI	35
3.3.2 BIMCOLLAB ZOOM.....	36
3.4 MODELO FEDERADO	36
3.5 DEFINIÇÃO DOS REQUISITOS A SEREM VERIFICADOS	36
3.6 LIMITAÇÕES	37
 4. ESTUDO DE CASO	 39
4.1 INTRODUÇÃO	39
4.2 CARACTERIZAÇÃO DOS MODELOS	39
4.2.1. MODELO ARQUITETÓNICO	39
4.2.2. MODELO ESTRUTURAL	40
4.2.1. MODELO ELÉTRICO	40
4.2.1. MODELO HVAC	41
4.2.1. MODELO DE SISTEMAS HIDRÁULICOS	41
4.3 PRÉ-VERIFICAÇÃO SINTÁTICA DOS MODELOS.....	42
4.4 CRIAÇÃO DOS FICHEIROS IDS	43
4.5 FICHEIROS XML E PDF CRIADOS.....	47
4.6 PROCESSO DE VERIFICAÇÃO AUTOMÁTICA	48
4.7 GESTÃO DE NÃO-CONFORMIDADES	50

5. ANÁLISE DOS RESULTADOS	53
5.1 INTRODUÇÃO	53
5.2 FICHEIROS IDS CRIADOS	53
5.3 RESULTADOS DA VERIFICAÇÃO AUTOMÁTICA	55
5.4 GESTÃO DE NÃO CONFORMIDADES	58
6. CONCLUSÕES	61
6.1 CONSIDERAÇÕES INICIAIS	61
6.2 CONCLUSÕES SOBRE OS RESULTADOS	62
6.2.1. AVALIAÇÃO DO FLUXO PROPOSTO E FERRAMENTAS UTILIZADAS	62
6.2.2. FLEXIBILIDADE, INTERPRETAÇÃO E ADAPTAÇÃO À PRÁTICA	62
6.2.3. BARREIRAS À INTEGRAÇÃO E CONTRIBUTOS DO ESTUDO	63
6.3 TRABALHOS FUTUROS	64
REFERÊNCIAS BIBLIOGRÁFICAS	65
ANEXOS	68
A1 FICHEIROS IDS CRIADOS	70
A2 FICHEIROS XML CRIADOS	81
A3 RELATÓRIOS HTML.....	95

ÍNDICE DE FIGURAS

Figura 1 - Curva de Macleamy (Lu et al., 2014)	6
Figura 2 - Métodos de especificação de informação	10
Figura 3 - Linha do tempo dos principais sistemas automáticos de verificação de conformidade (Amor & Dimyadi, 2021).....	13
Figura 4 - Estágios de Eastman (Eastman et al., 2009)	15
Figura 5 - Clash check do Solibri Checkpoint.....	21
Figura 6 - Property Check do Solibri Checkpoint.....	22
Figura 7 - Free Space Check do Solibri Checkpoint.....	22
Figura 8 - Conexões do Solibri Checkpoint	23
Figura 9 - Fluxo de trabalho proposto pela buildingSMART (buildingSMART, 2025a).....	25
Figura 10 - Estudos académicos analisados	28
Figura 11 - Fluxograma do processo proposto.....	34
Figura 12 - Fluxograma adaptado do processo proposto	38
Figura 13 - Modelo Arquitetónico	40
Figura 14 - Modelo Estrutural.....	40
Figura 15 - Modelo Elétrico	41
Figura 16 - Modelo HVAC	41
Figura 17 - Modelo de Sistemas Hidráulicos	42
Figura 18 - Lista de elementos IFC criada a partir do Blender.....	43
Figura 19 - Lista de elementos IFC de cada especialidade	43
Figura 20 - Aba identificação do editor IDS	44
Figura 21 - Aba filtros do editor IDS	44
Figura 22 - Aba requisitos do editor IDS	45
Figura 23 - Audit tool acusa erro de sintaxe	46
Figura 24 - Audit tool não acusa nenhum erro de sintaxe	46
Figura 25 - Exemplo de ficheiro PDF e XML a partir do IDS	47
Figura 26 - Modelos de cada especialidade abertos no Blender	48
Figura 27 - IFC Tester.....	48
Figura 28 - Excerto do relatório HTML	49
Figura 29 - Processo para visualizar as não-conformidades detetadas.....	49
Figura 30 - Projeto BCF preenchido.....	50
Figura 31 - <i>Issue</i> criado no Blender e importado no BimCollab Zoom.....	51
Figura 32 - Demonstração da distinção dos <i>Spatial Containers</i> no Blender	55

Figura 33 - Comparação de práticas de criação de especificações.....	57
Figura 34 - Menu suspenso do BIMCollab Zoom	58
Figura 35 - Sugestão de alteração da interface de usuário do Bonsai.	59
Figura 36 - Visualização de issues em ferramenta autoral	60
Figura 37 - Visualização de issues em ferramenta autoral	60
Figura 38 - Relatório IDS do Modelo Arquitetônico (Parte 1).....	71
Figura 39 - Relatório IDS do Modelo Arquitetônico (Parte 2).....	72
Figura 40 - Relatório IDS do Modelo Arquitetônico (Parte 3).....	73
Figura 41 - Relatório IDS do Modelo Arquitetônico (Parte 4).....	74
Figura 42 - Relatório IDS do Modelo Estrutural (Parte 1)	75
Figura 43 - Relatório IDS do Modelo Estrutural (Parte 2)	76
Figura 44 - Relatório IDS do Modelo de Sistemas Hidráulicos (<i>Plumbing</i>)	77
Figura 45 - Relatório IDS do Modelo HVAC (Parte 1)	78
Figura 46 - Relatório IDS do Modelo HVAC (Parte 2)	79
Figura 47 – Relatório IDS do Modelo Elétrico	80
Figura 48 - XML do IDS do Modelo Arquitetônico (Parte 1).....	82
Figura 49 - XML do IDS do Modelo Arquitetônico (Parte 2).....	83
Figura 50 - XML do IDS do Modelo Arquitetônico (Parte 3).....	84
Figura 51 - XML do IDS do Modelo Arquitetônico (Parte 4).....	85
Figura 52 - XML do IDS do Modelo Arquitetônico (Parte 5).....	86
Figura 53 - XML do IDS do Modelo Arquitetônico (Parte 6).....	87
Figura 54 - XML do IDS do Modelo Estrutural (Parte 1)	88
Figura 55 - XML do IDS do Modelo Estrutural (Parte 2)	89
Figura 56 - XML do IDS do Modelo Estrutural (Parte 3)	90
Figura 57 - XML do IDS do Modelo de Sistemas Hidráulicos (<i>Plumbing</i>)	91
Figura 58 - XML do IDS do Modelo HVAC (Parte 1)	92
Figura 59 - XML do IDS do Modelo HVAC (Parte 2)	93
Figura 60 - XML do IDS do Modelo Elétrico	94
Figura 61 - Relatório HTML do Modelo Arquitetônico (Parte 1).....	96
Figura 62 - Relatório HTML do Modelo Arquitetônico (Parte 2).....	97
Figura 63 - Relatório HTML do Modelo Arquitetônico (Parte 3).....	98
Figura 64 - Relatório HTML do Modelo Arquitetônico (Parte 4).....	99
Figura 65 - Relatório HTML do Modelo Estrutural (Parte 1)	100
Figura 66 - Relatório HTML do Modelo Estrutural (Parte 2)	101

Figura 67 - Relatório HTML do Modelo de Sistemas Hidráulicos	102
Figura 68 - Relatório HTML do Modelo HVAC (Parte 1).....	103
Figura 69 - Relatório HTML do Modelo HVAC (Parte 2).....	104
Figura 70 - Relatório HTML do Modelo Elétrico	105

ÍNDICE DE TABELAS

Tabela 1 - Síntese das iniciativas institucionais	19
Tabela 2 - Estudos anteriores sobre soluções de controle de qualidade baseadas em BIM. (Adaptado de M. Valinejadshoubi et al., 2024).....	27
Tabela 3 - Requisitos a serem verificados	37
Tabela 4 - Síntese dos parâmetros a serem avaliados por modelo	53
Tabela 5 - Síntese dos elementos verificados.....	55

SÍMBOLOS, ACRÓNIMOS E ABREVIATURAS

AEC - Arquitetura, Engenharia e Construção

ABCB - *Australian Building Codes Board*

ACC - *Automated Compliance Checking*

ACC - *Autodesk Construction Cloud*

AIM - *Asset Information Model*

AIR - *Asset Information Requirements*

API - *Application Programming Interface*

BCA - *Building Construction Authority*

BIM – *Building Information Modeling*

BCF – *BIM Collaboration Format*

bSDD – *buildingSMART Data Dictionary*

CDE – *Common Data Environment*

COBie – *Construction Operations Building Information Exchange*

CDE - Ambiente comum de dados (*Common Data Environment*)

EIR - *Exchange Information Requirements*

HTML – *HyperText Markup Language*

HVAC – *Heating, Ventilation and Air Conditioning*

ICC – *International Code Council*

IDS – *Information Delivery Specification*

IDM – *Information Delivery Manual*

IFC – *Industry Foundation Class*

IR – *Information Requirements*

ISO – *International Organization for Standardization*

JSON – *JavaScript Object Notation*

LLM – *Large Language Model*

LOD – *Level of Development*

LOIN – *Level of Information Needed*

MEP – *Mechanical, Electrical and Plumbing systems*

MVD – *Model View Definition*

OWL – *Web Ontology Language*

OIR - *Organizational Information Requirements*

PDF – *Portable Document Format*

PIM - *Project Information Model*

PIR - *Project Information Requirements*

Pset – *Property Set*

QA – *Quality Assurance*

QC – *Quality Control*

SHACL – *Shapes Constraint Language*

SMC – *Solibri Model Checker*

SPARQL – *SPARQL Protocol and RDF Query Language*

SWRL – *Semantic Web Rule Language*

UE – *União Europeia*

XML – *Extensible Markup Language*

XLS – *Excel Spreadsheet*

Fig. - *Figura*

Ref - *Referência*

Dept - *Departamento*

Tab - *Tabela*

1

INTRODUÇÃO

1.1. MOTIVAÇÃO

A adoção do *Building Information Modeling* (BIM) tem transformado significativamente a indústria da construção, promovendo maior eficiência, colaboração e precisão no desenvolvimento de projetos. No entanto, para que esses benefícios sejam plenamente alcançados, a qualidade dos modelos BIM precisa ser assegurada ao longo de todo o ciclo de vida do projeto.

A verificação da qualidade do modelo BIM tornou-se uma necessidade para empresas tanto por razões internas, garantindo a confiabilidade e coerência das informações, quanto por exigências externas, com clientes impondo requisitos informacionais cada vez mais rigorosos. Atualmente, é comum que os contratantes demandem que os modelos atendam a uma série de critérios normativos, como preenchimento de classificações (*OmniClass*, *COBie*, etc.) e presença de múltiplos de parâmetros específicos por elemento. A ausência de um processo estruturado para garantir o cumprimento dessas exigências pode resultar em retrabalho, inconsistências e dificuldades na gestão da informação.

Além disso, os processos tradicionais de verificação manual de modelos BIM são morosos, sujeitos a erros e pouco escaláveis. O crescimento no volume de projetos e a necessidade de coordenação entre múltiplas disciplinas reforçam a urgência por soluções automatizadas que possibilitem a análise eficiente de múltiplos modelos simultaneamente.

Nesse contexto, esta pesquisa busca explorar metodologias e tecnologias que permitam aprimorar e/ou agilizar o processo de controle de qualidade dos modelos BIM. O estudo analisará ferramentas e fluxos de trabalho que possibilitem a verificação automatizada de parâmetros, a interoperabilidade entre softwares e a apresentação de informações de forma clara e estruturada para as equipes envolvidas — incluindo tanto soluções *open source* quanto ferramentas autorais.

A pesquisa terá como objetivo não apenas identificar boas práticas e ferramentas existentes, mas também propor um fluxo de verificação otimizado, considerando as necessidades específicas da empresa e a viabilidade de implementação dentro de um ambiente BIM. O estudo poderá contribuir para a padronização de processos, aumento da eficiência operacional e redução de falhas na modelagem e gestão da informação, facilitando a adoção de novas tecnologias e promovendo um uso mais estratégico do BIM nas organizações.

1.2. OBJETIVOS

A presente dissertação tem como principal objetivo propor um novo fluxo de trabalho para o controle de qualidade de modelos BIM, combinando as melhores práticas e ferramentas existentes para estruturar e operacionalizar a verificação automatizada da informação de forma clara, acessível e replicável. Embora a completa automatização da verificação de qualidade ainda não seja uma realidade, a crescente

evolução das tecnologias, incluindo inteligência artificial e técnicas avançadas de análise de modelos, tem permitido avanços significativos na otimização dessa tarefa.

Para atingir esse objetivo, será realizada uma investigação aprofundada das soluções disponíveis no mercado, analisando suas capacidades, limitações e aplicabilidade prática. Com base nesse levantamento, pretende-se estruturar uma abordagem que integre diferentes métodos e ferramentas, otimizando o fluxo de verificação de qualidade de maneira escalável e adaptável à realidade da empresa estudada.

Além disso, busca-se desenvolver um processo de validação progressivo, onde cada nova aplicação do método contribua para seu aprimoramento contínuo. Em vez de buscar uma solução definitiva e estática, esta pesquisa pretende explorar um modelo dinâmico de otimização, capaz de evoluir a cada projeto e se tornar mais eficiente com o tempo.

Por fim, a dissertação pretende contribuir para o avanço da automação da verificação de qualidade em BIM, fornecendo insights sobre as barreiras ainda existentes e propondo alternativas que possam tornar esse processo cada vez mais eficiente e integrado às rotinas das empresas. O objetivo não é apenas melhorar a qualidade dos modelos no presente, mas também lançar as bases para futuras inovações, auxiliando na construção de um cenário onde a automação da verificação BIM seja uma realidade cada vez mais acessível e eficaz.

1.3. ESTRUTURA DA DISSERTAÇÃO

A presente dissertação está organizada em seis capítulos. O Capítulo 1 corresponde à introdução, onde se apresenta o contexto da pesquisa, a motivação para a sua realização, os objetivos definidos e a estrutura geral do documento.

O Capítulo 2 é dedicado ao Estado da Arte e tem como objetivo fundamentar teoricamente o trabalho, oferecendo uma visão abrangente sobre os principais temas relacionados à qualidade informacional no contexto do *Building Information Modeling* (BIM). São explorados os conceitos de *Quality Assurance* e *Quality Control* aplicados a modelos digitais, bem como a evolução das abordagens para verificação da informação ao longo das últimas décadas. A revisão bibliográfica inclui ainda o estudo de normas técnicas, metodologias e soluções tecnológicas utilizadas na prática, com ênfase nas abordagens automatizadas de verificação e nos fluxos baseados em padrões abertos, como o *Industry Foundation Classes* (IFC) e o *Information Delivery Specification* (IDS). Iniciativas institucionais, ferramentas consolidadas do mercado e soluções emergentes — como as baseadas em inteligência artificial — são analisadas de forma crítica, identificando seus potenciais, limitações e áreas de aplicação. Ao final do capítulo, apresenta-se uma síntese dos principais desafios enfrentados pela indústria e as lacunas que justificam o desenvolvimento de um novo fluxo de trabalho.

O Capítulo 3 apresenta a metodologia adotada para o desenvolvimento e validação da proposta. Inicialmente, descreve-se o racional por trás da escolha do IDS como elemento central do fluxo de verificação, bem como os critérios considerados na seleção das ferramentas utilizadas. Em seguida, são detalhadas as etapas que compõem o fluxo proposto, desde a definição dos requisitos informacionais até a verificação automática e o tratamento das não conformidades. O capítulo inclui também a descrição das ferramentas utilizadas em cada etapa — nomeadamente o *usBIM IDS Editor*, o Blender com o plugin Bonsai e o BIMcollab Zoom —, bem como as interações entre elas no contexto de um ambiente colaborativo baseado em *OpenBIM*. Além disso, discute-se a lógica subjacente ao fluxo, com foco na escalabilidade, clareza dos resultados e adaptação às necessidades específicas da empresa parceira. A

metodologia contempla ainda os critérios de validação, que foram definidos com base nos requisitos do projeto real utilizado como estudo de caso.

O Capítulo 4 apresenta o estudo de caso, desenvolvido em colaboração com uma empresa do setor da construção. São descritos os modelos fornecidos, as disciplinas envolvidas e os requisitos informacionais definidos para cada especialidade. A aplicação do fluxo de verificação é documentada em detalhe, incluindo a preparação dos ficheiros, a execução das verificações e o processo de comunicação e gestão das não conformidades. O capítulo busca evidenciar o funcionamento do fluxo proposto em um cenário prático, refletindo as particularidades, obstáculos e adaptações exigidas pela realidade do projeto.

O Capítulo 5 é dedicado à análise crítica dos resultados obtidos. São avaliados os impactos da abordagem proposta em termos de eficiência, clareza, reprodutibilidade e alinhamento com os objetivos de qualidade da empresa. A análise inclui ainda uma discussão sobre as limitações identificadas durante a aplicação do método, bem como sugestões de melhoria e potencial de generalização para outros projetos e organizações. O capítulo também explora o papel da interoperabilidade e dos padrões abertos como facilitadores da automação e da padronização na verificação de modelos BIM.

Por fim, o Capítulo 6 apresenta as conclusões da dissertação, sintetizando os principais contributos da pesquisa e propondo caminhos para estudos futuros. Reflete-se sobre a relevância dos resultados obtidos, a aplicabilidade da abordagem em contextos reais e as implicações para o avanço das práticas de verificação automatizada no setor. Ao final do documento, é disponibilizada uma seção de anexos contendo os principais ficheiros, documentos e configurações utilizadas ao longo do estudo, de modo a assegurar a transparência e a reprodutibilidade da proposta apresentada.

2

ESTADO DA ARTE

2.1. QUALITY ASSURANCE (QA) E QUALITY CONTROL (QC) NO CONTEXTO BIM

2.1.1. DEFINIÇÃO DE QUALIDADE E SUA APLICAÇÃO NA INDÚSTRIA AEC

O conceito de qualidade tem sido amplamente estudado e evoluiu ao longo do tempo, sendo interpretado de maneiras diferentes dependendo do contexto em que é aplicado. Na indústria da Arquitetura, Engenharia e Construção (AEC), o termo "qualidade" assume uma complexidade adicional, pois envolve múltiplos fatores, como processos regulatórios, requisitos do cliente e desempenho dos projetos ao longo do seu ciclo de vida (Arditi & Gunaydin, 1998). Diferentemente de um processo industrial, onde a qualidade pode ser assegurada por meio de testes e inspeções antes da entrega do produto final, na construção civil cada erro durante a execução da obra acarreta custos significativos e impactos diretos no cronograma e na eficiência da edificação (O'Brien, 1989).

Nesse contexto, torna-se evidente a dificuldade em definir o que é qualidade na construção civil e, mais ainda, como aplicar um controle e garantia eficazes desse conceito. A literatura do tema aponta algumas formas de abordar essa complexidade, destacando elementos que podem ser avaliados separadamente dentro do conceito de qualidade (Chen & Luo, 2014).

Um dos principais eixos contemporâneos dessa discussão é o *Building Information Modeling*, uma metodologia que integra processos, pessoas e tecnologias com o objetivo de maximizar o desempenho e a eficiência de um edifício ao longo de todo o seu ciclo de vida — do projeto à operação e eventual demolição.

Trata-se de uma abordagem baseada em dados e informações digitais, e não apenas em representações geométricas tridimensionais do produto final. O modelo BIM, portanto, é uma representação digital do edifício, enriquecida com informações paramétricas, que acompanha a edificação real ao longo de sua existência, servindo como base para decisões mais bem informadas e precisas em todas as fases do empreendimento.

Naturalmente, a análise da qualidade nesse contexto ultrapassa os aspectos puramente geométricos, envolvendo também os dados e parâmetros que caracterizam cada elemento do modelo. Com a adoção do BIM, novas oportunidades e desafios surgiram na área do controle e garantia da qualidade. O BIM não apenas permite reduzir erros de projeto e minimizar retrabalho (Lee et al., 2015), como também viabiliza a detecção automatizada de inconsistências nos modelos digitais. Quando implementado desde as fases iniciais, favorece um fluxo de trabalho mais integrado, resultando em projetos de maior qualidade, com custos reduzidos e prazos de entrega menores (Eastman et al., 2011).

A relação entre o momento da tomada de decisão e os custos associados às alterações no projeto pode ser ilustrada pela Curva de MacLeamy (Figura 1), que demonstra que a capacidade de influenciar

positivamente o desempenho do edifício é significativamente maior nas fases iniciais do ciclo de vida. Por outro lado, os custos e impactos, das mudanças aumentam de forma exponencial à medida que o projeto avança.

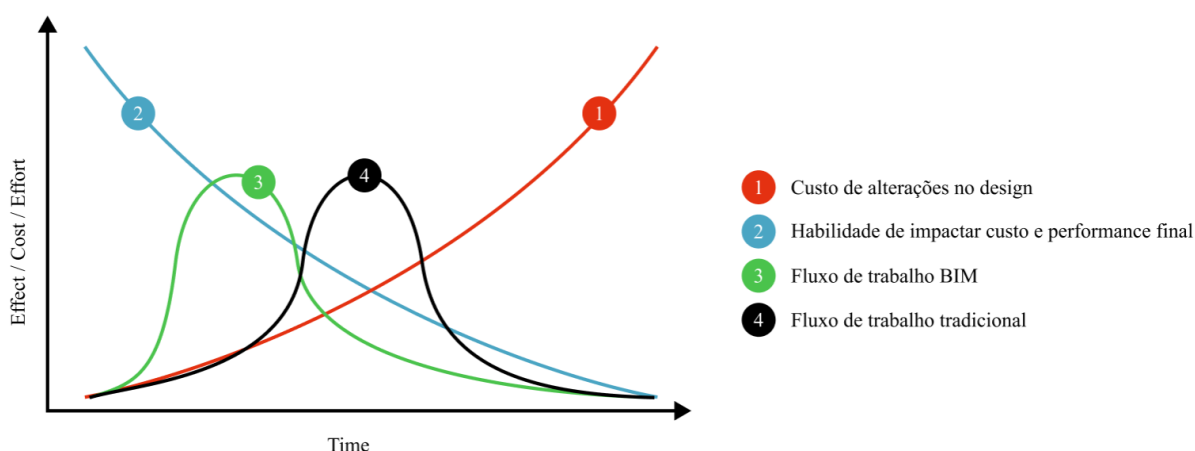


Figura 1 - Curva de Macleamy (Lu et al., 2014)

De acordo com a ISO 19650-1:2018 ((ISO), 2018), todas as partes envolvidas na geração de informações são responsáveis pela sua qualidade, devendo estabelecer planos e processos que garantam que os modelos sejam precisos, consistentes e completos. Com a crescente adoção do BIM, novos processos, tecnologias e atores passaram a demandar a criação de normativas específicas, capazes de assegurar a padronização e confiabilidade dessas informações ao longo do ciclo de vida do projeto.

A literatura especializada propõe que a qualidade no BIM pode ser analisada sob duas perspectivas principais (Donato et al., 2018):

- I. Qualidade dos processos BIM, que envolve a gestão da troca de informações, a colaboração entre *stakeholders* e a aderência aos protocolos estabelecidos para a modelação e gestão de dados. Este aspeto está relacionado à definição de métodos, responsabilidades e fluxos de trabalho;
- II. Qualidade dos modelos BIM, que se refere à precisão, coerência e completude das informações contidas nos modelos digitais. Um modelo de qualidade deve garantir que todos os elementos possuam informações corretas, estejam posicionados adequadamente e atendam aos requisitos do projeto.

2.1.2. CONCEITOS DE QUALITY ASSURANCE (QA) E QUALITY CONTROL (QC)

Antes de aprofundar a análise da qualidade em modelos BIM, é fundamental compreender dois conceitos centrais amplamente utilizados no contexto da gestão da qualidade: *Quality Assurance* (QA), ou garantia da qualidade, e *Quality Control* (QC), ou controlo da qualidade. Esses termos são frequentemente confundidos, mas representam abordagens distintas e complementares no processo de avaliação da qualidade (Salaberri Martin, 2022).

A literatura especializada frequentemente estrutura a discussão sobre qualidade em torno de dois eixos: *Quality Assurance* (QA) e *Quality Control* (QC). O QA refere-se a um programa que engloba as atividades necessárias para garantir a qualidade do trabalho em conformidade com os requisitos do

projeto, enquanto o QC diz respeito à implementação prática dessas atividades de verificação (Arditi & Gunaydin, 1998; Donato et al., 2018; Ferguson & Cleyton, 1988).

O QA pode ser entendido como um processo preventivo, voltado à definição de diretrizes, padrões e procedimentos que assegurem o cumprimento dos requisitos de qualidade antes mesmo do início da modelagem ou da execução. Trata-se de um sistema implementado para fornecer confiança de que os requisitos de qualidade de um produto ou serviço serão atendidos. No contexto BIM, isso se traduz na criação de planos de modelagem, protocolos de desenvolvimento, definição de requisitos de informação e processos de validação — elementos fundamentais previstos na norma ISO 19650, que orienta a padronização e a consistência das informações geradas ((ISO), 2018; BIM Dictionary, 2025).

O QC, por sua vez, caracteriza-se como um processo reativo, centrado na verificação da conformidade dos produtos entregues com os requisitos previamente definidos. Envolve atividades e técnicas destinadas a avaliar se um produto ou serviço atende aos critérios de qualidade estabelecidos. No contexto do BIM, essas atividades incluem checagens manuais ou automatizadas dos modelos, validação de parâmetros informacionais, verificação de conformidade normativa e identificação de conflitos geométricos (clash detection) (BIM Dictionary, 2025; Chen & Luo, 2014).

Embora distintos, QA e QC são conceitos complementares no contexto da gestão da qualidade em BIM. O primeiro estabelece uma base sólida de procedimentos e requisitos que orientam o desenvolvimento do modelo, enquanto o segundo garante que essas diretrizes foram devidamente seguidas e que o produto final cumpre os padrões exigidos (Mirarchi & Pavan, 2019). A integração eficiente entre QA e QC é essencial para assegurar modelos digitais precisos, coerentes e completos, contribuindo para a redução de retrabalho e para a melhoria do desempenho geral dos projetos.

2.1.3. MÉTODOS EXISTENTES DE MENSURAÇÃO DE QUALIDADE NOS MODELOS E PROCESSOS BIM

Alguns métodos vêm sendo propostos na literatura para mensuração da qualidade no contexto do BIM, tanto em relação aos modelos quanto aos processos envolvidos na sua produção e gestão. Um dos métodos mais referenciados é o modelo de BIM *Performance Measurement*, que organiza a análise da performance em cinco componentes principais (Succar et al., 2012):

- *BIM Capability Stages*: identifica o nível de adoção do BIM dentro da organização;
- *BIM Maturity Levels*: avalia a progressão na implementação do BIM ao longo do tempo;
- *BIM Competency Sets*: analisa as competências necessárias para o desenvolvimento dos modelos;
- *Organisational Scales*: observa como a aplicação do BIM varia em diferentes escalas organizacionais;
- *Granularity Levels*: define o nível de detalhe das informações contidas nos modelos.

No que diz respeito especificamente à avaliação da qualidade dos dados inseridos nos modelos, destacam-se quatro dimensões principais: completude, consistência, precisão e atualidade (Mirarchi & Pavan, 2019):

- *Accuracy* (Precisão): grau de correção e confiabilidade das informações;
- *Consistency* (Consistência): verificação do cumprimento das regras semânticas definidas;
- *Completeness* (Completude): análise da cobertura dos dados necessários à representação de um objeto ou sistema;
- *Timeliness* (Atualização): verificação da atualidade das informações em relação ao estado real do projeto.

Essas abordagens não se excluem e podem ser utilizadas de forma complementar, dependendo do foco da análise. Em conjunto, contribuem para que a avaliação da qualidade em BIM seja conduzida de forma mais estruturada, com base em critérios verificáveis ao longo das diferentes fases do projeto.

2.1.4. DESAFIOS E LIMITAÇÕES ATUAIS

Apesar dos avanços significativos proporcionados pelo uso do BIM na gestão da qualidade dos projetos, a verificação da qualidade dos modelos ainda enfrenta diversas limitações. Mesmo com a crescente adoção da metodologia, observa-se uma escassez de pesquisas voltadas especificamente à verificação da qualidade e ao uso eficiente das informações geradas. Isso indica que muitas práticas de controle permanecem pouco sistematizadas ou ainda carecem de padronização efetiva (Donato et al., 2018).

Entre os principais entraves está a dificuldade de interoperabilidade entre diferentes softwares utilizados no ambiente BIM, o que compromete a consistência dos dados e dos modelos. Soma-se a isso a existência de variações nos formatos de exportação, como o *Industry Foundation Class* (IFC), que podem afetar a integridade da informação durante a troca entre plataformas. As ferramentas de verificação automática, embora úteis, ainda apresentam limitações quanto à abrangência e à precisão na detecção de inconsistências.

Além dos aspectos técnicos, há também obstáculos relacionados à organização dos processos de verificação. A ausência de critérios de qualidade uniformes e a falta de definição clara nos fluxos de trabalho comprometem a aplicação consistente de práticas de QA/QC. Em muitos contextos, ainda há resistência à adoção de procedimentos automatizados, o que pode dificultar a implementação de rotinas mais eficientes e seguras.

Outro desafio importante reside na dimensão humana. A curva de aprendizado associada ao BIM é elevada, e muitos profissionais ainda não receberam capacitação adequada em metodologias específicas de garantia e controle da qualidade. Isso aumenta a probabilidade de erros manuais durante a modelagem e a inserção de dados, o que, por sua vez, compromete a confiabilidade dos modelos gerados.

Além disso, a falta de clareza conceitual entre os termos *Quality Assurance* e *Quality Control* contribui para confusões na prática profissional. Esses conceitos são frequentemente utilizados de forma intercambiável, o que pode levar a abordagens inadequadas de verificação e, consequentemente, a modelos menos robustos e confiáveis (Salaberri Martin, 2022).

Superar essas barreiras requer investimentos contínuos em normalização de processos, no aprimoramento das ferramentas digitais e, sobretudo, na capacitação dos profissionais envolvidos. A consolidação de uma cultura de qualidade, aliada à sistematização de boas práticas, é fundamental para fortalecer a confiabilidade e a eficiência dos projetos desenvolvidos com o uso do BIM.

2.2. O “I” DO BIM E SEUS REQUISITOS

Diferentemente das metodologias tradicionais que precederam o BIM, esta abordagem tem como núcleo um elemento essencial já destacado em seu próprio nome: a informação. O “I” do *Building Information Modeling* representa justamente a ênfase na gestão e estruturação da informação que, quando devidamente organizada, passa a desempenhar um papel central em todas as fases do ciclo de vida do empreendimento.

A gestão eficiente dessa informação é, portanto, um dos pilares do BIM. É ela que permite que os modelos digitais deixem de ser apenas representações geométricas e se tornem repositórios confiáveis

de dados, utilizados desde as etapas iniciais de concepção até a operação e manutenção da edificação. Para isso, é fundamental que as informações estejam bem definidas, organizadas de forma consistente e passíveis de verificação objetiva. Só assim os modelos poderão atender às necessidades específicas de cada fase do projeto, contribuindo para decisões mais precisas e articuladas.

Nesse contexto, ganha relevância um conceito fundamental para a análise qualitativa dos modelos: os requisitos de informação, ou *Information Requirements* (IR).

2.2.1. REQUISITOS DE INFORMAÇÃO E SEUS MÉTODOS DE ESPECIFICAÇÃO

Os Requisitos de Informação (*Information Requirements*) definem os dados que precisam ser entregues ao longo do ciclo de vida do projeto, estabelecendo o que é necessário para garantir a conformidade e a qualidade dos modelos BIM. Conforme estabelecido na ISO 19650 ((ISO), 2018), esses requisitos devem ser definidos de forma estratégica pela parte contratante, com o objetivo de assegurar que a informação seja relevante, útil e apropriada à fase em que será utilizada.

Paralelamente, os Métodos de Especificação de Requisitos determinam como essas informações devem ser estruturadas, organizadas e verificadas para garantir que atendam aos requisitos definidos. Esses métodos são essenciais para viabilizar a padronização, a interoperabilidade entre plataformas e a automação na verificação de conformidade dos modelos (Tomczak et al., 2022).

A ISO 19650 ((ISO), 2018) propõe a utilização de quatro documentos fundamentais para estruturar os Requisitos de Informação:

- *Organizational Information Requirements* (OIR): informações necessárias para apoiar os objetivos estratégicos da organização;
- *Asset Information Requirements* (AIR): dados necessários à operação e manutenção do ativo, em resposta ao OIR;
- *Project Information Requirements* (PIR): informações essenciais para o desenvolvimento do projeto, abrangendo as fases de concepção, construção e entrega;
- *Exchange Information Requirements* (EIR): requisitos definidos contratualmente para cada entrega, baseados no AIR e no PIR. Incluem critérios técnicos (ex: softwares a serem utilizados), gerenciais (ex: fluxos de trabalho) e comerciais (ex: prazos de entrega).

Esses requisitos devem ser definidos com base nas fases do projeto quando se referem ao desenvolvimento do Project Information Model (PIM). Já no caso da fase operacional do ativo, que corresponde ao desenvolvimento do Asset Information Model (AIM), os requisitos devem ser expressos por meio de eventos-gatilho ao longo do ciclo de vida. A figura a seguir ilustra esse processo.

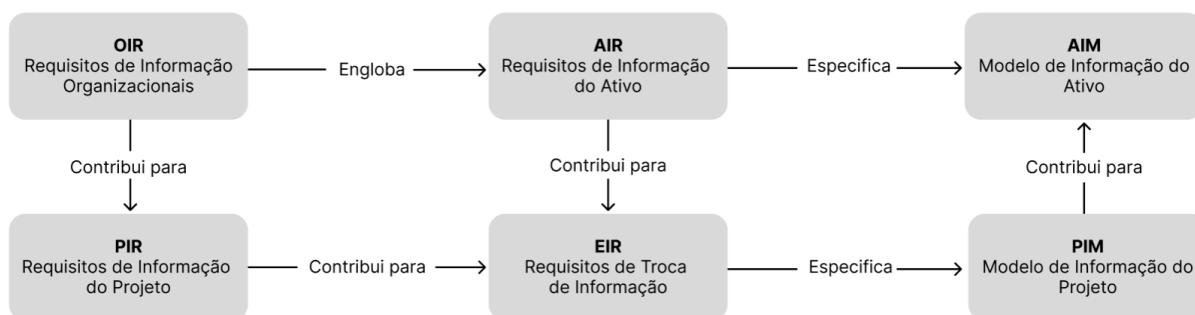


Figura 2 - Métodos de especificação de informação

Diversos métodos e padrões têm sido desenvolvidos para garantir que os requisitos de informação sejam corretamente estruturados, interpretados e validados. Uma análise comparativa dessas abordagens, avaliando-as segundo 19 critérios agrupados em sete categorias — padronização, aplicabilidade, campos de informação, restrições de valor, representação geométrica, metadados e interoperabilidade — pode ser encontrada em (Tomczak et al., 2022). Entre os métodos analisados, destacam-se:

- Product Data Templates (PDT)
- Model View Definition (mvdXML)
- Information Delivery Manual (IDM)
- Level of Information Need (LOIN)
- Dicionários de Dados (ISO 12006)
- Templates de Propriedades IFC
- Information Delivery Specification (IDS)
- Linked Data com SHACL
- Documentos Textuais ou Planilhas Não Padronizadas (DOC & XLS)
- Soluções Proprietárias (por exemplo, Solibri)
- Scripts de Programação Visual Dedicados

A análise conclui que nenhuma das abordagens atende integralmente a todos os critérios de avaliação. Ainda assim, o IDS é apontado como a solução mais vantajosa para a verificação automatizada de conformidade, por permitir a validação de requisitos de informação alfanuméricos.

Este método será analisado com maior profundidade adiante, mas antes é necessário compreender o raciocínio *open-source* e a lógica de padronização que sustentam sua estrutura.

2.3. INTEROPERABILIDADE, OPENBIM E A IMPORTÂNCIA DOS PADRÕES ABERTOS

Como referido anteriormente, um dos principais entraves da indústria é a dificuldade de interoperabilidade entre diferentes softwares utilizados no ambiente BIM, sobretudo em contextos colaborativos que envolvem múltiplas disciplinas, ferramentas e fases de projeto. Em termos simples, interoperabilidade refere-se à capacidade de diferentes sistemas trocarem informações de forma eficiente, confiável e sem perdas relevantes de dados.

Do ponto de vista dos utilizadores, como projetistas, empreiteiros e entidades públicas, a possibilidade de trocar informações entre sistemas distintos representa um ganho direto em eficiência, autonomia e sustentabilidade informacional. No entanto, no contexto das soluções comerciais, a interoperabilidade impõe também desafios técnicos e estratégicos aos fornecedores de software.

A ausência dessa interoperabilidade entre plataformas, muitas vezes referida na literatura como *vendor lock-in*, tem sido historicamente observada em setores intensivos em tecnologia. Embora soluções proprietárias ofereçam alto desempenho em tarefas específicas, seus formatos fechados e estruturas exclusivas dificultam a portabilidade das informações e limitam o reaproveitamento dos modelos em outros ambientes digitais. Essa limitação compromete a flexibilidade dos utilizadores e aumenta sua dependência de fornecedores específicos, o que representa um risco tecnológico significativo, especialmente em setores como o da construção, cujos ativos devem ser mantidos por décadas.

Uma analogia útil para ilustrar a importância da interoperabilidade é a recente decisão da União Europeia (UE) de padronizar os carregadores de dispositivos eletrônicos. A partir de 2024, todos os novos smartphones, tablets e máquinas fotográficas comercializados na UE deverão adotar a porta USB-C, independentemente do fabricante. A medida visa reduzir a fragmentação causada por padrões proprietários, diminuir o desperdício eletrônico e simplificar a experiência do consumidor. De forma análoga, a adoção de padrões abertos na construção digital, como o *Industry Foundation Classes*, permite que diferentes agentes compartilhem informações com maior transparência e durabilidade, promovendo inovação e concorrência em condições mais equitativas.

Nesse contexto, surge o conceito de *OpenBIM*, promovido desde 1995 pela *buildingSMART International* (buildingSMART, 2025a). O *OpenBIM* defende o uso de padrões abertos para representar, compartilhar e validar dados da construção de forma neutra em relação às ferramentas utilizadas.

O formato IFC, em particular, consolidou-se como o principal padrão aberto para representação semântica de modelos BIM. Trata-se de uma especificação desenvolvida de forma colaborativa e mantida por entidades independentes. Embora seja considerado um formato *lossy*, ou seja, com potencial de perda de informação na conversão a partir de modelos nativos, ainda é a forma mais eficiente e durável de garantir a integridade mínima dos dados ao longo do tempo. Em contrapartida, os formatos nativos são frequentemente inacessíveis sem licenças específicas, carecem de documentação pública detalhada e estão sujeitos a alterações unilaterais por parte dos desenvolvedores.

A tendência regulatória em diversos países tem reforçado a importância do IFC como formato obrigatório em projetos públicos. China, Dinamarca, Japão, Polónia e Espanha, por exemplo, já estabelecem diretrizes ou legislações que exigem sua adoção, com o objetivo de garantir transparência, auditabilidade e controle de qualidade nas obras financiadas pelo setor público (buildingSMART, 2025b). Esse movimento pressiona empresas e profissionais a adotarem ferramentas compatíveis com os princípios do *OpenBIM*.

Nesse ecossistema, o uso de ferramentas *open source*, como Blender (com o *add-on* Bonsai), IfcOpenShell, xBIM Toolkit e BIMServer — fortalece a independência tecnológica e permite maior controle sobre os fluxos de verificação da qualidade. Diferente de soluções *freeware* ou com licença restrita, ferramentas abertas podem ser auditadas, personalizadas e adaptadas às necessidades específicas de cada projeto. Isso possibilita o desenvolvimento de fluxos de verificação próprios, sem dependência direta de fornecedores ou de atualizações externas imprevisíveis.

É importante reconhecer, no entanto, que essas ferramentas apresentam limitações. Por serem frequentemente desenvolvidas por comunidades voluntárias ou grupos acadêmicos, tendem a ter um

nível de maturidade inferior em comparação com soluções comerciais consolidadas. Isso pode se refletir em interfaces menos intuitivas, fluxos mais complexos e funcionalidades inconsistentes entre versões.

Além disso, a ausência de suporte técnico estruturado representa um obstáculo relevante para empresas que operam sob prazos rigorosos e exigem estabilidade operacional. Eventuais falhas ou incompatibilidades podem não ter resolução imediata, dependendo de fóruns públicos, *issues* em plataformas como GitHub ou da disponibilidade pontual dos colaboradores do projeto.

Outro desafio recorrente é a exigência de conhecimentos técnicos mais avançados por parte dos utilizadores. Instalar, configurar e adaptar essas soluções pode demandar familiaridade com ambientes de desenvolvimento, linhas de comando ou integração por meio de *Application Programming Interfaces* (APIs), competências nem sempre presentes nas equipes de engenharia e arquitetura. Isso pode resultar em uma curva de aprendizado mais acentuada e exigir investimentos adicionais em capacitação ou suporte especializado.

Apesar dessas limitações, a adoção de ferramentas abertas tem crescido de forma constante, especialmente em ambientes acadêmicos, instituições públicas e empresas que valorizam transparência, auditabilidade e autonomia digital. A possibilidade de adaptar o código-fonte às necessidades específicas de um projeto ou integrá-lo com outras soluções por meio de APIs constitui uma vantagem estratégica importante, sobretudo em contextos que exigem flexibilidade e rastreabilidade.

Nesse sentido, embora ainda enfrentem desafios de usabilidade e maturidade, as soluções *open source* representam um campo promissor, alinhado com uma tendência global de abertura, colaboração e autonomia tecnológica no setor da construção.

2.4. VERIFICAÇÃO AUTOMÁTICA DE CONFORMIDADE DE MODELOS BIM

Tendo em vista as dificuldades associadas à garantia da qualidade nos modelos BIM e à diversidade de métodos disponíveis para a verificação de requisitos de informação, torna-se pertinente abordar um dos temas centrais desta dissertação: a automação do processo de verificação de conformidade.

2.4.1. INTRODUÇÃO À VERIFICAÇÃO AUTOMÁTICA DE CONFORMIDADE

A verificação automática de conformidade, também conhecida como *Automated Compliance Checking* (ACC), consiste na utilização de regras predefinidas e algoritmos computacionais para avaliar se um modelo BIM atende a requisitos normativos, regulatórios e de projeto. Essa abordagem visa reduzir erros, aumentar a confiabilidade das informações e melhorar a eficiência dos processos de coordenação e aprovação de projetos no setor da Arquitetura, Engenharia e Construção (Marco et al., 2024).

A verificação automática não modifica um projeto, mas avalia sua conformidade com base na configuração dos objetos, suas relações e atributos. O objetivo é facilitar a detecção de não conformidades desde as fases iniciais do projeto, minimizando a necessidade de correções tardias e retrabalho (Eastman et al., 2009).

O desenvolvimento da verificação automática de conformidade tem suas raízes nos anos 70, quando os chamados sistemas especialistas dominavam as pesquisas da área. Esses sistemas utilizavam conhecimento codificado manualmente por especialistas para simular decisões de projeto, com estruturas baseadas em regras lógicas do tipo *if-then*. Apesar de pioneiros, apresentavam limitações consideráveis em termos de escalabilidade, interoperabilidade e manutenção, sobretudo quando aplicados a contextos normativos mais complexos e variáveis.

A virada conceitual ocorreu com a introdução do formato IFC, posteriormente padronizado pela ISO 16739:2013, e com a consolidação da filosofia *OpenBIM*. Essa nova abordagem priorizou a padronização dos modelos e a interoperabilidade entre ferramentas, o que permitiu que a ACC deixasse de depender de estruturas fechadas e proprietárias, abrindo espaço para fluxos de verificação mais flexíveis, auditáveis e amplamente reutilizáveis.

Nesse novo cenário, diversas soluções passaram a explorar o potencial da integração entre BIM e verificação normativa automática. Destacam-se ferramentas como DesignCheck, Solibri Model Checker, Fornax, SMARTCodes, Jotne EDM, e CORENET, entre outras. Esses sistemas incorporaram mecanismos de validação de modelos com base em regras, aplicadas diretamente sobre dados estruturados em IFC, como ilustrado na Figura 3, que apresenta uma linha temporal do desenvolvimento desses sistemas e evidencia a mudança de paradigma impulsionada pelo OpenBIM.

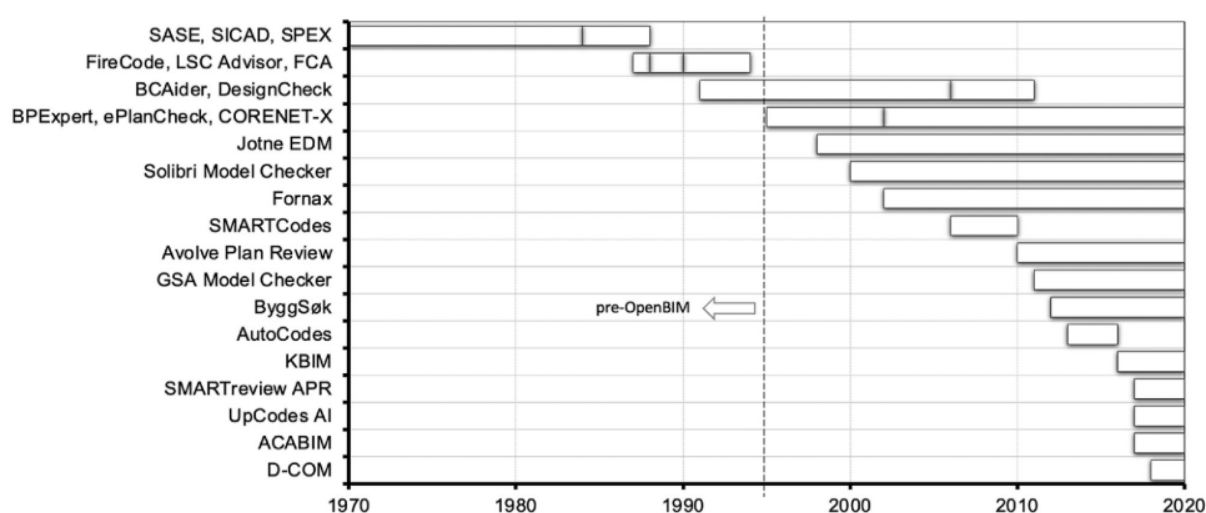


Figura 3 - Linha do tempo dos principais sistemas automáticos de verificação de conformidade (Amor & Dimyadi, 2021).

O OpenBIM tem desempenhado um papel essencial na criação de fluxos de verificação mais integrados e interoperáveis, ao possibilitar o uso articulado de diversos componentes do seu ecossistema (Amor & Dimyadi, 2021):

- IFC como formato comum de intercâmbio de dados entre softwares;
- IDM e MVDs, que permitem estruturar os requisitos de informação e os momentos de entrega;
- Tecnologias semânticas como OWL, RDF, SPARQL e SHACL, que ampliam a capacidade de raciocínio automático sobre os modelos;
- bSDD (*buildingSMART Data Dictionary*), que conecta os termos utilizados no modelo aos termos presentes nos regulamentos normativos, promovendo alinhamento semântico.

Retomando os conceitos anteriormente apresentados sobre *Quality Control* e *Quality Assurance*, a verificação automática, por si só, configura-se como uma atividade de controle de qualidade, ao avaliar resultados pontuais do modelo. No entanto, quando inserida de forma sistemática ao longo do projeto, por meio de rotinas estruturadas, ela também contribui para a garantia da qualidade, ao estabelecer procedimentos consistentes de verificação.

A importância da verificação automática está diretamente relacionada ao crescimento da adoção do BIM e à complexidade dos regulamentos e padrões de modelagem, que variam entre países e setores. A automação desse processo permite:

- I. Redução de erros humanos na verificação manual de modelos BIM.
- II. Maior eficiência e rapidez na análise da conformidade dos projetos.
- III. Melhoria na qualidade da informação presente nos modelos.
- IV. Padronização da verificação de normas e regulamentos em diferentes tipos de projetos.

2.4.2. ESTÁGIOS DA VERIFICAÇÃO AUTOMÁTICA

Ao longo da revisão da literatura, nota-se a consolidação de uma estrutura composta por quatro estágios fundamentais que orientam o processo de verificação automática de modelos BIM. Essa divisão tem sido amplamente utilizada por diferentes autores como forma de organizar um procedimento complexo em etapas compreensíveis e aplicáveis, conforme originalmente proposto por Eastman (Eastman et al., 2009).

O primeiro estágio corresponde à interpretação e estruturação lógica das regras. Consiste na conversão de normas e requisitos, geralmente redigidos em linguagem natural, para uma lógica formal que possa ser processada por sistemas computacionais. As regras codificadas precisam ser compatíveis com os objetos, propriedades e relações existentes nos modelos BIM.

A segunda etapa refere-se à preparação do modelo. Nessa fase, é necessário garantir que o modelo contenha as informações exigidas pelas regras e esteja estruturado de acordo com os padrões esperados pelos sistemas de verificação. Isso pode incluir a extração e organização de dados, a inserção manual de informações por parte dos projetistas ou a derivação automática de atributos a partir do próprio modelo.

O terceiro estágio envolve a execução das regras. A partir da aplicação das lógicas codificadas sobre o modelo preparado, o sistema realiza a verificação e identifica os elementos que estão ou não em conformidade com os critérios definidos.

Por fim, a quarta etapa diz respeito à geração do relatório de verificação. Os resultados da análise são organizados e apresentados, permitindo a identificação das inconformidades encontradas e orientando os ajustes necessários no projeto.

Durante essas etapas, diferentes estratégias podem ser adotadas para garantir a disponibilidade das informações exigidas, como a inserção explícita de dados no modelo, o uso de inferência computacional para deduzir atributos implícitos ou a aplicação de simulações para estimar parâmetros mais complexos, como desempenho estrutural ou energético. A eficácia do processo depende do equilíbrio entre a redução da carga documental para os projetistas e a ampliação das capacidades interpretativas dos sistemas (Solihin et al., 2015).

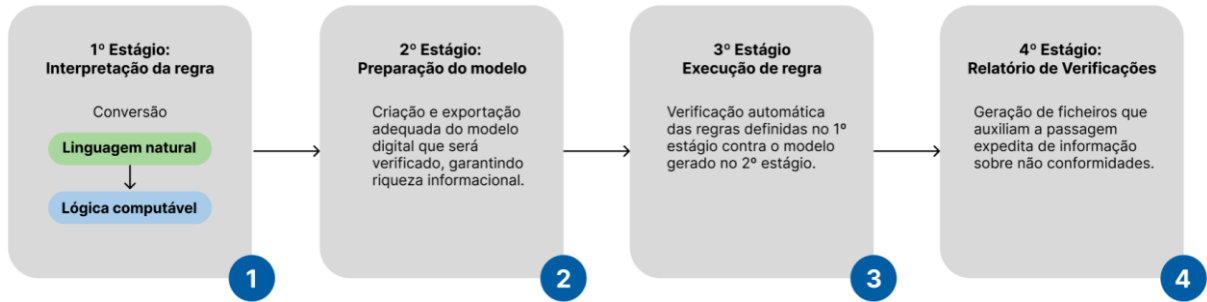


Figura 4 - Estágios de Eastman (Eastman et al., 2009)

2.4.2.1. Interpretação da regra

A interpretação de regras é a conversão de normas e regulamentos escritos em linguagem humana para um formato que possa ser processado por um sistema automatizado. Isso envolve a tradução das regras em uma lógica computacional clara e estruturada.

Principais desafios:

- Ambiguidade das normas: Regulamentos podem ser escritos de forma vaga ou interpretável de diferentes maneiras, dificultando a tradução direta para código.
- Falta de padronização: Os requisitos variam entre países, órgãos reguladores e tipos de projetos, o que exige adaptações manuais.
- Complexidade lógica: Algumas regras envolvem relações condicionais complexas, tornando difícil sua formalização em código.
- Ontologias e terminologias diferentes: Modelos BIM podem conter diferentes nomenclaturas para os mesmos elementos, dificultando a aplicação universal das regras.

Diversas abordagens têm sido exploradas para a verificação automática de conformidade na indústria da construção, incluindo métodos baseados em IFC, padrões de dados como JSON *Schema* e XSD, e técnicas de *Linked Data* como OWL, SWRL, SPARQL e SHACL. A interpretação automática de regras enfrenta desafios significativos, principalmente pela necessidade de traduzir normas em linguagem natural para formatos processáveis por máquina. A padronização surge, nesse contexto, como fator essencial para alinhar requisitos normativos às informações contidas nos modelos BIM (Nuyts et al., 2024).

2.4.2.2. Preparação do modelo

Antes de executar a verificação automática, o modelo BIM deve ser preparado para garantir que contenha todas as informações necessárias e esteja estruturado corretamente para que as regras possam ser aplicadas. Dentre os principais desafios, estão:

- Inconsistência na modelagem: Modelos BIM são frequentemente criados sem seguir padrões rígidos, o que pode dificultar a extração de informações para a verificação.
- Falta de dados necessários: Alguns elementos essenciais para a análise podem estar ausentes no modelo, seja por erro humano ou limitações do software utilizado.
- Conversão para formatos compatíveis: A exportação para IFC pode resultar na perda ou modificação de informações.

- Geração de vistas específicas: Alguns métodos de verificação exigem que o modelo seja segmentado ou que vistas específicas sejam criadas para análise.

2.4.2.3. Execução de regras

Este estágio envolve a aplicação das regras definidas ao modelo BIM preparado. O sistema processa os dados, verifica a conformidade com os requisitos e gera resultados baseados nas regras estabelecidas.

Principais desafios:

- Volume de informações: Modelos BIM podem ser extremamente grandes e detalhados, exigindo alto poder computacional para verificação eficiente.
- Interoperabilidade limitada: Nem todas as ferramentas de verificação conseguem processar modelos criados em diferentes softwares BIM sem perda de dados.
- Erros de validação sintática: Alguns modelos contêm erros de estruturação que impedem a execução correta das regras.
- Regras contraditórias ou sobrepostas: Certos requisitos podem entrar em conflito uns com os outros, dificultando a priorização e interpretação das regras.

2.4.2.4. Reportagem de regras

Após a execução das regras, os resultados precisam ser apresentados de forma clara e compreensível para os usuários, destacando as não conformidades e fornecendo insights sobre possíveis correções.

Principais desafios:

- Falsos positivos e negativos: Erros na definição das regras ou falhas na modelagem podem gerar resultados incorretos, dificultando a confiabilidade do sistema.
- Dificuldade na interpretação dos relatórios: Usuários nem sempre conseguem entender facilmente os problemas apontados pelo sistema, especialmente quando há muitas regras em análise simultânea.
- Falta de rastreabilidade: É essencial que o relatório consiga relacionar cada erro encontrado à regra correspondente e ao elemento do modelo que causou a não conformidade.
- Formato inadequado para tomada de decisão: Relatórios mal estruturados podem dificultar a priorização das correções, especialmente em projetos grandes e complexos.

Conclui-se, portanto, que a eficácia da verificação automática de modelos BIM depende de uma implementação coordenada e robusta de seus quatro estágios fundamentais. Compreender os desafios específicos de cada etapa é essencial para avaliar o desempenho das ferramentas atuais e propor melhorias. Na seção seguinte, serão apresentados alguns dos principais sistemas de verificação existentes, com o objetivo de compreender como cada um lida com essas etapas e quais estratégias têm adotado para superar as limitações identificadas.

2.5. SISTEMAS DE VERIFICAÇÃO AUTOMÁTICA EXISTENTES

Como parte do levantamento sobre o estado atual da verificação automática de modelos BIM, foram pesquisadas diversas iniciativas que contribuíram, ou ainda contribuem, para o desenvolvimento dessa prática. Essas soluções surgiram em contextos distintos, com propósitos variados, mas compartilham o objetivo comum de aprimorar a análise da qualidade da informação nos modelos digitais.

Para fins de análise nesta dissertação, as abordagens identificadas foram agrupadas em cinco categorias principais:

- I. iniciativas institucionais,
- II. soluções proprietárias,
- III. abordagens baseadas em padrões abertos,
- IV. propostas desenvolvidas em contexto acadêmico,
- V. e abordagens recentes orientadas à experiência do usuário ou à incorporação de tecnologias emergentes.

Cada uma dessas categorias será explorada a seguir.

2.5.1. INICIATIVAS INSTITUCIONAIS

As iniciativas institucionais referem-se aos esforços conduzidos por entidades governamentais para implementar sistemas de verificação automática de modelos BIM. Motivadas pela necessidade de padronizar requisitos em escala nacional, essas iniciativas foram pioneiras na consolidação da verificação automatizada da conformidade e continuam a servir como referência para outras abordagens.

Uma das principais vantagens desse tipo de solução é a clareza e robustez dos requisitos utilizados, geralmente bem definidos e formalizados em normas técnicas. No entanto, por serem concebidas para atender a regulamentações específicas de um determinado país ou região, sua aplicabilidade tende a ser limitada ao âmbito da sua jurisdição.

2.5.1.1. CORENET - Singapura

A iniciativa CORENET e-PlanCheck, liderada pelo governo de Singapura desde o início dos anos 2000, foi uma das primeiras tentativas estruturadas de implementar a verificação automática de conformidade normativa em modelos BIM. Desenvolvida com o objetivo de digitalizar e automatizar a submissão de projetos para licenciamento, a plataforma permitia que modelos BIM fossem analisados automaticamente com base nos requisitos do *Building and Construction Authority* (BCA).

Para tornar isso possível, a CORENET foi construída sobre uma infraestrutura técnica robusta, centrada no uso do EDM *Model Checker*, um motor de verificação desenvolvido pela empresa norueguesa Jotne. Essa ferramenta opera sobre modelos IFC e permite a definição de regras personalizadas usando a linguagem EXPRESS, a mesma que estrutura o próprio esquema IFC. O EDM *Model Checker* é complementado pelo EDM *Model Server*, que armazena os modelos como objetos semânticos, permitindo análises em larga escala e o gerenciamento simultâneo de múltiplas versões.

Buscando ampliar as capacidades semânticas da verificação, o governo de Singapura também promoveu o desenvolvimento do FORNAX, criado pela empresa local novaCITYNETS. O FORNAX atua como uma camada adicional ao EDM, estendendo o modelo IFC com objetos derivados e cálculos

automatizados (como relações espaciais e propriedades geométricas). Isso permitiu verificar não apenas atributos diretamente presentes no modelo, mas também informações inferidas, o que ampliou significativamente o escopo da verificação normativa. O sistema ainda conta com um visualizador gráfico próprio e funcionalidades de exportação de relatórios em múltiplos formatos (HTML, PDF, Word).

Apesar de sua sofisticação, o sistema apresenta limitações. A codificação das regras diretamente em linguagens de programação tradicionais torna sua manutenção complexa, e a interoperabilidade é limitada devido ao uso de extensões proprietárias. Além disso, a documentação pública sobre os critérios de validação dos modelos é escassa, o que levanta dúvidas sobre a consistência dos resultados.

Ainda assim, a CORENET permanece em operação até hoje e segue sendo uma das referências mais consolidadas em termos de automação da verificação normativa, influenciando iniciativas em outros países e servindo como base para estudos acadêmicos e técnicos.

2.5.1.2. DesignCheck – Austrália

O DesignCheck foi desenvolvido como parte de um projeto financiado pelo *Cooperative Research Centre for Construction Innovation*, com apoio do *Australian Building Codes Board* (ABCB). A plataforma visava automatizar a verificação de conformidade dos projetos com o *Building Code of Australia* (BCA), utilizando modelos IFC como base.

A solução adotava o EDM *Model Server* como repositório e usava mapeamentos Express-X para traduzir os dados do modelo para o ambiente de verificação. Diferente de outras iniciativas focadas apenas em órgãos reguladores, o DesignCheck também era voltado para uso por arquitetos e projetistas, permitindo a verificação em diferentes fases do projeto.

Uma limitação relevante era a ausência de uma interface gráfica de visualização dos modelos, o que tornava os relatórios textuais menos acessíveis para alguns profissionais. O sistema não está atualmente em operação, mas sua abordagem técnica influenciou diversas pesquisas subsequentes.

2.5.1.3. SMARTcodes – Estados Unidos

O projeto SMARTcodes, promovido pelo *International Code Council* (ICC), buscou transformar regulamentos técnicos da construção civil em regras digitalmente interpretáveis. Desenvolvido em colaboração com empresas como AEC3 e *Digital Alchemy*, o sistema utilizava classificações como o *Omniclass* e ferramentas como o *SMARTCodes Builder* para converter textos normativos em estruturas computacionais.

A verificação dos modelos podia ser feita por softwares como *Solibri Model Checker* ou AEC3 XABIO, e os resultados eram apresentados em formatos variados (HTML, PDF, XML etc.), com destaque visual para inconformidades.

Embora inovador, o projeto enfrentou barreiras na conversão manual dos códigos em regras lógicas, exigindo grande esforço técnico. O SMARTcodes teve aplicação limitada e não chegou a ser implementado em larga escala, mantendo-se principalmente como uma referência teórica e acadêmica.

2.5.1.4. Statsbygg – Noruega

O órgão governamental Statsbygg, responsável por projetos públicos na Noruega, tem investido desde 2012 em métodos de verificação automatizada com base em IFC. Um dos marcos dessa atuação foi o projeto HITOS, que integrou ferramentas como dRofus e Solibri Model Checker para a verificação de requisitos espaciais e de acessibilidade.

A iniciativa combinava um EDM Model Server com sistemas de verificação em diferentes fases do projeto, além de buscar — junto com a *Standards Norway* — o desenvolvimento de uma plataforma aberta para formalização de regras normativas.

Embora enfrente limitações similares a outras iniciativas (como a dependência da qualidade dos modelos e o custo de adaptação de regras), o Statsbygg permanece ativo e continua promovendo a integração entre regulamentação, tecnologia e interoperabilidade.

Com base na análise realizada, a Tabela 1 apresenta uma síntese comparativa das principais iniciativas institucionais de verificação automática de modelos BIM.

Tabela 1 - Síntese das iniciativas institucionais

Iniciativa	Foco Principal	Pontos Importantes	Limitações
CORENET	Licenciamento automático de projetos BIM	Pioneirismo, integração com processos oficiais, cobertura normativa ampla	Manutenção difícil, uso de extensões proprietárias, pouca interoperabilidade
Design Check	Verificação com base na Building Construction Authority	Aplicável em diferentes fases de projeto, voltado também para projetistas	Sem visualização gráfica, dependência técnica elevada
SMARTcodes	Formalização computacional de regulamentos	Estrutura semântica robusta, colaboração com entidades normativas	Conversão manual trabalhosa, uso limitado fora do meio acadêmico
Statsbygg	Verificação em projetos públicos	Base em <i>OpenBIM</i> , foco em integração de múltiplas ferramentas	Dependência da qualidade do modelo IFC, necessidade de adaptação contínua

2.5.2. SISTEMAS PROPRIETÁRIOS

Entre as soluções privadas desenvolvidas para a verificação automática de modelos BIM, destaca-se a plataforma Solibri, uma das ferramentas mais consolidadas e amplamente utilizadas no setor. Diferente das iniciativas institucionais, que buscam atender a normas nacionais específicas, os sistemas

proprietários têm como foco principal as necessidades do mercado e dos seus clientes, oferecendo produtos com maior suporte técnico, interfaces intuitivas e ciclos contínuos de atualização.

Esse modelo, no entanto, implica também em algumas limitações, como custos elevados de licenciamento, dependência de formatos fechados e menor transparência na definição e compartilhamento das regras de verificação. Além disso, como os dados ficam atrelados a um ambiente proprietário, há incertezas quanto à longevidade e interoperabilidade das informações no longo prazo, um ponto crítico quando se considera que edifícios são projetados para durar décadas.

Apesar dessas restrições, a Solibri desempenhou um papel central na evolução das práticas de *Automated Code Compliance* e *Quality Assurance* dentro do universo BIM. As subseções a seguir apresentam suas principais soluções e funcionalidades voltadas à verificação automatizada de modelos.

2.5.2.1 Solibri

A Solibri Inc., empresa finlandesa com mais de 25 anos de atuação no setor da construção digital, é responsável por uma das soluções mais consolidadas no campo da verificação automática de modelos BIM. Suas ferramentas são voltadas à garantia da qualidade e controle de conformidade com foco em fluxos baseados no formato aberto IFC.

O *Solibri Model Checker* (SMC) é uma das ferramentas mais amplamente utilizadas para verificação automatizada de modelos BIM. Compatível com o formato IFC, o software permite importar modelos provenientes de diversas plataformas de autoria, centralizando a análise em uma interface gráfica unificada e acessível.

Sua principal funcionalidade reside na aplicação de *rulesets* — conjuntos de regras predefinidas ou personalizadas — que podem ser configurados para validar aspectos como:

- Requisitos de acessibilidade
- Normas de segurança contra incêndios
- Qualidade de dados
- Detecção de interferências (*clash detection*)
- Presença ou ausência de propriedades obrigatórias

As não conformidades são comunicadas diretamente no modelo, com suporte ao BCF (*BIM Collaboration Format*) para facilitar a rastreabilidade e a gestão de *issues* em ambientes colaborativos.

A interface do SMC é conhecida por sua usabilidade, permitindo a criação de verificações sem necessidade de conhecimentos avançados em programação. No entanto, a personalização de regras mais complexas — além do escopo oferecido nas bibliotecas internas — requer o uso da API do Solibri, que não está disponível para todos os usuários, limitando a flexibilidade da ferramenta em projetos mais específicos.

Outro recurso relevante é o *Solibri Autorun*, que permite automatizar a execução de verificações em ciclos regulares durante as fases de projeto. Os resultados podem ser exportados em formatos como PDF, XLS e XML, além de visualizações internas no ambiente do próprio SMC.

Apesar de sua ampla adoção, o SMC enfrenta críticas quanto à falta de padronização formal das regras fora do seu ambiente proprietário, o que dificulta sua reutilização em outras plataformas ou sua integração com fluxos *OpenBIM* mais transparentes.

Em janeiro de 2025, a Solibri lançou o Solibri *Checkpoint*, uma plataforma baseada em nuvem voltada à coordenação e verificação de modelos BIM diretamente pelo navegador, sem necessidade de instalação local. A ferramenta marca um movimento estratégico da empresa em direção à computação em nuvem, após a aquisição da startup holandesa Xinaps BV, responsável pelo produto Verifi3D. Inicialmente, a ferramenta está disponível apenas nos Estados Unidos, Reino Unido e Irlanda.

O *Checkpoint* oferece três tipos principais de verificação:

- *Clash Check*: Permite a identificação de interferências entre elementos, definindo um alvo base, um alvo de colisão e um critério de tolerância. A Figura 5 abaixo mostra como essa funcionalidade é apresentada na interface da ferramenta, permitindo configurar os parâmetros de verificação e visualizar os resultados no *Data Viewer*, onde os elementos em conflito podem ser agrupados, analisados e convertidos em *issues* para rastreamento.

The screenshot shows the 'Rule' configuration interface in Solibri Checkpoint. At the top right, there are two buttons: '3D Preview' and 'Save'. Below them is a text input field containing 'Clash Check'. Underneath is a 'Rule Type' dropdown menu, also set to 'Clash Check'. The 'Base target' section shows a selection box with 'Hospital_Sprinkler_2022.rvt' and a plus icon. The 'Colliding target' section shows a selection box with 'Hospital_Structural_2022.rvt' and a plus icon. At the bottom, the 'Tolerance' is set to '5' with a unit dropdown menu currently showing 'mm'.

Figura 5 - Clash check do Solibri Checkpoint

- *Property Check*: Focado na validação de propriedades, permite que usuários filtrem elementos (por exemplo, todas as portas do modelo) e verifiquem atributos específicos, selecionando o *property set*, a propriedade, a origem do dado e o operador lógico para a comparação (ex.: "igual a", "não igual a", "vazio", "não vazio", etc.). A Figura 6 a seguir ilustra essa funcionalidade, evidenciando como os filtros são definidos e quais parâmetros podem ser configurados para a verificação automática.

The screenshot shows the 'Property Check' rule configuration in Solibri Checkpoint. At the top, there are '3D Preview' and 'Save' buttons. The rule name is 'Property Check'. The 'Rule Type' is set to 'Property Check'. Under the 'Target' section, 'Elements you want to check' includes 'Doors'. The 'Properties' section has a '+Add property' button and a trash icon. Below this, a table defines the property check:

Property set	Property	Defined On	Operator
	Classification.UniFormat.2010.Number	Default	Is Not Empty

Figura 6 - Property Check do Solibri Checkpoint

- *Free Space Check*: Avalia requisitos espaciais, permitindo definir um elemento-alvo, um espaço livre obrigatório e elementos que devem ou não estar dentro desse espaço. A Figura 7 abaixo demonstra como essa funcionalidade é configurada, permitindo ao usuário especificar zonas de tolerância e restrições de ocupação para garantir o cumprimento de requisitos espaciais no modelo.

The screenshot shows the 'Free Space Check' rule configuration in Solibri Checkpoint. At the top, there are '3D Preview' and 'Save' buttons. The rule name is 'Free Space Check: FA Panels Clearance'. The 'Rule Type' is set to 'Free Space Check'. Under the 'Target' section, 'Elements to have free space box' includes 'Fire Alarm Control Panel'. The 'Required' section, 'Elements that must be inside free space box', is empty. The 'Disallowed' section, 'Elements that can't be inside free space box', includes 'All elements'. The 'Box dimensions' section defines the free space box with the following values:

Dimension	Value	Unit	Relative
Width	W 1	m	☐
Height	H 2	m	☐
Depth	D 0.9	m	☒

The 'Box alignment' section shows three icons for box alignment: 'X 0', 'Y 0', and 'Z 0'.

Figura 7 - Free Space Check do Solibri Checkpoint

Além das verificações, Solibri *Checkpoint* conta com um *Data Viewer* para análise estruturada das informações e permite exportação de relatórios em diversos formatos. O rastreamento das não conformidades pode ser feito internamente na plataforma ou por meio da integração com ferramentas externas, como *Procore Issues*, *Autodesk Construction Cloud (ACC) Issues* e *BIM 360 Issues*.

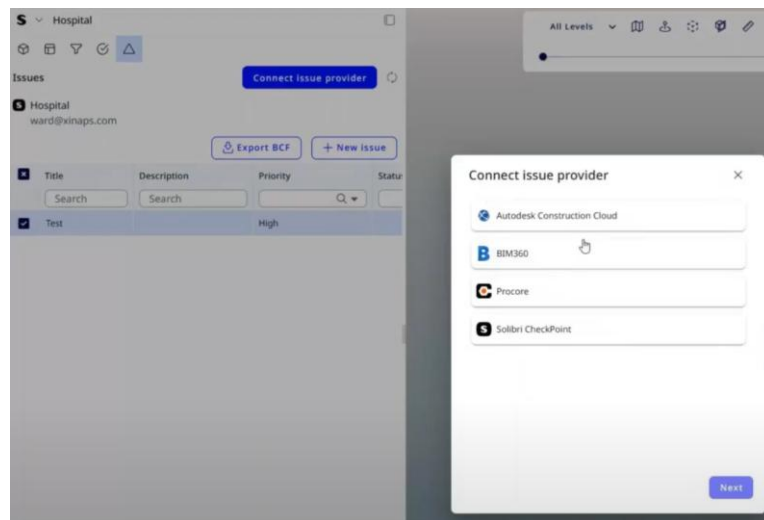


Figura 8 - Conexões do Solibri Checkpoint

A ferramenta suporta arquivos RVT e IFC e permite sincronização ao vivo dos modelos, desde que o software vinculado suporte essa funcionalidade. Diferente de outras soluções da Solibri, o *Checkpoint* opera com um modelo de assinatura separado, o que pode representar custos adicionais para a entidade contratante, dependendo da necessidade de integração com outros produtos da marca.

2.5.3. ABORDAGENS BASEADAS EM PADRÕES ABERTOS

As abordagens baseadas em padrões abertos têm como objetivo central promover a interoperabilidade entre plataformas, reduzindo a dependência de formatos proprietários e tornando os processos de modelação e verificação mais acessíveis, transparentes e sustentáveis no longo prazo. Diferentemente das soluções comerciais fechadas, essas iniciativas são geralmente lideradas por organizações sem fins lucrativos, consórcios internacionais ou grupos de pesquisa, buscando criar uma base comum para a troca estruturada de informação.

Nesse contexto, destacam-se duas grandes frentes: (i) as abordagens baseadas em tecnologias semânticas e ontológicas — que, combinadas, exploram a formalização computacional do conhecimento da construção civil — e (ii) o fluxo de trabalho *OpenBIM* promovido pela *buildingSMART*, que organiza e operacionaliza essa lógica em processos replicáveis, suportados por padrões abertos.

2.5.3.1. Tecnologias Semânticas e Ontológicas

As abordagens semânticas e ontológicas buscam representar de forma computacionalmente formal o conhecimento contido nos projetos e regulamentos da construção civil. Nesse contexto, as ontologias desempenham um papel central ao definirem, por meio de classes, propriedades e relações, os conceitos fundamentais do domínio da construção. Essa estrutura permite que os sistemas percebam o significado

dos dados, viabilizando inferências automáticas e análises baseadas em regras mais complexas. Já as tecnologias semânticas fornecem a infraestrutura técnica — como linguagens, formatos e mecanismos de consulta — que permite operar sobre essas ontologias com alto grau de precisão e consistência.

Com base em tecnologias como OWL (*Web Ontology Language*), RDF (*Resource Description Framework*), SHACL (*Shapes Constraint Language*) e SPARQL (*SPARQL Protocol and RDF Query Language*), essas metodologias permitem estruturar regras, conceitos e relações de maneira que possam ser interpretadas por máquinas com alto grau de precisão semântica.

Esse tipo de abordagem é particularmente útil para expressar requisitos complexos e contextuais, que não podem ser verificados apenas por atributos isolados no modelo BIM. Por exemplo, pode-se descrever que “todo compartimento destinado a permanência prolongada deve ter ao menos uma abertura para o exterior com área mínima de 1/8 da área do piso”, e a lógica envolvida — incluindo inferência sobre o uso do espaço — pode ser representada por meio de grafos semânticos e validações baseadas em *shapes*.

Embora tecnicamente promissoras, essas tecnologias apresentam barreiras significativas para sua adoção no setor AEC. O principal desafio é a curva de aprendizagem: elas exigem conhecimentos avançados de modelagem ontológica, lógica formal, linguagens de marcação e programação semântica, competências raramente encontradas entre os profissionais da indústria. Além disso, a construção de ontologias relevantes e interoperáveis demanda esforço colaborativo e maturidade institucional, o que limita sua aplicação em escala comercial no curto prazo.

Ainda assim, essas abordagens têm sido exploradas em diversos projetos acadêmicos e iniciativas-piloto, servindo de base para a formalização de requisitos regulatórios e para o desenvolvimento de sistemas de verificação mais flexíveis e inteligentes.

2.5.3.2. O fluxo OpenBIM da buildingSMART

Como resposta pragmática aos desafios da interoperabilidade e da verificação automatizada de modelos, a organização internacional *buildingSMART* propôs um fluxo de trabalho *OpenBIM* estruturado em etapas claras, apoiado por um conjunto de padrões abertos interconectados. Esse fluxo se apresenta como uma alternativa escalável e colaborativa aos sistemas proprietários, permitindo a comunicação eficaz entre diferentes agentes e ferramentas ao longo do ciclo de vida de um projeto.

O processo proposto é dividido em quatro etapas principais: Requerer, Produzir, Verificar e Entregar.

- **Requerer:** define os requisitos de informação do projeto, especificando o que deve ser modelado, com que nível de detalhe e em que formato;
- **Produzir:** consiste na elaboração do modelo com base nos requisitos estabelecidos;
- **Verificar:** envolve a checagem da conformidade do modelo em relação aos requisitos, com o apoio de ferramentas automatizadas;
- **Entregar:** refere-se ao compartilhamento dos dados validados com os stakeholders, garantindo rastreabilidade e integridade da informação.

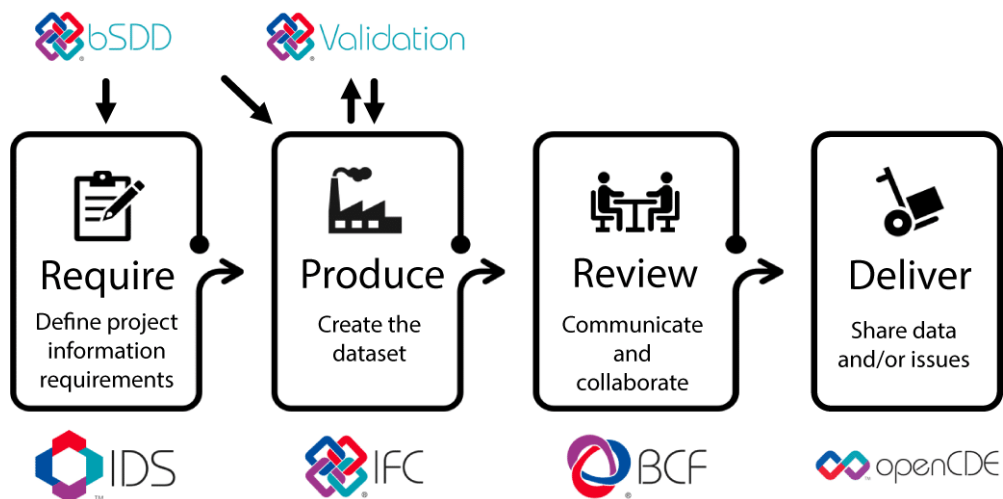


Figura 9 - Fluxo de trabalho proposto pela buildingSMART (buildingSMART, 2025a)

Diversos padrões desenvolvidos ou mantidos pela buildingSMART dão suporte a esse fluxo:

- **IDS** (*Information Delivery Specification*): formato baseado em XML que permite formalizar requisitos de informação específicos — por exemplo, “cada escada deve conter uma propriedade ‘número de degraus’ com valor superior a 1”. Além de sua estrutura legível por máquinas, pode ser convertido em PDF para facilitar a interpretação humana e o uso em contratos.
- **IFC** (*Industry Foundation Classes*): principal padrão aberto para representação semântica de modelos BIM, permitindo a troca de dados entre diferentes plataformas de forma neutra e durável.
- **BCF** (*BIM Collaboration Format*): protocolo leve para comunicação de *issues* detetadas durante o processo de verificação, facilitando a colaboração entre modeladores, coordenadores e revisores de qualidade.
- **openCDE**: especificação para ambientes comuns de dados (CDE) abertos e interoperáveis, permitindo o armazenamento, rastreamento e controle da informação em plataformas diversas.

O formato IFC é geralmente escolhido para integrar dados de diferentes disciplinas da engenharia, sobretudo quando há a possibilidade de os utilizadores não disporem do software necessário para abrir um formato proprietário específico (Mojtaba Valinejadshoubi et al., 2024).

O fluxo *OpenBIM* cobre integralmente o ciclo de verificação de qualidade, correspondendo diretamente aos quatro estágios descritos por Eastman (interpretação das regras, preparação do modelo, execução da verificação e análise dos resultados). Além disso, oferece respostas diretas às limitações identificadas em sistemas anteriores, como a opacidade das regras, a dificuldade de visualização dos resultados e a dependência de habilidades técnicas avançadas.

Ao se basear em padrões abertos, gratuitos e bem documentados, o *OpenBIM* promove processos mais transparentes, reproduzíveis e sustentáveis. Essa abordagem se mostra especialmente eficaz em contextos colaborativos e em projetos que envolvem múltiplas ferramentas e disciplinas, oferecendo uma estrutura sólida para implementar fluxos de QA/QC interoperáveis e automatizáveis.

2.5.4. EXPERIMENTOS ACADÊMICOS

Esta categoria diz respeito a investigações conduzidas em ambiente acadêmico, normalmente por estudantes ou grupos de investigação, que se dedicam a propor e testar fluxos de verificação automática com caráter exploratório. Tais estudos, desprovidos de vínculos comerciais, tendem a priorizar a validação de conceitos, a análise crítica da interoperabilidade entre ferramentas e a aplicação de requisitos informacionais formalmente definidos.

Na maior parte dos casos, essas investigações experimentais utilizam padrões abertos (como o IFC), ferramentas de código livre, linguagens semânticas e metodologias estruturadas, em especial a sequência de estágios proposta por Eastman, como base para analisar o potencial e os entraves do uso de tecnologias abertas na verificação da qualidade de modelos BIM. Embora seus resultados raramente estejam prontos para aplicação direta em escala de projeto real, essas pesquisas são fundamentais para revelar entraves técnicos e indicar boas práticas emergentes.

Uma das contribuições recentes mais relevantes é uma revisão abrangente da literatura que mapeia estudos acadêmicos voltados ao controle de qualidade em modelos BIM. Essa análise identifica as ferramentas utilizadas, os métodos aplicados e as principais lacunas nos processos de verificação abordados (M. Valinejadshoubi et al., 2024). A seguir, apresenta-se a Tabela 2 com a síntese com os principais resultados desse levantamento.

As conclusões extraídas por Valinejadshoubi et al. revelam importantes fragilidades no uso do padrão IFC como base para interoperabilidade entre modelos. Embora amplamente adotado, sua aplicação prática ainda enfrenta diversos desafios. Um dos principais diz respeito à perda de dados durante a conversão entre formatos proprietários e o formato IFC, afetando propriedades paramétricas, ligações entre elementos e até materiais atribuídos, com relatos de alterações não intencionais após a exportação.

Os autores também destacam que os modelos em IFC frequentemente não obedecem de forma consistente aos padrões exigidos, o que gera ambiguidade na interpretação dos dados e limita sua reutilização. Além disso, a estrutura dos conjuntos de propriedades (*property sets*) — embora extensível — requer nomes únicos e conformes às normas, tornando o processo menos acessível e mais propenso a erros, especialmente considerando que muitos profissionais dependem de softwares comerciais com opções limitadas de personalização.

Outra limitação importante é a dificuldade do IFC em representar informações de qualidade não estruturadas ou relações específicas presentes em bases de dados de controle. Isso compromete sua aplicabilidade em contextos que exigem parâmetros personalizados ou definições mais detalhadas de requisitos.

Em suma, embora o IFC represente uma base essencial para práticas de controle de qualidade baseadas em BIM, os desafios associados à integridade dos dados, personalização de parâmetros e interoperabilidade prática demonstram a necessidade de soluções mais robustas e adaptáveis (M. Valinejadshoubi et al., 2024)

Tabela 2 - Estudos anteriores sobre soluções de controle de qualidade baseadas em BIM. (Adaptado de M. Valinejadshoubi et al., 2024)

Autor	Ferramenta Desenvolvida	Propósito	Limitações
(Noardo et al., 2021)	Sistema QC baseado em IFC	Encontrar padrões em <i>datasets</i> baseados em IFC para detetar discrepâncias	*Possibilidade de perda de informação durante exportação IFC. *Não automatizado.
(Hamledari et al., 2018)	Algoritmo de controle de qualidade para modelo de dados IFC	Analisar o modelo de dados IFC para identificar discrepâncias de forma automatizada	*Possibilidade de perda de informação durante exportação IFC.
(Choi et al., 2020)	Sistema QC baseado em IFC	Realizar o controle de qualidade de modelos BIM	*Investigação limitada apenas à qualidade de modelos arquitetónicos. *Contém erros durante exportação IFC. *Não automatizado.
(Donato et al., 2018)	Método de controle de qualidade em BIM	Avaliar a qualidade de modelos BIM com base em listas de verificação personalizadas	*Investigação limitada apenas à qualidade de modelos arquitetónicos. *Não automatizado.
(Xu et al., 2018)	Soluções integradas em IFC	QC baseadas Utilizar algoritmos de redes neurais para avaliar a qualidade da construção	*Possibilidade de perda de informação durante exportação IFC. * Não considera a qualidade do projeto. *Não automatizado.
(Temel & Başağa, 2020)	Sistema de ACC baseado em IFC	Automatizar a verificação de conformidade	*Possibilidade de perda de informação durante exportação IFC.
(Pazlar & Turk, 2008)	Sistema QC baseado em IFC	Propor um método para verificar a qualidade de modelos BIM	*Possibilidade de perda de informação durante exportação IFC. *Não automatizado.

Com o intuito de complementar essa revisão e reforçar os pontos discutidos, foram selecionados quatro estudos de caso que propõem fluxos de verificação completos, alinhados com os quatro estágios estabelecidos por Eastman. Esses exemplos ilustram, de forma concreta, como diferentes estratégias têm sido testadas em ambiente acadêmico e quais ferramentas e métodos têm se mostrado mais promissores. A análise dos trabalhos a seguir foca nos aspectos técnicos, operacionais e conceituais de cada fluxo, permitindo identificar boas práticas, lacunas recorrentes e oportunidades de melhoria no uso de soluções abertas para a verificação automatizada da qualidade de modelos BIM.

	1. Interpretação de regras	2. Preparação do Modelo	3. Execução da regra	4. Relatório da verificação da regra
Automated BIM QA/QC Procedures with Practical Application Julio Joaquin Salaberri Martin	Lista de verificação personalizada criada no Autodesk Model Checker Configurator com base em IRS, LOIN, BEP e códigos locais.	O modelo BIM é fornecido pelas equipes de engenharia (Autodesk Revit) [foram disponibilizados modelos federados nos formatos IFC e NWD]. O Dynamo é utilizado para preencher lacunas de dados.	O Autodesk Model Checker é acionado manualmente. Múltiplos modelos podem ser verificados. A verificação pode ser automatizada, mas isso não foi implementado nesta dissertação.	O relatório gerado permite o registro de issues por meio do formato BCF (que pode ser utilizado pelo BIMCollab) ou a geração de relatórios de gestão através do Excel e do PowerBI.
Verificação automática de Modelos BIM Ricardo Manuel Monteiro Moço	É criado um conjunto de regras no Solibri Model Checker adaptado aos critérios do método MC-FEUP	Modelo público do Autodesk Revit 16 , sistema de classificação Omniclass	É utilizado o Solibri Model Checker , consoante a alguns ajustes no modelo	O próprio Solibri produz um issue report em Excel
Verificação de atendimento de requisitos de informação por elemento utilizando Information Delivery Specification em projetos de edificações residenciais apoiados em BIM Luiz Gusthavo Gonzaga Braga Silva	Criação de documento IDS com o IDS Editor da usBIM (ACCA)	Modelo elaborado em IFC para extração de quantitativos. Desenvolvido sem o auxílio do IDS.	Solibri e Blender fazem a verificação do modelo contra as regras definidas no mesmo IDS.	O Solibri apresenta um issue report em BCF. O Blender pode apresentar os resultados em HTML, ODS ou BCF.
The Development of an automated system for a quality evaluation of engineering BIM Models: a case study Mojtaba Valinejadshoubi	Lista de verificação personalizada com base no LOD, LOI e no BEP.	O modelo BIM é fornecido pelas equipes de engenharia de cada disciplina (estrutura, arquitetura, etc.), utilizando o Autodesk Revit.	O Dynamo extrai e analisa os dados, gerando uma pontuação de qualidade por disciplina, que é armazenada numa base de dados externa (Excel ou MySQL).	Esses dados são posteriormente carregados para o ProjectWise (ou Speckle) na nuvem, podendo então ser facilmente integrados ao PowerBI para visualização da informação.

Figura 10 - Estudos acadêmicos analisados

A imagem revela a diversidade de abordagens adotadas para enfrentar os desafios de cada etapa da verificação automática de modelos BIM. Essas abordagens englobam tanto soluções *open source* quanto ferramentas proprietárias consolidadas, combinadas para responder às especificidades de cada estudo. Essa variedade evidencia a ausência de um padrão amplamente adotado e, ao mesmo tempo, sublinha a flexibilidade que a pesquisa acadêmica proporciona ao explorar caminhos ainda pouco estruturados.

No que diz respeito à interpretação de regras, reconhecida como uma das etapas mais complexas e determinantes para o sucesso do processo de verificação, a maior parte dos autores recorreu à criação de *checklists* personalizados. Essas listas, elaboradas manualmente e adaptadas a cada projeto, representam uma tentativa de transpor exigências normativas para estruturas logicamente verificáveis. No entanto, os métodos utilizados para operacionalizar essa interpretação em ambientes digitais variaram significativamente. Alguns estudos utilizaram ferramentas comerciais com interfaces acessíveis, como o *Solibri Model Checker* e o *Autodesk Model Checker* (Moço, 2015; Salaberri Martin, 2022), ainda que menos flexíveis. Outros adotaram soluções gratuitas e orientadas a padrões abertos, como o *usBIM IDS Editor*, da *ACCA Software* (Silva, 2024), para criar conjuntos de regras baseados em IDS. Há ainda propostas mais estruturadas que sugerem o uso de parâmetros sistematizados a partir de documentos como LOD, LOI e BEP (M. Valinejadshoubi et al., 2024).

É importante notar que, em todos os casos, os modelos utilizados para verificação já haviam sido previamente desenvolvidos, o que permitiu avaliar a qualidade a posteriori com base em requisitos definidos apenas na fase de análise. Enquanto alguns estudos trabalharam com um único modelo multidisciplinar, outros optaram por analisar modelos separados, mas pertencentes ao mesmo projeto, a fim de detetar fragilidades disciplinares específicas.

Na etapa de execução das regras, os estudos refletem uma divisão clara entre aqueles que utilizaram ferramentas proprietárias com sistemas de checagem integrados e aqueles que adotaram abordagens customizadas ou *open source*. Um exemplo disso é o estudo que comparou a aplicação de um mesmo conjunto de regras, codificado via IDS, em duas plataformas distintas: o Blender, com o *add-on* Bonsai, e o Solibri. Ambas as ferramentas oferecem suporte à leitura de ficheiros IDS, mas apresentaram diferenças na forma como interpretam determinados filtros e critérios de seleção (Silva, 2024). Essa variação não comprometeu a confiabilidade do processo de verificação, mas evidencia a importância de uma comunicação clara dos requisitos e da escolha criteriosa dos softwares utilizados, especialmente quando se busca garantir consistência entre diferentes ambientes de modelação e análise.

Por outro lado, uma abordagem distinta foi adotada por outro autor, que utilizou o Dynamo — uma plataforma de programação visual baseada no Revit — para extrair dados do modelo e calcular automaticamente uma pontuação de qualidade por disciplina. Essa pontuação era então armazenada em uma base de dados externa, a partir da qual se geravam os relatórios de conformidade (M. Valinejadshoubi et al., 2024).

A diversidade de métodos se estende também à etapa de relatório, na qual os autores selecionaram formas distintas de comunicar os resultados das verificações. Aqueles que utilizaram ferramentas proprietárias, como o Autodesk Model Checker ou o Solibri, exploraram os relatórios automáticos oferecidos pelas próprias plataformas. Ambos permitem exportar resultados em formatos versáteis, como planilhas Excel e issues no formato BCF, que podem ser posteriormente integrados a ferramentas de coordenação, como o BIMCollab. A possibilidade de exportar relatórios em tabelas estruturadas, compatíveis com softwares amplamente utilizados na indústria como o Excel ou PowerBI, foi destacada como um diferencial importante para viabilizar a adoção desses fluxos no cotidiano dos escritórios de projeto.

Os fluxos baseados em IDS demonstraram, por sua vez, uma atenção especial à rastreabilidade dos resultados. O uso do Blender possibilitou a geração automatizada de relatórios nos formatos HTML, ODS e BCF — o que reforça a compatibilidade com diferentes sistemas de gestão e promove a interoperabilidade da informação. Em uma das propostas analisadas, os dados extraídos foram carregados para plataformas como ProjectWise ou Speckle, e posteriormente integrados ao Power BI,

permitindo uma visualização interativa e rica dos indicadores de qualidade (M. Valinejadshoubi et al., 2024).

Essas abordagens, embora diversas, oferecem lições valiosas sobre as escolhas que devem ser feitas ao se propor um fluxo de trabalho de verificação automatizada. Algumas conclusões emergem com clareza a partir dos estudos analisados:

- A prática de construir listas de verificação manualmente continua sendo uma estratégia recorrente, mas sua limitação é evidente quando se busca escalar ou adaptar regras em diferentes contextos;
- Nenhum dos fluxos estudados se mostrou plenamente automatizado e confiável, o que levou os próprios autores a recomendarem a realização de verificações manuais complementares;
- A maior parte do esforço necessário para viabilizar a verificação automática continua concentrada na etapa inicial, de interpretação e formalização das regras;
- As ferramentas comerciais, embora maduras, apresentam baixa flexibilidade para acomodar novos requisitos, tornando-se pouco adaptáveis em projetos com exigências específicas;
- Em fluxos baseados em IDS, foi observada uma variação significativa nos resultados da verificação, o que aponta para uma interpretação distinta das regras entre diferentes plataformas e, consequentemente, reforça a necessidade de uma padronização mais rigorosa da estrutura da informação;
- A eficácia da verificação está intimamente relacionada à forma como o modelo foi desenvolvido, isto é, ao grau de estruturação, completude e coerência dos dados inseridos durante a modelação.

A análise detalhada desses experimentos acadêmicos permite identificar os avanços mais relevantes e, sobretudo, os pontos críticos que ainda impedem a consolidação de uma abordagem universal para a verificação automatizada. A partir dessa compreensão, torna-se possível justificar de forma mais precisa as escolhas metodológicas adotadas na presente dissertação, que busca equilibrar flexibilidade, rastreabilidade e grau de automatização a partir de um fluxo baseado em ferramentas interoperáveis. A diversidade de soluções observadas reforça a convicção de que, para além da ferramenta, o verdadeiro diferencial está na estruturação inteligente do processo como um todo, do requisito à verificação, e da verificação à gestão dos resultados.

2.5.5. ABORDAGENS RECENTES

Além das soluções já consolidadas, vêm ganhando espaço nos últimos anos novas abordagens que exploram tecnologias emergentes, melhorias na experiência do usuário e maior integração com os ambientes de modelação. Embora muitas dessas propostas ainda estejam em fase experimental, elas refletem uma tendência crescente de diversificação dos métodos de verificação automática.

Entre essas iniciativas, destacam-se os *add-ons* incorporados a softwares de modelação, como o *UpCodes* para Revit, que oferecem feedback imediato aos projetistas; as soluções centradas no utilizador, como o ACABIM, que combinam automação com interação humana em fluxos semiautomatizados; e o uso de inteligência artificial e aprendizado de máquina, aplicados à classificação, inferência e enriquecimento semântico dos modelos.

Mais recentemente, também têm surgido estudos exploratórios com modelos de linguagem (LLMs) para auxiliar na interpretação automática de regulamentos e na formulação de regras de verificação,

apontando para um possível avanço na tradução de requisitos normativos complexos em lógica computacional legível por máquina.

2.6. SÍNTESE DO ESTADO DA ARTE

Este capítulo apresentou uma análise crítica e estruturada das principais abordagens, ferramentas e metodologias relacionadas à verificação automatizada da qualidade de modelos BIM. A crescente complexidade dos regulamentos, aliada à necessidade de precisão e rastreabilidade nos fluxos de informação, tem impulsionado o desenvolvimento de soluções voltadas à *Automated Code Compliance* e ao controle da qualidade informacional em ambientes digitais.

Ao longo da revisão, foram discutidos os fundamentos do QA/QC em BIM, o papel central dos requisitos de informação como ponto de partida para a verificação, os diferentes tipos de soluções existentes (institucionais, proprietárias, baseadas em padrões abertos, acadêmicas e emergentes), e os estágios envolvidos no processo de verificação automatizada, conforme propostos por Eastman.

Verificou-se que, embora haja avanços significativos, a verificação automática de modelos BIM ainda não alcançou maturidade suficiente para ser amplamente adotada na prática cotidiana do setor. A análise das soluções existentes revelou uma série de limitações técnicas, operacionais e institucionais que impedem sua implementação em larga escala.

Entre as abordagens analisadas, o *OpenBIM* desponta como a proposta mais alinhada com os princípios da interoperabilidade, da transparência e da padronização. Ao combinar padrões abertos como IFC, IDS, BCF e *openCDE* em um fluxo estruturado de verificação, oferece uma base promissora para fluxos de QA/QC mais robustos, acessíveis e replicáveis. No entanto, a sua adoção ainda é desigual e carece de validações práticas em contextos reais de projeto.

Em resposta às limitações identificadas ao longo deste capítulo, esta dissertação propõe um fluxo de trabalho automatizado para a verificação da qualidade de modelos BIM, fundamentado no uso de ferramentas interoperáveis baseadas em padrões abertos. A proposta visa, por um lado, avaliar com maior rigor a consistência e a conformidade dos modelos desenvolvidos e, por outro, estabelecer um método replicável e adaptável à diversidade de contextos e equipas de projeto. Ao enfrentar os principais desafios técnicos e metodológicos identificados na literatura espera-se contribuir para o fortalecimento das práticas de QA/QC em BIM, promovendo modelos mais fiáveis, auditáveis e alinhados com os requisitos de qualidade nas fases de concepção e desenvolvimento do projeto.

2.6.1. PRINCIPAIS LIMITAÇÕES

Apesar do progresso alcançado, a verificação automática de modelos BIM ainda enfrenta limitações estruturais que comprometem sua eficácia e aplicabilidade. Os principais entraves identificados na literatura e na análise das soluções existentes incluem:

- Criação e manutenção de regras computáveis: a conversão de normas regulatórias para lógica formal é trabalhosa, sujeita a interpretações diversas e exige atualizações constantes, dificultando a escalabilidade do processo (Amor & Dimyadi, 2021).
- Falta de padronização normativa entre regiões: a diversidade de códigos e regulamentos locais dificulta a reutilização de regras e a aplicação de soluções universais.

- Dependência da qualidade e completude dos modelos BIM: informações mal estruturadas ou ausentes no modelo IFC comprometem diretamente os resultados da verificação (Eastman et al., 2009).
- Baixa interoperabilidade em soluções proprietárias: o uso de formatos fechados e ambientes restritos limita a integração entre ferramentas e reduz a flexibilidade dos fluxos de trabalho.
- Necessidade de conhecimento técnico especializado: muitas soluções exigem domínio de linguagens como EXPRESS, XML ou SHACL, afastando profissionais da indústria sem perfil técnico avançado (Nawari, 2018).
- Dificuldade na interpretação e visualização dos resultados: mesmo em ferramentas com interface gráfica, a comunicação das não conformidades nem sempre é clara ou contextualizada, o que compromete a tomada de decisão.

Essas limitações indicam que, para que a verificação automática de informações se torne uma prática robusta e confiável, será necessário um esforço conjunto e contínuo entre academia, indústria e governos, incluindo ações de normalização, formação e desenvolvimento tecnológico.

2.6.2. LACUNAS DE PESQUISA E OPORTUNIDADES

Além das limitações técnicas já discutidas, observa-se um conjunto de lacunas teóricas e metodológicas que ainda precisam ser superadas para viabilizar fluxos de verificação automatizada verdadeiramente operacionais, interoperáveis e aplicáveis à realidade das empresas. Entre elas, destacam-se:

- A falta de critérios de qualidade amplamente aceitos e replicáveis, capazes de orientar a criação de regras de verificação de forma consistente e compreensível.
- A ausência de fluxos validados de ponta a ponta, que contemplem desde a definição dos requisitos até a geração e rastreamento dos relatórios de verificação.
- A escassez de estudos aplicados que demonstrem como padrões como o IDS e o IFC podem ser utilizados com ferramentas acessíveis e dentro de prazos realistas.
- O foco excessivo na detecção de colisões geométricas, em detrimento da verificação semântica e da conformidade com requisitos informacionais.
- A baixa maturidade na adoção do *OpenBIM* como base para QA/QC, com poucas empresas utilizando fluxos padronizados com IDS, BCF ou *openCDE* de forma integrada.
- O descompasso entre os avanços acadêmicos e a aplicação prática, com soluções promissoras como tecnologias semânticas, LLMs e IA ainda restritas a contextos experimentais.

Neste cenário, esta dissertação propõe-se a contribuir com o avanço da área ao desenvolver, aplicar e avaliar um fluxo de verificação baseado em padrões *OpenBIM*, acessível, claro e adaptável à realidade de projetos reais, priorizando a qualidade da informação e a viabilidade de adoção no setor.

3

METODOLOGIA

3.1 INTRODUÇÃO

Este capítulo tem como objetivo apresentar a metodologia adotada para o desenvolvimento da dissertação, explicando com detalhe o processo que será implementado no estudo de caso, descrito no capítulo seguinte. As decisões metodológicas aqui descritas foram construídas a partir das reflexões, lacunas e oportunidades identificadas no Capítulo 2 – Estado da Arte, que abordou o panorama atual da verificação de qualidade em modelos BIM, suas limitações e as soluções mais promissoras, com destaque para o potencial da abordagem *OpenBIM*.

Importa referir que o fluxo de trabalho aqui proposto não pretende substituir por completo os processos já utilizados pela organização, mas sim analisar criticamente práticas existentes e explorar alternativas mais alinhadas com os padrões abertos de produção e partilha de informação. A intenção é mitigar a dependência de tecnologias proprietárias, ou *vendor lock-in*, e propor soluções compatíveis com os fluxos atuais da empresa, preservando a continuidade dos dados e facilitando a interoperabilidade entre equipas internas e externas.

O estudo de caso que fundamenta esta investigação será realizado com base em um conjunto de modelos fornecidos pela empresa que acolhe esta dissertação. Esses modelos foram anteriormente submetidos a um processo interno de verificação de qualidade da informação, conduzido de forma mais manual, e agora serão reavaliados sob a ótica de um fluxo de trabalho automatizado. A disponibilização dos modelos por parte da empresa contempla arquivos nativos do Revit (.rvt) e versões exportadas no formato IFC (.ifc), abrangendo as principais disciplinas envolvidas no projeto: arquitetura, estrutura, sistemas elétricos, climatização (HVAC) e sistemas hidráulicos (*plumbing*). Além dos modelos, a empresa também forneceu uma tabela em formato Excel contendo os parâmetros que foram exigidos e verificados manualmente durante o processo original. Essa tabela será utilizada como referência para estabelecer os requisitos de informação no fluxo automatizado proposto, permitindo uma comparação direta entre os dois métodos.

3.2 DESCRIÇÃO DA METODOLOGIA

A metodologia proposta consiste na adoção de um fluxo de trabalho baseado em *OpenBIM*, conforme sugerido pela *buildingSMART*. Inicialmente, a equipa de coordenação BIM da empresa — responsável por avaliar a qualidade dos modelos — será encarregue de produzir um ficheiro IDS utilizando o editor usBIM da ACCA. Essa ferramenta permite a criação de dois formatos: um ficheiro .xml, interpretável por máquinas, e um ficheiro .pdf, legível por humanos. Ambos os ficheiros são enviados às equipas

modeladoras e funcionam como um guia prático para a elaboração dos modelos, indicando de forma clara os requisitos de informação que serão futuramente verificados.

Uma vez concluído o modelo, as equipas modeladoras devem exportá-lo no formato IFC, na versão previamente definida. O arquivo IFC é então enviado à equipa de coordenação, que o submeterá à verificação automática utilizando o software Blender, complementado pelo plugin Bonsai, capaz de interpretar o ficheiro IDS gerado originalmente. Como resultado, o processo gera um relatório em HTML que apresenta, de forma simples, os elementos que estão ou não em conformidade com os requisitos estipulados.

Além da geração do relatório, a plataforma Blender permite que os elementos reprovados na verificação sejam convertidos em *issues*, os quais podem ser exportados no formato BCF (*BIM Collaboration Format*). Esses *issues* são então enviados para um software de gestão compatível com o formato BCF, que neste estudo será o BIMcollab Zoom — amplamente utilizado no setor e já adotado pela empresa parceira. Com isso, as equipas modeladoras recebem os *issues*, realizam os ajustes necessários no modelo e o reenviam para nova rodada de verificação, fechando o ciclo de controle e garantia da qualidade com base em padrões abertos. A seguir, na Figura 11, apresenta-se um fluxograma ilustrando esse processo.

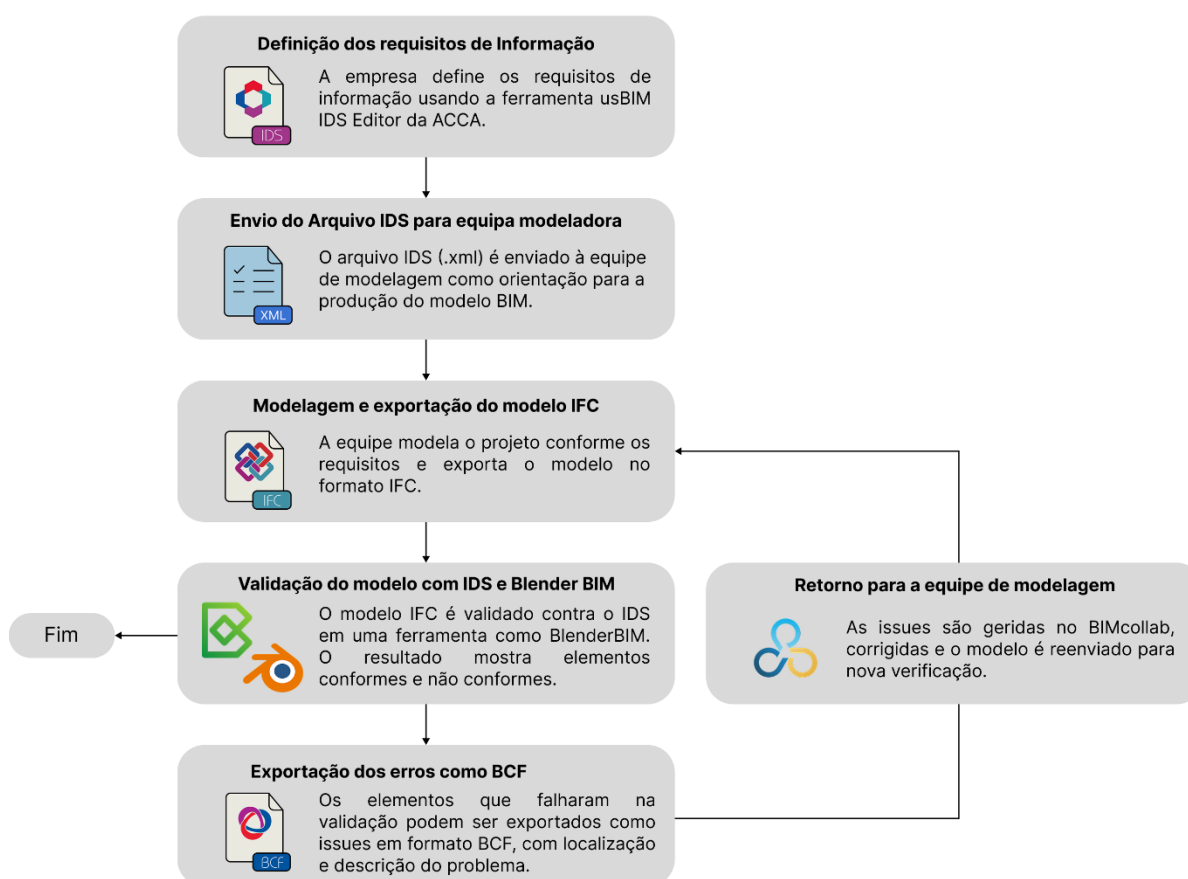


Figura 11 - Fluxograma do processo proposto

3.3 FERRAMENTAS UTILIZADAS

3.1.1 USBIM IDS EDITOR

O *usBIM.IDS Editor*, desenvolvido pela empresa italiana ACCA Software, foi a ferramenta selecionada para a criação do conjunto de requisitos de informação (IDS) que orientou a verificação dos modelos. Este editor permite definir especificações de entrega de informações de forma estruturada e conforme as normas da *buildingSMART*, facilitando a interoperabilidade em processos *OpenBIM*.

A ferramenta possibilita a geração de dois tipos principais de ficheiros: o ficheiro IDS em formato XML e o relatório em formato PDF. O ficheiro XML é o documento técnico que contém as regras e requisitos de informação de forma legível por máquinas, sendo utilizado diretamente nos processos automáticos de verificação do modelo. Já o ficheiro PDF é um extrato legível por humanos, que apresenta os requisitos de forma resumida e organizada, permitindo uma consulta rápida e manual dos parâmetros exigidos.

A escolha por esta ferramenta baseou-se em dois fatores principais: a possibilidade de utilizá-la sem custos, o que a torna viável para fins académicos, e a ampla oferta de materiais de suporte disponíveis na comunidade de usuários, o que facilitou a aprendizagem e a implementação prática do software em um curto espaço de tempo. Esses recursos facilitaram tanto o aprendizado quanto a aplicação prática da ferramenta em um curto período.

3.3.2 BLENDER E BONSAI

O Blender é um software de modelação tridimensional de código aberto, amplamente utilizado em diversas áreas como animação, efeitos visuais, jogos e, mais recentemente, em aplicações de arquitetura e construção.

No contexto deste trabalho, inicialmente foi utilizada a versão 4.0 do Blender como plataforma de visualização e verificação dos modelos IFC exportados, com suporte a fluxos *OpenBIM*. Durante o desenvolvimento, em março de 2025, o software foi atualizado para a versão 4.4, o que gerou algumas incompatibilidades temporárias com o plugin Bonsai. Apesar disso, a integração foi restabelecida e a ferramenta pôde ser utilizada normalmente. A natureza *open source* do Blender e sua capacidade de integrar extensões específicas para a área de construção civil foram fatores determinantes para a sua escolha.

A utilização do Blender permitiu a inspeção detalhada dos modelos e a execução do processo de verificação de informações com a integração do plugin Bonsai.

O Bonsai é um complemento (*add-on*) de código aberto desenvolvido para o Blender, destinado à criação, auditoria e análise de modelos IFC no contexto *OpenBIM*. O projeto, anteriormente conhecido como BlenderBIM, evoluiu para o nome atual com o objetivo de estabelecer uma identidade própria, distinta do Blender, embora continue sendo totalmente integrado a esta plataforma.

Construído sobre o motor de geometria e *toolkit* IFC da iniciativa *IfcOpenShell*, o Bonsai permite trabalhar diretamente com modelos IFC sem a necessidade de conversões ou softwares proprietários. A ferramenta oferece suporte à verificação automática de requisitos de informação, geração de relatórios de não conformidade e análise detalhada de dados do modelo.

Para a realização deste trabalho, foi utilizada a versão 0.8.2 do Bonsai, disponível à época da pesquisa, integrada ao Blender 4.0 e, posteriormente, na versão 4.4. O Bonsai foi utilizado para importar os

modelos IFC, aplicar as verificações definidas no IDS e gerar os ficheiros BCF correspondentes às não conformidades identificadas, possibilitando a continuidade do fluxo de gestão de qualidade.

3.3.2 BIMCOLLAB ZOOM

O BIMcollab Zoom é um software proprietário com uma versão gratuita de demonstração, desenvolvido para a visualização de modelos BIM e a gestão de *issues* através do formato BCF (*BIM Collaboration Format*). A ferramenta é parte do ecossistema BIMcollab, voltado para o gerenciamento colaborativo de modelos e comunicação entre equipas de projeto.

No contexto deste trabalho, o BIMcollab Zoom foi utilizado para importar os ficheiros BCF gerados no Bonsai e organizar visualmente as não conformidades identificadas. A interface da ferramenta permitiu a visualização precisa dos elementos associados a cada *issue*, bem como a sua categorização por tipo de erro, elemento afetado e especialidade correspondente.

Foi utilizada a versão 9.4 do BIMcollab Zoom, disponível à data da realização do estudo.

3.4 MODELO FEDERADO

Para a realização do estudo de caso que será apresentado no capítulo 4 e a validação do fluxo de trabalho proposto, a empresa parceira disponibilizou os ficheiros dos modelos digitais correspondentes a um projeto real, em formatos IFC e RVT.

Os ficheiros foram disponibilizados de forma segregada por especialidade, abrangendo as disciplinas de arquitetura, estruturas, instalações elétricas, climatização (HVAC) e sistemas hidráulicos.

Essa abordagem permitiu a análise individualizada dos modelos e a aplicação do fluxo proposto em diferentes contextos de modelação, algo que pode facilitar tanto a análise por parte dos coordenadores quanto a gestão de não conformidades no final do processo, uma vez que a passagem de informação pode ser mais direcionada e organizada por especialidade.

Cabe ressaltar que, embora os modelos não tenham sido desenvolvidos com base em um ficheiro IDS, o que reflete a prática comum em projetos anteriores à adoção plena de processos *OpenBIM*, os parâmetros informacionais foram previamente definidos entre a empresa e o cliente e documentados no BEP (*BIM Execution Plan*) do projeto. A planilha utilizada neste estudo para a criação dos ficheiros IDS reflete exatamente os parâmetros estabelecidos nesse acordo, servindo como base para a estruturação das regras de verificação implementadas posteriormente.

3.5 DEFINIÇÃO DOS REQUISITOS A SEREM VERIFICADOS

A definição dos requisitos de informação a serem verificados neste trabalho seguiu orientações fornecidas pela empresa parceira, com base em exigências contratuais frequentemente presentes em projetos internacionais. Esses requisitos foram inicialmente disponibilizados em formato textual e, posteriormente, organizados de forma sistemática para facilitar sua implementação no fluxo de verificação.

Além dessas exigências externas, a empresa também adota parâmetros informacionais voltados a necessidades internas, como a extração de quantitativos e a estruturação de medições. É o caso, por

exemplo, dos parâmetros nomeados com o prefixo GEG_5D, utilizados para identificar atributos relevantes ao processo de orçamentação e compatibilização com ferramentas de gestão de custos.

Foram definidos, para cada especialidade, os parâmetros que deveriam ser verificados, incluindo a identificação do tipo de informação esperada (por exemplo, atributos do tipo *string* ou *boolean*). A tabela a seguir apresenta a distribuição dos requisitos por especialidade, especificando os parâmetros correspondentes e o tipo de dado associado.

Tabela 3 - Requisitos a serem verificados

Parâmetros	Tipo	Arquitetônico	Estrutural	Elétrico	HVAC	Sistemas Hidráulicos
<i>Name</i>	Texto	Sim	Sim	Sim	Sim	Sim
<i>Material Name</i>	Texto	Sim	Sim	Sim	Sim	Sim
<i>IfcElement</i>	Texto	Sim	Sim	Sim	Sim	Sim
<i>Building Story</i>	Texto	Sim	Sim	Sim	Sim	Sim
<i>System Type</i>	Texto	Sim	Sim	Sim	Sim	Sim
<i>System</i>	Texto	Sim	Sim	Sim	Sim	Sim
<i>Is External</i>	Booleana	Sim	Não	Não	Não	Não
<i>Fire Rating</i>	Texto	Sim	Não	Não	Não	Não
<i>Load Bearing</i>	Booleana	Sim	Sim	Não	Não	Não
<i>GEG_5D_Article</i>	Texto	Não	Sim	Não	Não	Não
<i>GEG_5D_Description</i>	Texto	Não	Sim	Não	Não	Não

3.6 LIMITAÇÕES

Apesar do potencial do fluxo de trabalho proposto, é importante destacar algumas limitações presentes neste estudo de caso. Devido ao tempo reduzido disponível para a realização da dissertação e à ausência de uma equipa de modelagem alocada exclusivamente para colaborar com a investigação, não será possível simular o processo completo com todas as suas iterações. No entanto, as etapas que serão efetivamente realizadas, como a criação do IDS, a checagem automática do modelo, a geração de *issues* e sua gestão, representam partes fundamentais do fluxo de trabalho original. Mesmo que etapas como a modelagem inicial a partir do IDS ou a retrabalho após detecção de *issues* não sejam executadas, o que será feito permite observar e avaliar a robustez, aplicabilidade e eficácia do método proposto. Isso porque essas etapas isoladas concentram os momentos mais críticos da verificação automatizada e oferecem uma base sólida para interpretar o funcionamento e os benefícios gerais do processo como um todo.

Assim, a aplicação prática do método será concentrada em quatro etapas principais: (1) a geração do ficheiro IDS com base na tabela de parâmetros fornecida pela empresa em formato Excel; (2) a verificação automática do modelo IFC entregue, utilizando o IDS gerado; (3) a exportação das não conformidades detetadas como *issues* em formato BCF; e (4) a gestão dessas *issues* no ambiente do BIMcollab. Essas etapas foram organizadas no fluxograma abaixo:

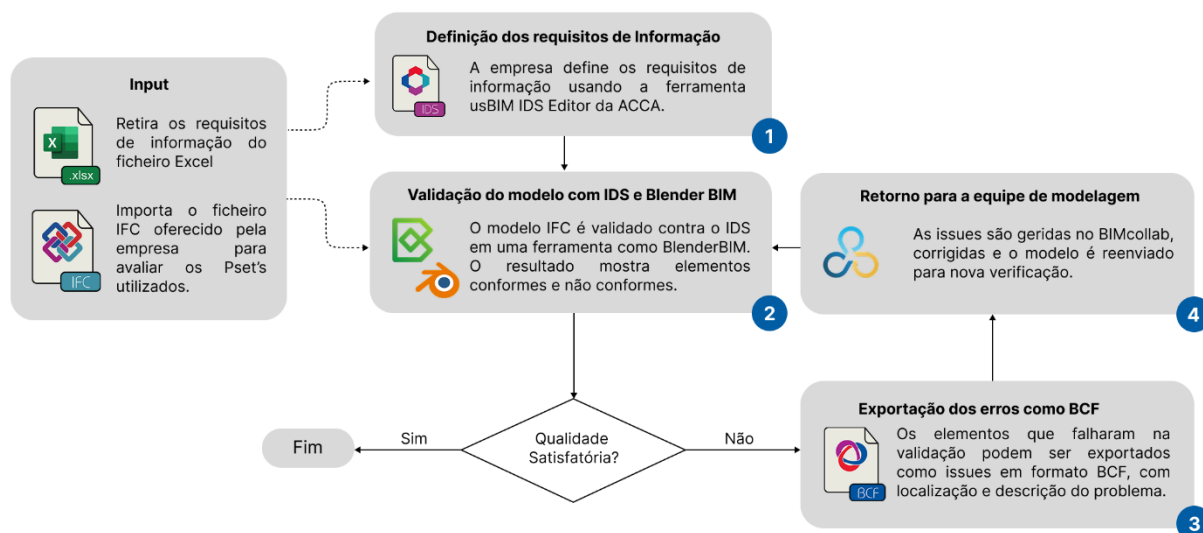


Figura 12 - Fluxograma adaptado do processo proposto

Esse recorte metodológico tem como principal objetivo analisar como a utilização de um fluxo automatizado baseado em *OpenBIM* pode beneficiar diretamente as equipas de coordenação BIM, especialmente no que diz respeito à eficiência na verificação de requisitos informacionais. A proposta busca compreender os ganhos em termos de tempo, produtividade e clareza de comunicação técnica que podem ser alcançados com a adoção de ferramentas padronizadas e interoperáveis, mesmo em um cenário com recursos limitados.

Além disso, é também um objetivo deste trabalho propor um fluxo de trabalho que seja adaptável às transformações constantes do setor, permitindo que ele evolua progressivamente até um processo 100% *OpenBIM*. Essa flexibilidade está alinhada à filosofia de que cada projeto possui suas particularidades, e, portanto, os métodos de verificação e controle da informação precisam ser dinâmicos. Ao mesmo tempo, é necessário evitar processos excessivamente maleáveis que comprometam sua estrutura e reprodutibilidade. A busca, portanto, é por um padrão que seja replicável, mas que mantenha margem para adaptação, conciliando estabilidade com adaptabilidade. Esta dissertação pretende mapear os pontos fortes do fluxo proposto, identificar suas limitações e levantar oportunidades de melhoria, oferecendo subsídios tanto para estudos futuros quanto para o desenvolvimento de novas soluções baseadas na lógica *OpenBIM*.

4

ESTUDO DE CASO

4.1 INTRODUÇÃO

Este capítulo apresenta a aplicação prática do fluxo de trabalho adaptado para a verificação de qualidade de modelos BIM, desde a criação do ficheiro IDS até a gestão das não conformidades identificadas. O estudo de caso foi conduzido com os modelos reais referidos no Capítulo 3, que serão caracterizados em maior detalhe a seguir. As próximas seções detalham cada etapa do fluxo de trabalho proposto.

4.2 CARACTERIZAÇÃO DOS MODELOS

O estudo de caso foi realizado com cinco modelos disciplinares de um mesmo projeto real, que neste trabalho não será identificado por motivos de confidencialidade. Cada modelo representa uma especialidade do edifício: arquitetura, estruturas, instalações elétricas, climatização (HVAC) e sistemas hidráulicos (*plumbing*). A opção por trabalhar com um conjunto de modelos inter-relacionados permitiu aumentar a diversidade de elementos verificados e possibilitou uma análise mais detalhada e realista das limitações do fluxo de verificação proposto. Além disso, o facto de se tratar de um modelo que já havia passado por uma verificação de qualidade manual na empresa parceira permitiu uma comparação qualitativa entre os dois métodos e a identificação de eventuais oportunidades de melhoria no processo atual.

4.2.1. MODELO ARQUITETÓNICO

O modelo arquitetónico destacou-se como um dos maiores em termos de tamanho de ficheiro, apresentando também a maior quantidade de *property sets* entre todos os modelos analisados. O nível de detalhe geométrico e informacional foi elevado, o que possibilitou a verificação de uma ampla variedade de parâmetros ao longo do processo.

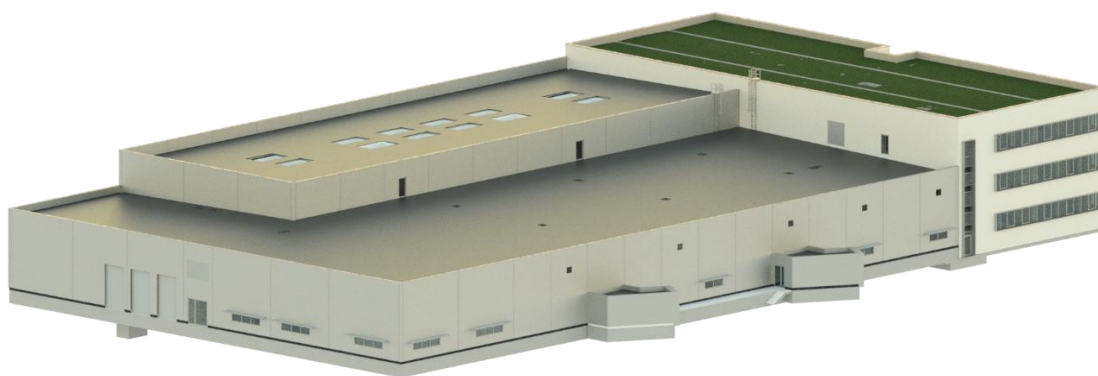


Figura 13 - Modelo Arquitetónico

4.2.2. MODELO ESTRUTURAL

O modelo estrutural apresentou uma organização informacional clara e bem estruturada. As propriedades dos elementos estavam, em geral, bem definidas, o que contribuiu para uma aplicação mais direta das regras de verificação automática.

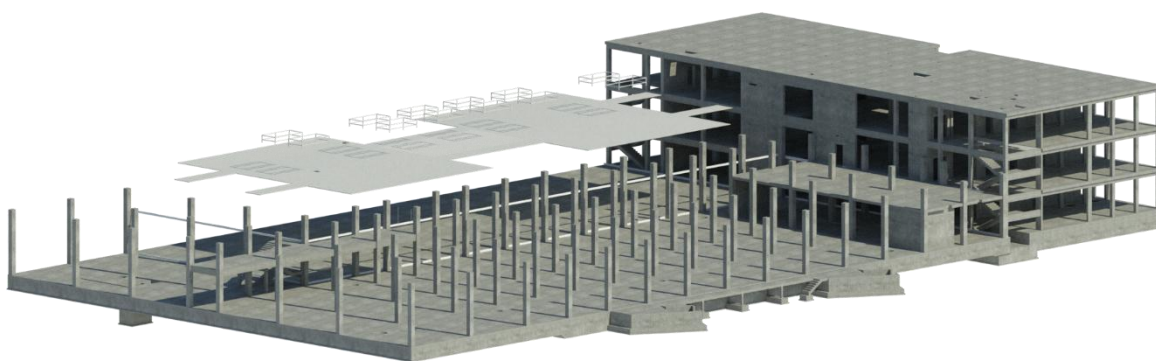


Figura 14 - Modelo Estrutural

4.2.1. MODELO ELÉTRICO

Apesar de ser um modelo detalhado em termos geométricos e de componentes, o modelo elétrico continha um número reduzido de propriedades informacionais úteis para a verificação proposta. Isso limitou o número de regras que puderam ser aplicadas no ficheiro IDS correspondente.

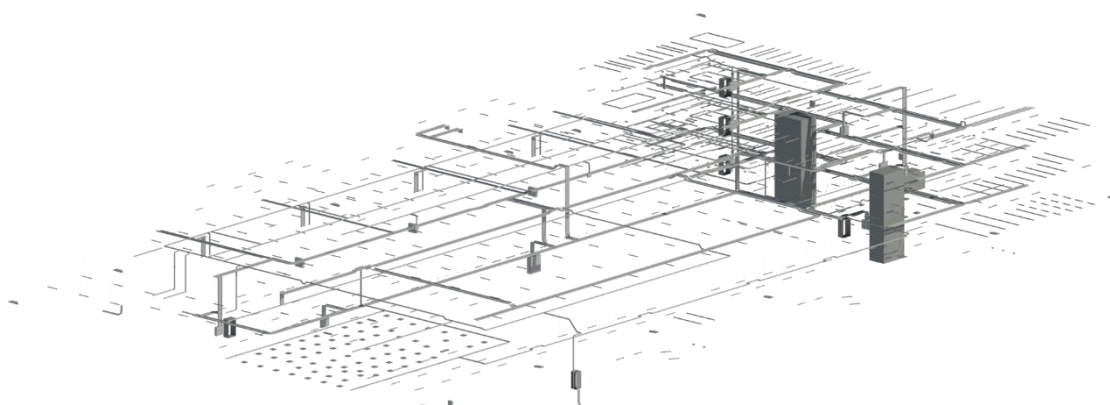


Figura 15 - Modelo Elétrico

4.2.1. MODELO HVAC

O modelo HVAC apresentou-se como o maior em termos de tamanho de ficheiro, refletindo a complexidade da disciplina e a elevada quantidade de componentes modelados. Apesar do elevado nível de detalhamento geométrico, a quantidade de *property sets* e parâmetros informacionais disponíveis foi inferior à observada no modelo arquitetónico.

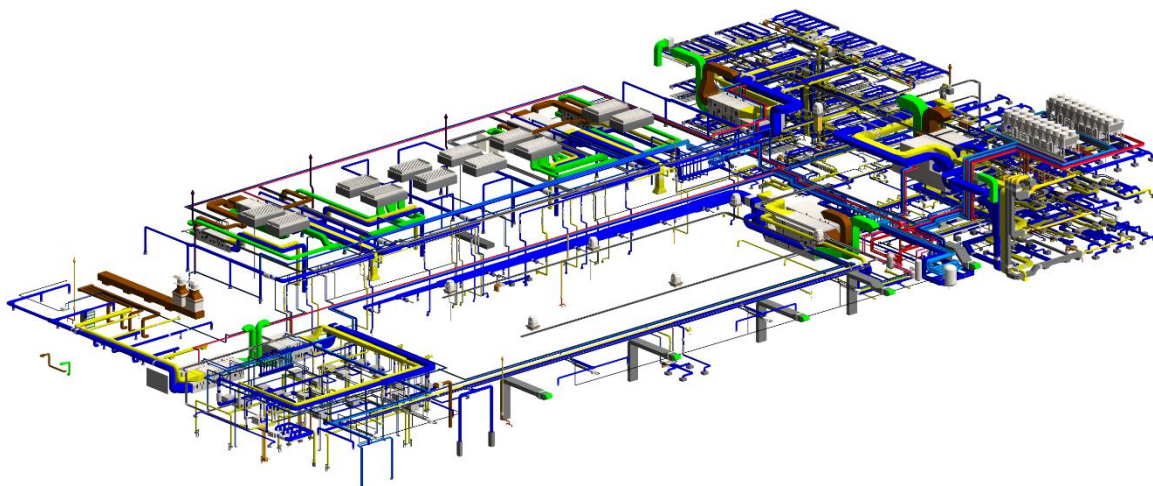


Figura 16 - Modelo HVAC

4.2.1. MODELO DE SISTEMAS HIDRÁULICOS

O modelo de sistemas hidráulicos foi o ficheiro que apresentou menor densidade de informações verificáveis. A carência de propriedades associadas aos elementos dificultou a aplicação de regras de validação e evidenciou a necessidade de maior rigor no preenchimento de dados para esta especialidade.

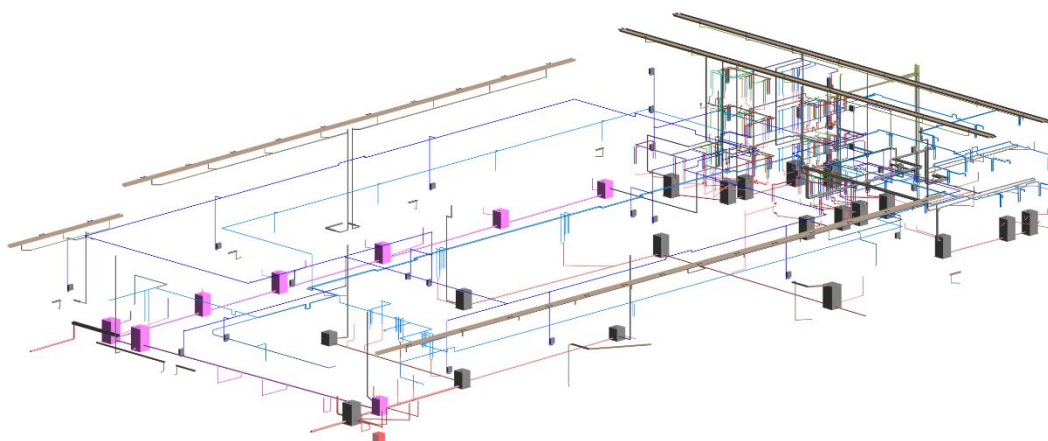


Figura 17 - Modelo de Sistemas Hidráulicos

4.3 PRÉ-VERIFICAÇÃO SINTÁTICA DOS MODELOS

Antes da aplicação de qualquer processo de verificação baseado em regras, é essencial garantir que os modelos contêm corretamente as informações mínimas necessárias para suportar esse tipo de análise. Trata-se de uma verificação preliminar, focada na estrutura sintática do modelo IFC, tal como defendido por autores como Eastman (Eastman et al., 2009), que destacam a importância de assegurar a integridade e a legibilidade computacional dos dados antes de qualquer verificação semântica ou de regras.

No caso deste estudo, essa etapa inicial permitiu confirmar que todos os modelos disponibilizados estavam estruturados de acordo com o esquema IFC 2x3 — informação relevante, já que o ficheiro IDS a ser criado posteriormente deve estar alinhado à mesma versão do esquema.

A pré-verificação também foi fundamental para identificar em quais *property sets* (Psets) se encontram as propriedades que seriam objeto de verificação. Para isso, utilizou-se a funcionalidade “*Object Information*” do Bonsai, que permite a visualização detalhada das propriedades de cada elemento do modelo. Esse procedimento foi realizado diversas vezes durante a criação do IDS, dado que a associação correta entre propriedades e seus respectivos Psets é essencial para que a verificação automática funcione adequadamente. Por motivos de confidencialidade, os *property sets* identificados nesta fase não foram explicitamente listados.

Além disso, foi criada manualmente uma lista em formato de texto com todas as classes IFC presentes em cada modelo, por meio de um recurso nativo do Blender. Essa lista serviu como referência direta na definição dos filtros no IDS, facilitando a correspondência entre os elementos do modelo e os requisitos estabelecidos.

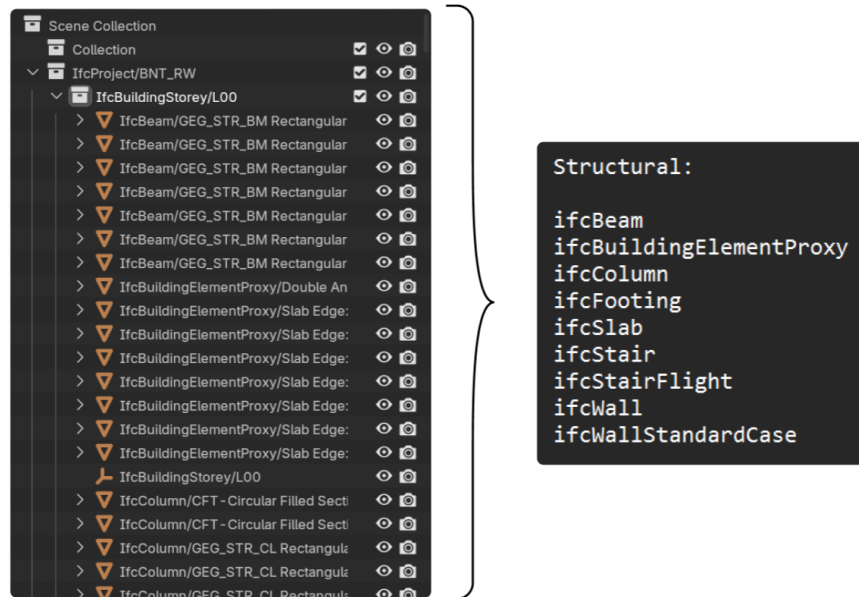


Figura 18 - Lista de elementos IFC criada a partir do Blender

Embora essa prática não seja um pré-requisito formal para a criação de um IDS, sua adoção neste estudo se mostrou útil, especialmente considerando que os modelos analisados não foram originalmente concebidos com base nesse tipo de fluxo de trabalho. A seguir, apresenta-se uma imagem com as listas de elementos extraídas de cada modelo.



Figura 19 - Lista de elementos IFC de cada especialidade

4.4 CRIAÇÃO DOS FICHEIROS IDS

O ficheiro IDS criado neste estudo, como já mencionado, tem dois objetivos principais: servir como guia de referência para as equipas modeladoras e permitir a verificação automatizada dos modelos no Blender, com o apoio do plugin Bonsai. Dentro desse contexto, o editor usBIM.IDS da ACCA desempenha o papel de uma ferramenta de tradução, convertendo os requisitos informacionais definidos previamente em uma estrutura formal compatível com os padrões *OpenBIM*.

A partir da interface da plataforma usBIM, é possível criar um ficheiro IDS do zero e preencher informações gerais como título, autor, propósito, data de criação e descrição. No âmbito deste estudo, como o foco está na validação do método proposto, foram preenchidos apenas os campos essenciais para a geração e funcionamento do ficheiro.

A construção do IDS propriamente dita é feita por meio da criação de especificações, cada uma composta por três componentes principais: identificação, filtros e requisitos.

Na aba de identificação, define-se o nome da especificação, a versão do esquema IFC a ser utilizada (IFC2x3, IFC4 ou IFC4x3), além de descrições e instruções complementares. Para este estudo, foi selecionado o esquema IFC2x3, conforme identificado na etapa de pré-verificação sintática.

Wall

General Filters Requirements

Name	Wall	Description	
Instructions		Identifier	
IFC versions	IFC2X3		

Figura 20 - Aba identificação do editor IDS

A aba de filtros permite ao utilizador seleccionar quais classes de elementos devem ser avaliadas no modelo. É possível indicar se o modelo deve, pode ou não deve conter determinados elementos. Neste trabalho, os filtros foram definidos com base na lista de classes IFC extraída previamente no Blender.

Wall

General Filters Requirements

☐ The model MUST contain entities that have... +

☒	1	IFC class IFCWALL (or: IFCWALLSTANDARDCASE)	☐	☐
-------------------------------------	---	---	--------------------------	--------------------------

Filter by IFC class

IFC class Name

with predefined type Name

Figura 21 - Aba filtros do editor IDS

Por fim, a aba de requisitos permite indicar quais propriedades os elementos filtrados devem conter, especificando também em qual *property set* essas propriedades devem estar organizadas. É possível definir não apenas os nomes das propriedades, mas também os tipos de dados esperados (como *IfcBoolean*, *IfcText*, *IfcDate*, entre outros), além de valores esperados quando aplicável.

Wall

General Filters Requirements

	that MEET the following requirements	+
☐	1 MUST HAVE property GEG_SD_Article of PSet Identity Data	
☐	2 MUST HAVE property GEG_SD_Description of PSet Identity Data	
☒	3 MUST HAVE property LoadBearing of PSet Pset_WallCommon (IFCBOOLEAN)	

Requirement for property

mode MUST HAVE...

property Name LoadBearing

of Pset Name Pset_WallCommon

data type IFCBOOLEAN

with value Boolean + Field NOT REQUIRED

Instructions Field NOT REQUIRED

Uri Field NOT REQUIRED

Figura 22 - Aba requisitos do editor IDS

Vale destacar que o editor permite a criação de múltiplas especificações dentro de um mesmo ficheiro IDS, e que cada especificação pode conter diversos filtros e requisitos. Neste estudo, optou-se por organizar uma especificação para cada tipo de elemento, prática que se mostrou útil para facilitar a colaboração entre equipas durante o processo de verificação. Essa decisão será retomada no Capítulo 5, onde se analisa o impacto dessa escolha na clareza e eficiência do processo.

Uma funcionalidade relevante identificada durante o uso do editor foi a possibilidade de duplicar especificações e requisitos, bem como o próprio ficheiro IDS. Essa função se revelou especialmente útil em contextos onde diferentes modelos compartilham estruturas semelhantes de informação, permitindo maior agilidade na construção dos documentos e contribuindo para a escalabilidade do método. No presente estudo, essa funcionalidade foi empregue para replicar especificações, alterando apenas os filtros conforme necessário.

Antes da conclusão do ficheiro IDS, foi utilizado o recurso de verificação automática do próprio editor, chamado *Audit Tool*. Essa funcionalidade realiza uma validação do próprio ficheiro IDS, identificando erros de estrutura como propriedades mal escritas, *property sets* inválidos ou especificações incompletas, o que ajuda a evitar falhas futuras quando o ficheiro for utilizado para a verificação dos modelos no Blender.

No caso deste estudo, a ferramenta acusou inicialmente um erro relacionado à grafia da propriedade *Load Bearing*, que precisou ser corrigida para *LoadBearing*, conforme o padrão reconhecido pelo esquema IFC 2x3. A figura a seguir mostra o aviso de erro identificado pela ferramenta.

The screenshot shows the 'Requirement for property' configuration window with the following settings:

- mode: MUST HAVE...
- property: Name (Load Bearing)
- of Pset: Name (Pset_WallCommon)
- data type: IFCBOOLEAN
- with value: Boolean (Field NOT REQUIRED)
- Instructions: Field NOT REQUIRED
- Uri: Field NOT REQUIRED

Below the configuration window, the 'Invalid document' error message is displayed:

Invalid document (IDS Audit Tool v1.0.81)

Specification	Code	Description	Line	Pos.
Wall	103	Invalid property baseName 'Load Bearing' (10 accepted values exist, starting with AcousticRating, Combustible, Compartmentation, ExtendToStructure, FireRating...) in the context of Ifc2x3 on 'simpleValue'.	235	14

Figura 23 - Audit tool acusa erro de sintaxe

Após a correção, o ficheiro foi validado com sucesso, como demonstrado na imagem abaixo.

The screenshot shows the 'Requirement for property' configuration window with the following settings:

- mode: MUST HAVE...
- property: Name (LoadBearing)
- of Pset: Name (Pset_WallCommon)
- data type: IFCBOOLEAN
- with value: Boolean (Field NOT REQUIRED)
- Instructions: Field NOT REQUIRED
- Uri: Field NOT REQUIRED

Below the configuration window, the 'Valid document' success message is displayed:

Valid document (IDS Audit Tool v1.0.81)

Specification	Code	Description	Line	Pos.
---------------	------	-------------	------	------

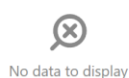


Figura 24 - Audit tool não acusa nenhum erro de sintaxe

Essa etapa, apesar de simples, se mostrou essencial para garantir a consistência técnica do ficheiro IDS e evitar falhas na etapa seguinte do processo de verificação.

4.5 FICHEIROS XML E PDF CRIADOS

Com todas as especificações devidamente preenchidas, o editor usBIM.IDS permite a geração de dois ficheiros distintos: um ficheiro em formato XML, voltado para leitura por máquinas, e um ficheiro em PDF, voltado para leitura humana. A Figura 25 apresenta ambos os ficheiros gerados a partir de uma especificação aplicada a elementos do tipo *IfcWall*: na parte superior, o ficheiro em formato PDF; na parte inferior, o ficheiro correspondente em formato XML.

PDF

Wall

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCWALL (or: IFCWALLSTANDARDCASE)**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **LoadBearing** of PSet **Pset_WallCommon (IFCBOOLEAN)**

XML

```

202 <ids:specification ifcVersion="IFC2X3" name="Wall">
203   <ids:applicability minOccurs="1" maxOccurs="unbounded">
204     <ids:entity>
205       <ids:name>
206         <xs:restriction base="xs:string">
207           <xs:enumeration value="IFCWALL" />
208           <xs:enumeration value="IFCWALLSTANDARDCASE" />
209         </xs:restriction>
210       </ids:name>
211     </ids:entity>
212   </ids:applicability>
213   <ids:requirements>
214     <ids:property cardinality="required">
215       <ids:propertySet>
216         <ids:simpleValue>Identity Data</ids:simpleValue>
217       </ids:propertySet>
218       <ids:baseName>
219         <ids:simpleValue>GEG_5D_Article</ids:simpleValue>
220       </ids:baseName>
221     </ids:property>
222     <ids:property cardinality="required">
223       <ids:propertySet>
224         <ids:simpleValue>Identity Data</ids:simpleValue>
225       </ids:propertySet>
226       <ids:baseName>
227         <ids:simpleValue>GEG_5D_Description</ids:simpleValue>
228       </ids:baseName>
229     </ids:property>
230     <ids:property dataType="IFCBOOLEAN" cardinality="required">
231       <ids:propertySet>
232         <ids:simpleValue>Pset_WallCommon</ids:simpleValue>
233       </ids:propertySet>
234       <ids:baseName>
235         <ids:simpleValue>LoadBearing</ids:simpleValue>
236       </ids:baseName>
237     </ids:property>
238   </ids:requirements>
239 </ids:specification>

```

Figura 25 - Exemplo de ficheiro PDF e XML a partir do IDS

Em um fluxo de trabalho completo, o ficheiro IDS seria elaborado antes do início da modelação e utilizado para orientar a equipa modeladora quanto às propriedades e estruturas de dados exigidas. Nesse caso, o ficheiro PDF serviria como um guia de consulta, e o ficheiro XML poderia ser utilizado para realizar uma verificação interna prévia, evitando retrabalhos e iterações desnecessárias com a equipa de coordenação.

No entanto, considerando que este estudo se propõe a validar um processo adaptado, com modelos já prontos e requisitos definidos *a posteriori*, o ficheiro IDS gerado foi utilizado exclusivamente na etapa de verificação automática. Para isso, o ficheiro XML foi exportado na versão 1.0, compatível com a ferramenta de verificação utilizada no Blender.

4.6 PROCESSO DE VERIFICAÇÃO AUTOMÁTICA

A verificação automática dos modelos foi conduzida por meio da funcionalidade *IFC Tester*, disponível no *plugin* Bonsai, integrado ao Blender. Nesta etapa, cada modelo foi aberto individualmente no software, e, para cada um deles, foi carregado o ficheiro IDS correspondente previamente gerado.

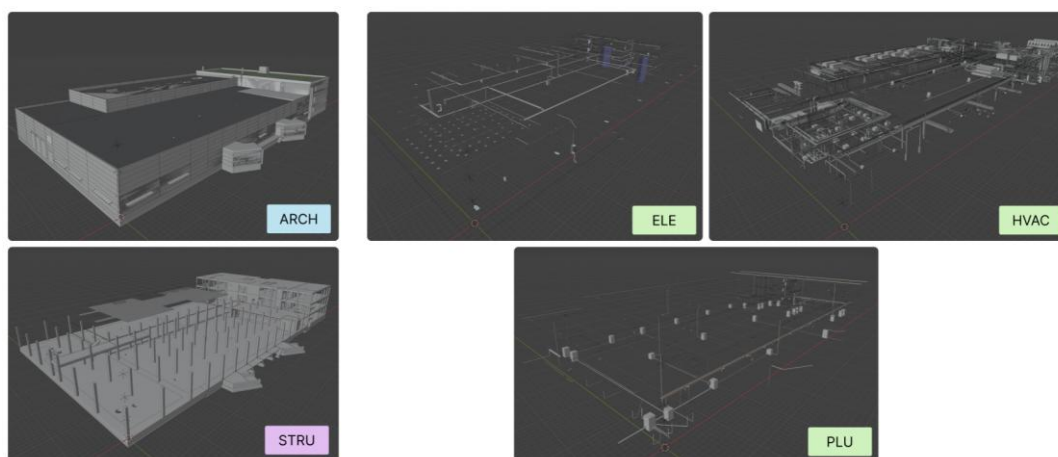


Figura 26 - Modelos de cada especialidade abertos no Blender

A partir da aba “Qualidade e Coordenação”, o *IFC Tester* permite a seleção do modelo IFC, do ficheiro IDS e da funcionalidade de geração automática de um relatório em formato HTML. Após a configuração, a verificação é executada com um único comando, e o relatório é gerado e aberto automaticamente em um navegador predefinido pelo sistema.

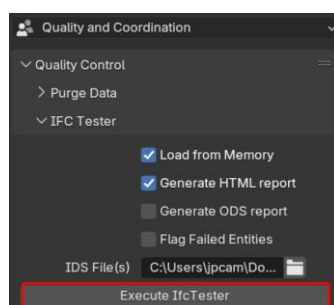


Figura 27 - IFC Tester

Esse relatório apresenta os resultados de todas as especificações aplicadas, organizadas por tipo de elemento, e permite observar quais requisitos foram atendidos e quais falharam. Os resultados serão analisados com maior profundidade no Capítulo 5.

A Figura 28 apresenta um excerto do relatório HTML gerado. Já nesta fase é possível perceber como a separação das especificações por tipo de elemento contribui para a clareza na interpretação dos resultados. Essa prática oferece indicações relevantes sobre quais modelos estão mais consistentes em termos de qualidade informacional e quais exigem maior atenção no processo de modelação.

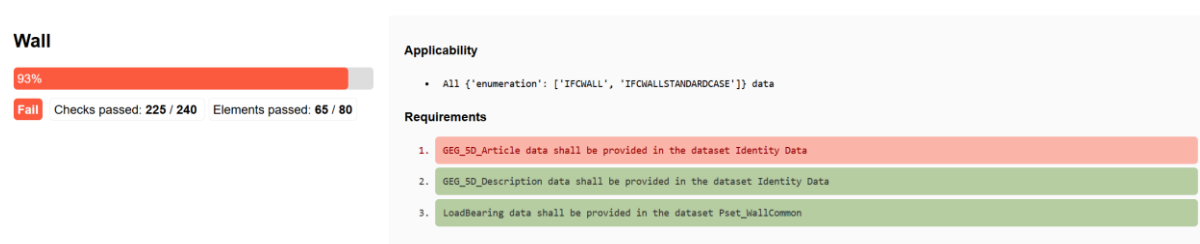


Figura 28 - Excerto do relatório HTML

Além do relatório externo, o próprio Blender apresenta um painel interno com um resumo das verificações realizadas. Nesse painel, é possível visualizar, em tempo real, quais especificações falharam, quais requisitos não foram atendidos, e selecionar diretamente os elementos envolvidos para localizá-los dentro do modelo.

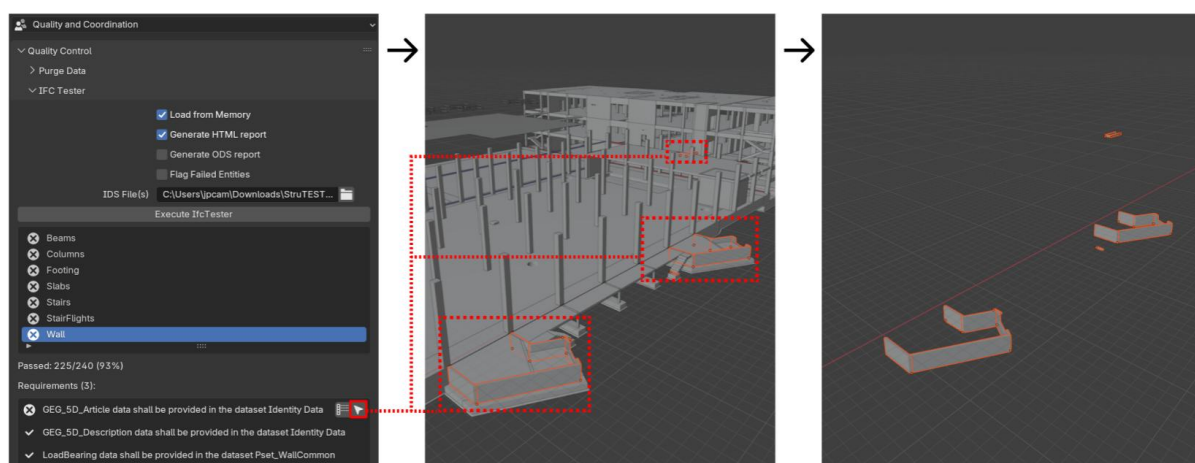


Figura 29 - Processo para visualizar as não-conformidades detetadas

A seleção dos elementos não conformes permitiu isolar visualmente esses objetos e criar vistas específicas com base nesses subconjuntos. Essas vistas foram utilizadas na etapa seguinte, para a criação dos ficheiros BCF, como forma de contextualizar graficamente cada não conformidade identificada.

4.7 GESTÃO DE NÃO-CONFORMIDADES

Após a verificação automática no Blender, o próximo passo consistiu na gestão das não conformidades identificadas. Para isso, foi utilizada a funcionalidade *BCF Project*, localizada na aba “Colaboração” do Bonsai, que permite a criação de *issues* diretamente associadas aos elementos que não atenderam aos requisitos estabelecidos.

Cada *issue* foi gerada manualmente, com base nas informações apresentadas no relatório HTML e nas visualizações com os elementos isolados criadas na etapa anterior. Embora o processo de criação seja relativamente simples, o fato de cada *issue* precisar ser criada individualmente torna essa etapa um ponto potencial de melhoria, especialmente em projetos de grande escala. A possibilidade de automatizar esse processo e estabelecer boas práticas específicas será discutida como sugestão para trabalhos futuros.

Ao criar um projeto BCF, foi selecionada a versão 2.1 do formato, uma vez que a versão 3.0 ainda apresenta limitações de compatibilidade com o BIMcollab Zoom.

Durante a criação das *issues*, é possível preencher diversos campos, como nome, descrição, tipo (por exemplo, erro, sugestão ou observação), prioridade, responsável, e a data-limite para resolução. Esta última deve seguir o formato:

YYYY-MM-DDT00:00:00

Exemplo: 2025-05-01T00:00:00

Uma prática adotada neste estudo foi a de reutilizar as mensagens do relatório HTML como descrição da *issue*, o que permite ao modelador compreender rapidamente o motivo da não conformidade ao visualizar a vista associada.

Ao selecionar previamente a vista com os elementos não conformes, o utilizador pode, com um único clique, vinculá-la à *issue* criada. A Figura 30 a seguir apresenta um exemplo de preenchimento completo de um projeto BCF.

BCF Viewpoints	
Type	Issue
Status	Active
Priority	Critical
Stage	Design
Assigned To	joacampelo@feup.com
Due Date	2025-05-01T00:00:00
Date	2025-04-23T13:13:58.862750
Author	john@doe.com

Figura 30 - Projeto BCF preenchido

Antes de exportar o ficheiro, é necessário nomeá-lo manualmente e adicionar a extensão “.bcf”, pois o software não a insere automaticamente.

Uma vez exportado, o ficheiro BCF pode ser importado diretamente no BIMcollab Zoom. A ferramenta reconhece automaticamente as propriedades atribuídas às *issues*, como descrições, tipos e visualizações, permitindo que a equipa de coordenação aceda às não conformidades de forma clara e organizada.

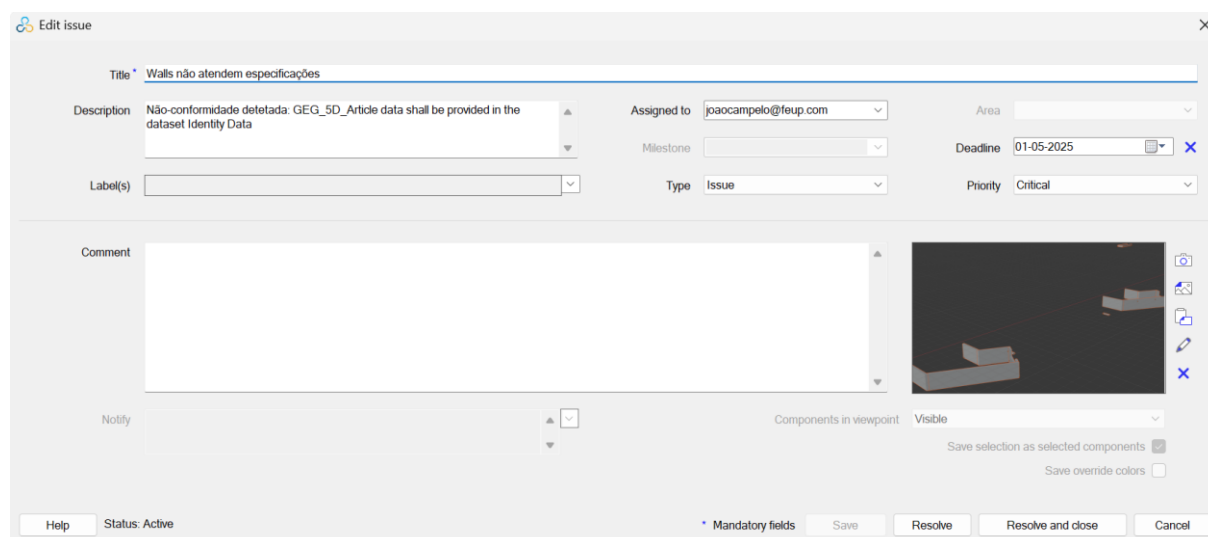


Figura 31 - *Issue* criado no Blender e importado no BimCollab Zoom

Em um fluxo de trabalho completo, a partir dessa etapa, as *issues* seriam comunicadas à equipa modeladora, que realizaria os ajustes no modelo, geraria um novo ficheiro IFC e o reenviaria à equipa de coordenação, dando início a uma nova iteração do processo de verificação.

No entanto, considerando os objetivos e as limitações deste estudo, o ciclo se encerra neste ponto. Ainda assim, foi possível percorrer as etapas mais relevantes do processo e avaliar a sua viabilidade prática, estabelecendo uma base concreta para discussões futuras.

A partir das alterações introduzidas no fluxo de verificação, o capítulo seguinte apresenta uma análise dos resultados obtidos, discutindo os impactos dessas modificações, bem como as limitações identificadas ao longo do estudo.

5

ANÁLISE DOS RESULTADOS

5.1 INTRODUÇÃO

Este capítulo apresenta a análise dos resultados obtidos com a aplicação prática do fluxo de verificação proposto, conforme descrito no capítulo 4. O objetivo é avaliar a eficácia do método em termos de clareza na comunicação dos requisitos, facilidade de execução, identificação de não conformidades e potencial de integração em fluxos reais de coordenação de modelos.

Embora o estudo tenha sido realizado com um fluxo adaptado, sem iterações completas entre modelação e verificação, foi possível observar de forma concreta os benefícios e limitações da abordagem. A análise a seguir discute os principais pontos observados durante a execução das etapas, com base nos relatórios gerados, nos *issues* identificados e na experiência prática de utilização das ferramentas envolvidas.

5.2 FICHEIROS IDS CRIADOS

Os parâmetros verificados foram definidos com base nas listas fornecidas pela empresa parceira e estão sintetizados na Tabela 4 - Síntese dos parâmetros a serem avaliados por modelo. Cada ficheiro foi estruturado e testado individualmente, sendo analisadas a facilidade de implementação das regras, os desafios técnicos encontrados e o grau de preenchimento dos dados nos modelos.

Tabela 4 - Síntese dos parâmetros a serem avaliados por modelo

Modelos	Parâmetros
Arquitetónico (<i>Architectural</i>)	<i>Name, Material Name, Building Story, Load Bearing, IsExternal, Fire Rating</i>
Estrutural (<i>Structural</i>)	<i>Name, Material Name, Building Story, Load Bearing, GEG_5D_Article, GEG_5D_Description</i>
MEP (<i>Electrical, HVAC and Plumbing</i>)	<i>Name, Material Name, Building Story, System Type</i>

Um dos principais desafios comuns a todos os modelos na criação dos ficheiros IDS foi a ausência de um parâmetro padronizado para indicar o piso ou compartimento espacial ao qual os elementos pertenciam. O parâmetro *Building Story*, embora incluído nas listas de requisitos, não existia diretamente

nos modelos. Como alternativa, foram utilizadas propriedades distintas presentes nos *property sets* dos próprios elementos, como “*Reference Level*”, “*Base Level*”, “*Level*” ou “*Work Plane*”, com variações significativas entre disciplinas e tipos de objeto. Objetos do tipo *Stair flights* e fundações, por exemplo, não possuíam parâmetros informacionais relacionados à posição vertical, o que exigiu uma análise manual do conteúdo informacional de cada tipo de elemento.

Esse problema foi particularmente evidente nos modelos estrutural e arquitetônico, nos quais, mesmo dentro de um mesmo tipo IFC, havia inconsistência na nomenclatura dos parâmetros utilizados para indicar a localização. No modelo elétrico, a ausência total de informações sobre localização limitou a verificação a propriedades básicas, como nome e tipo de sistema, sem qualquer possibilidade de análise contextual. Em contraste, o modelo HVAC apresentou maior variedade de *property sets*, permitindo uma configuração de regras mais completa, ainda que com nomenclaturas igualmente irregulares. O modelo de sistemas hidráulicos, por sua vez, apresentou baixa densidade de dados, com poucos parâmetros informacionais preenchidos, o que restringiu a verificação a atributos mínimos.

Ainda assim, os ficheiros IDS foram implementados com sucesso para os parâmetros que estavam presentes e definidos de forma consistente nos modelos. Elementos classificados como *IfcBuildingElementProxy* foram excluídos da lista de verificação, pois não apresentavam propriedades comuns ou comportamento padronizado no modelo. Segundo a definição oficial da *buildingSMART*:

"The IfcBuildingElementProxy is a proxy definition that provides the same functionality as an IfcBuildingElement, but without having a defined meaning of the special type of building element it represents."

Ainda segundo a mesma fonte:

"It should be used to exchange special types of building elements for which the current IFC Release does not yet provide a semantic definition, or for which the participating applications cannot provide additional semantic classification."

Por fim, vale destacar que, embora os parâmetros de piso não tenham sido verificados diretamente, o *Blender* foi capaz de identificar corretamente o *spatial container* dos elementos — metadado que associa o objeto a uma entidade espacial, como um piso ou zona. A Figura 32 ilustra um exemplo dessa identificação automática, com a distinção clara dos *spatial containers* atribuídos aos diferentes elementos do modelo. No editor *usBIM.IDS*, essa informação pode ser utilizada como critério de verificação, por meio da filtragem dos requisitos com base na relação “part of”. Essa possibilidade amplia o potencial de verificação contextual em fluxos baseados em *OpenBIM*, permitindo validar a correta associação dos elementos às suas respectivas entidades espaciais.

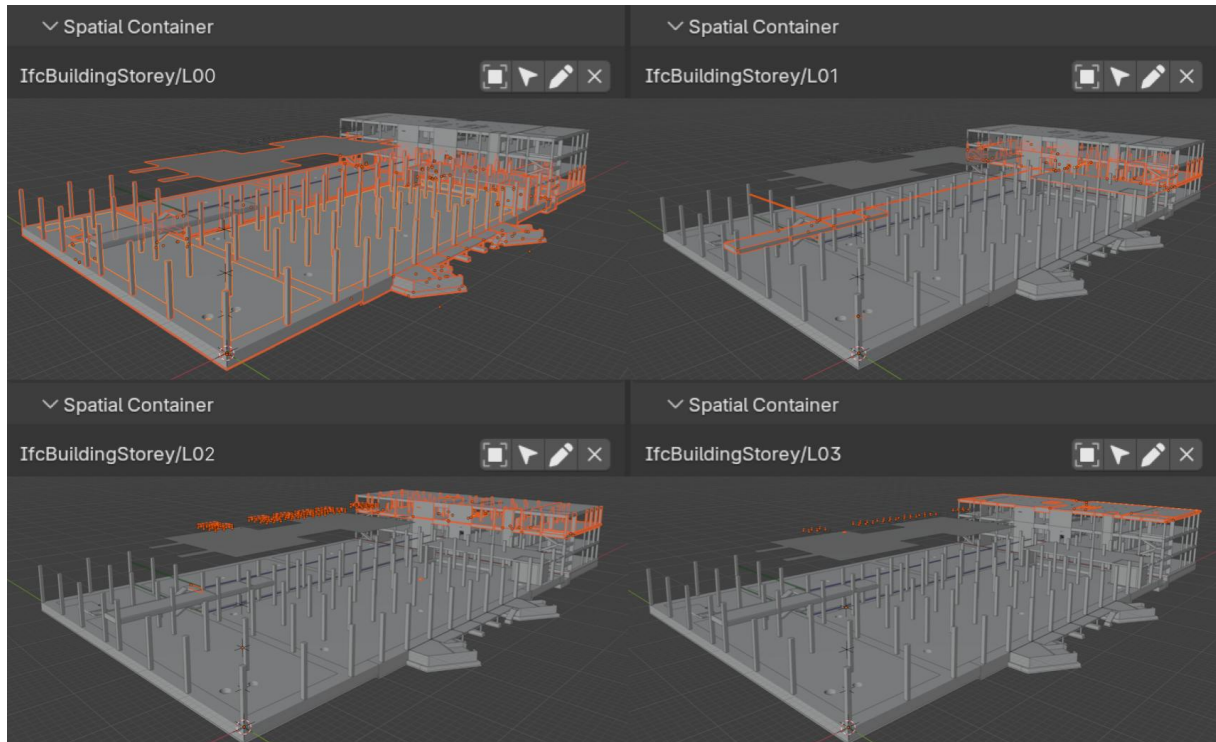


Figura 32 - Demonstração da distinção dos *Spatial Containers* no Blender

Todos os ficheiros PDF e XML gerados pelo ficheiro IDS podem ser encontrados no Anexo A1 e A2 ao final da presente dissertação.

5.3 RESULTADOS DA VERIFICAÇÃO AUTOMÁTICA

A verificação automática dos modelos retornou uma grande quantidade de dados, permitindo avaliar a completude informacional dos elementos com rapidez e clareza. No modelo estrutural, por exemplo, foram realizadas mais de 5.000 checagens instantâneas, o que evidencia o ganho de eficiência em relação ao processo tradicional manual.

Uma tabela-resumo com os resultados por modelo será apresentada na Tabela 5, incluindo a quantidade de elementos verificados e a taxa de conformidade geral em cada caso.

Tabela 5 - Síntese dos elementos verificados

Modelos	Elementos verificados	Elementos em conformidade
Arquitetónico	7133	6530
Estrutural (<i>Structural</i>)	882	447
Elétrico	1978	1350
HVAC	40339	38170
Sistemas Hidráulicos	7929	7929

A título de exemplo, observou-se que a maioria das *IfcBeams*, ou vigas, do modelo estrutural, estava corretamente preenchida. No entanto, as não conformidades remanescentes estavam distribuídas de forma aleatória no modelo, sem concentração por zona ou pavimento. Esse padrão sugere que os erros têm origem em falhas humanas pontuais no momento da modelação, o que reforça o potencial da verificação automatizada não apenas como apoio, mas como eventual substituto ao controle manual.

No modelo elétrico, foi possível identificar elementos com campos preenchidos com o texto “*user to populate*” — uma indicação clara de que o preenchimento desses parâmetros foi delegado ao utilizador antes da exportação IFC, mas acabou não sendo concluído. Situações como essa mostram como a verificação automatizada pode detetar falhas rotineiras que passariam despercebidas numa revisão visual tradicional.

Esse comportamento repetiu-se em outros tipos de elementos, como *IfcColumn*, em que as não conformidades estavam frequentemente associadas a estruturas pequenas ou secundárias que, embora classificadas corretamente, não exigiam todos os parâmetros previstos para os elementos principais. Isso evidencia o caráter interpretativo dos relatórios: apesar de identificarem lacunas objetivas, cabe à equipa de coordenação decidir o que pode ou não ser desconsiderado com base no contexto do projeto. A automatização, nesse caso, não substitui o juízo técnico, mas oferece uma triagem precisa e rastreável.

No modelo HVAC, a variedade e qualidade dos *property sets* permitiram uma verificação mais robusta. Já o modelo de sistemas hidráulicos apresentou baixos níveis de informação, o que limitou as possibilidades de checagem. A ferramenta operou normalmente, mas os resultados refletiram diretamente a ausência de dados no modelo.

Uma prática adotada que se revelou fundamental para a qualidade do processo foi a separação das especificações do ficheiro IDS por classes IFC específicas, em vez de criar uma única regra genérica para todos os elementos. Embora fosse tecnicamente possível filtrar todas as classes ao mesmo tempo para verificar a ausência de um determinado parâmetro, essa abordagem teria tornado a interpretação dos resultados excessivamente complexa e suscetível a confusões. Como já discutido, o relatório gerado possui caráter interpretativo e exige contextualização: muitos elementos pertencem à mesma classe IFC, mas exercem funções distintas no projeto e, portanto, não necessitam do mesmo nível de preenchimento informacional. Para que essa interpretação ocorra de forma segura e eficiente, o volume de dados a ser analisado precisa ser compatível com a capacidade humana de triagem e tomada de decisão. Ao agrupar todos os elementos numa única verificação, qualquer ausência de parâmetro em qualquer objeto geraria alertas generalizados, resultando num relatório poluído e difícil de analisar. Além disso, na exportação dos erros para *issues*, a presença de múltiplas classes misturadas num único *viewpoint* poderia dificultar significativamente o trabalho do modelador responsável pela correção, aumentando o tempo necessário para localizar e resolver as não conformidades. A Figura 33 ilustra a diferença prática entre uma verificação organizada por classes e uma verificação genérica sem essa separação.

Sem criar uma especificação para cada classe:

The model MUST contain entities that have

- IFC class **IFCBEAM** (or: **IFCWALL**; **IFCWALLSTANDARDCASE**; **IFCCOLUMN**; **IFCSLAB**; **IFCSTAIR**; **IFCSTAIRFLIGHT**; **IFCFOOTING**)

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**

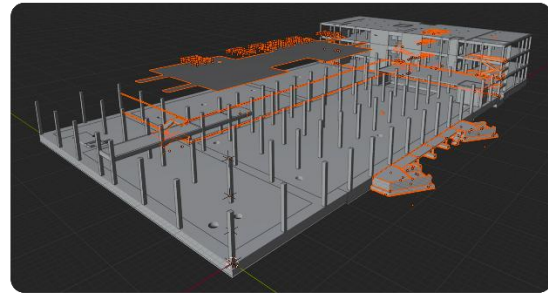
All IFC Classes need GEG_5D_Article

52%

Fail

Checks passed: 463 / 882

Elements passed: 463 / 882



A criar uma especificação para cada classe:

The model MUST contain entities that have

- IFC class **IFCBEAM**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**

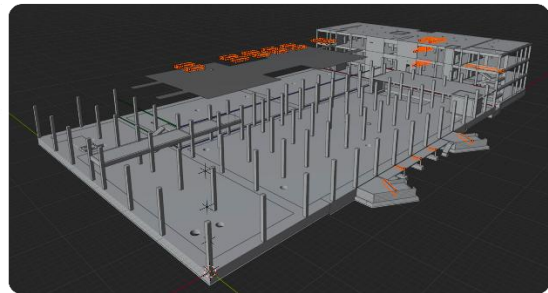
Beams

75%

Fail

Checks passed: 929 / 1224

Elements passed: 54 / 204



The model MUST contain entities that have

- IFC class **IFCCOLUMN**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**

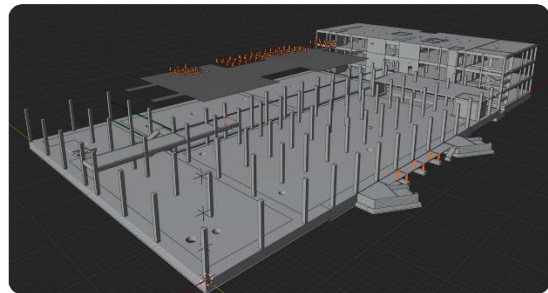
Columns

88%

Fail

Checks passed: 2342 / 2640

Elements passed: 288 / 440



The model MUST contain entities that have

- IFC class **IFCFOOTING**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**

Footing

75%

Fail

Checks passed: 36 / 48

Elements passed: 0 / 12

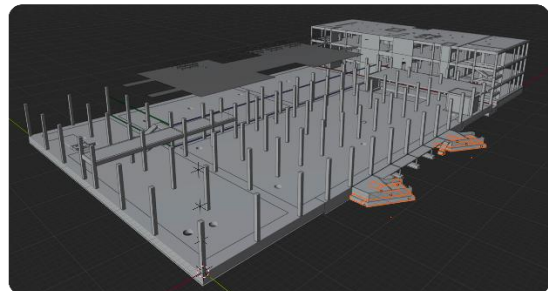


Figura 33 - Comparação de práticas de criação de especificações

De modo geral, o processo de verificação correu de forma fluida nas disciplinas em que os parâmetros estavam bem estruturados. A exportação dos relatórios HTML foi realizada sem dificuldades técnicas, e os resultados apresentados pelo Bonsai demonstraram boa organização e legibilidade, mesmo em modelos com grande volume de dados.

Todos os relatórios gerados, bem como os ficheiros IDS utilizados nesta fase, encontram-se disponíveis para consulta na seção de anexos ao final desta dissertação.

5.4 GESTÃO DE NÃO CONFORMIDADES

A criação de *issues* em formato BCF (*BIM Collaboration Format*) a partir do Blender, utilizando o plugin Bonsai, revelou-se funcional, porém marcada por limitações técnicas que exigem atenção por parte da equipa coordenadora. Esta etapa, que visa permitir a gestão colaborativa das não conformidades identificadas, foi a mais demorada de todo o processo.

A geração de *viewpoints* no Bonsai ocorre de forma relativamente simples, permitindo que o coordenador selecione visualmente os elementos não conformes e crie uma anotação associada. No entanto, o preenchimento dos campos de metadados — como tipo, prioridade, responsável, status e data-limite — é inteiramente manual. Isso exige que o utilizador tenha familiaridade com a plataforma de gestão de *issues* que será utilizada, neste caso o *BIMcollab Zoom*. Por exemplo, no campo “*Type*”, o coordenador deve escrever termos que estejam de acordo com as convenções do *BIMcollab*, como *issue*, *clash*, *failure*, *remark*, entre outros. O Bonsai não oferece sugestões automáticas nem menus suspensos, o que pode levar a erros de digitação ou inconsistências na categorização. A Figura 34 ilustra esse ponto. Essa limitação se estende a todos os demais campos de preenchimento, como *Status*, *Priority* ou *Label*, que também não oferecem validação ou preenchimento assistido.

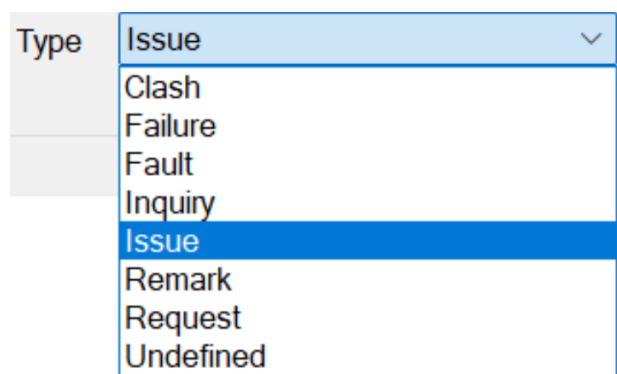


Figura 34 - Menu suspenso do BIMCollab Zoom

Outra limitação observada diz respeito ao campo “*Assign To*”, que pode ser preenchido diretamente no Bonsai. No entanto, esse preenchimento não aciona notificações automáticas aos responsáveis, como seria de esperar num fluxo contínuo de coordenação. A atribuição deve ser posteriormente complementada dentro do ambiente *BIMcollab*, exigindo um duplo trabalho e maior vigilância da equipa de coordenação.

A definição da data-limite (*Due Date*) para resolução de *issues* também apresentou dificuldades. O campo requer um formato muito específico, conforme descrito no Capítulo 4, o que o torna suscetível a erros humanos. A ausência de uma interface mais intuitiva — como um seletor de data — compromete

a usabilidade e pode limitar a adoção dessa funcionalidade. A Figura 35 sugere uma possível melhoria para esse ponto.

Type	Issue	✕
Status	Active	✕
Priority	Critical	✕
Stage		
Assigned To	john@doe.com	✕
Due Date	00:00:00	📅

00:00:00

maio de 2025

D	S	T	Q	Q	S	S
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Figura 35 - Sugestão de alteração da interface de usuário do Bonsai.

O processo de exportação dos projetos BCF exigiu cuidados adicionais. Primeiramente, foi necessário adicionar manualmente a extensão “.bcf” ao nome do ficheiro no momento da exportação, caso contrário, o arquivo gerado não era reconhecido pelo *BIMcollab*. Além disso, foi identificado um conflito de versão com a atualização do Blender para a versão 4.4 (março de 2025), que causava o encerramento abrupto do programa e a perda dos dados criados. Este erro não foi isolado e foi relatado por outros utilizadores da ferramenta, conforme registado em fóruns da comunidade.

Quando a exportação era concluída com sucesso, os *issues* foram corretamente importados no *BIMcollab Zoom*, com todos os campos preenchidos adequadamente, sem perdas de dados. Os tópicos criados puderam ser filtrados, visualizados e organizados de acordo com os critérios da equipa coordenadora.

Após a importação no *BIMcollab*, os *issues* também ficaram acessíveis dentro de ferramentas autorais, como o *Revit*, por meio do *BCF Manager*. Isso permite que, numa situação real, uma equipa modeladora possa localizar com precisão os elementos a corrigir, fechando o ciclo de identificação e resolução de não conformidades.

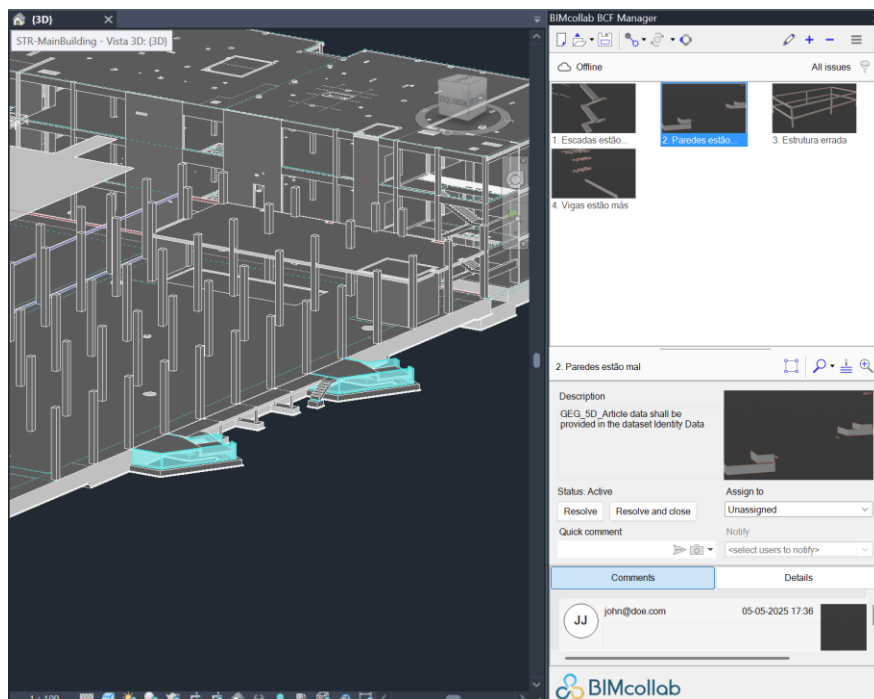


Figura 36 - Visualização de issues em ferramenta autoral

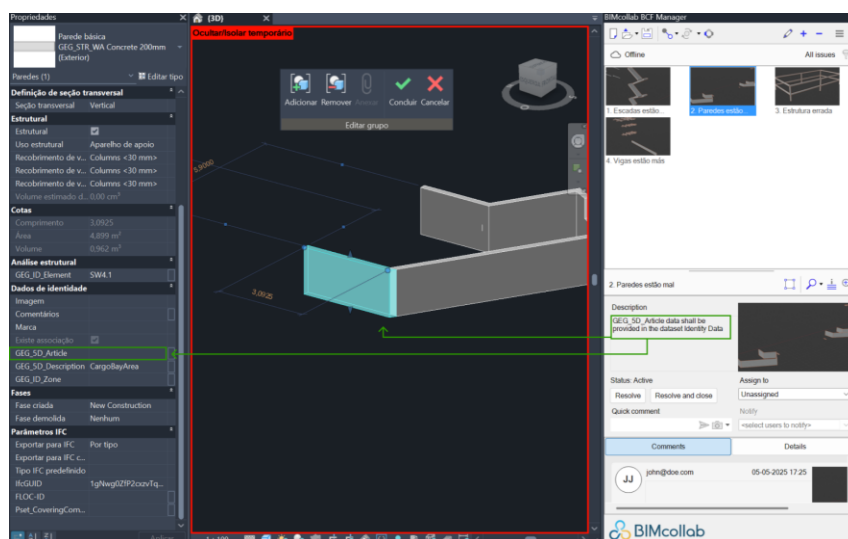


Figura 37 - Visualização de issues em ferramenta autoral

Uma funcionalidade valorizada em plataformas de gestão de *issues* como o BIMcollab é a capacidade de manter o histórico detalhado de cada *issue*, permitindo o acompanhamento da sua evolução com comentários, capturas de tela, datas e responsáveis ao longo do tempo. Embora o Blender permita a criação de comentários e *viewpoints* associados a elementos do modelo, a manutenção desse histórico dentro do fluxo Blender + Bonsai depende da constante substituição manual de ficheiros atualizados, o que aumenta significativamente o risco de perda de informação ou confusão na gestão documental. Essa limitação pode comprometer a rastreabilidade e a organização interna da equipa, caso não sejam adotados procedimentos de controle muito bem definidos.

6

CONCLUSÕES

6.1 CONSIDERAÇÕES INICIAIS

O presente trabalho teve como ponto de partida um problema específico identificado pela empresa parceira: a ausência de um processo estruturado e automatizado para a verificação da qualidade informacional de modelos BIM. A partir desse desafio concreto, buscou-se investigar formas de tornar esse processo mais eficiente. Inicialmente, não havia um método previamente definido; o foco era compreender o estado atual das ferramentas, abordagens e boas práticas existentes, para então propor um fluxo baseado nas soluções mais promissoras identificadas ao longo da pesquisa.

Durante esse percurso, uma abordagem baseada em *Information Delivery Specifications* se destacou, sobretudo por estar alinhada a princípios *OpenBIM* e por representar um método em desenvolvimento contínuo, com potencial de adaptação e evolução futura. A clareza estrutural oferecida pelos ficheiros IDS, somada à sua compatibilidade com fluxos automatizados, revelou-se uma oportunidade concreta de modernização do processo de verificação de modelos. Além disso, por se basear em um padrão aberto e normatizado pela *buildingSMART*, o uso de IDS aponta para uma solução com alto grau de perenidade, capaz de acompanhar a evolução das ferramentas e das exigências informacionais ao longo do tempo, sem se tornar obsoleta ou dependente de tecnologias proprietárias.

A proposta final consistiu na construção e teste de um fluxo de trabalho baseado na criação de ficheiros IDS, sua aplicação em ferramentas de verificação automatizada e posterior gestão das não conformidades detetadas. O estudo foi desenvolvido em parceria com uma empresa de engenharia, que forneceu os modelos utilizados no caso prático. Esses modelos, já submetidos anteriormente a uma verificação manual, permitiram uma comparação realista entre os métodos.

Foram analisados cinco modelos disciplinares de um mesmo edifício — arquitetura, estruturas, instalações elétricas, climatização (HVAC) e sistemas hidráulicos — o que possibilitou uma avaliação detalhada do desempenho do fluxo em diferentes contextos de modelação.

Embora tenha-se priorizado o uso de ferramentas baseadas em normas abertas e com ampla acessibilidade, algumas etapas do processo exigiram softwares com licenças comerciais, como o BIMcollab Zoom. Ainda assim, buscou-se demonstrar a viabilidade técnica e metodológica da integração entre soluções abertas e profissionais, com foco na clareza, interoperabilidade e rastreabilidade dos resultados.

A seguir serão apresentadas as conclusões sobre o fluxo proposto.

6.2 CONCLUSÕES SOBRE OS RESULTADOS

A aplicação do fluxo proposto permitiu avaliar, com base em um cenário real, o potencial das ferramentas baseadas em *OpenBIM* para automatizar a verificação da qualidade informacional de modelos BIM. Mais do que ganhos em eficiência, o processo revelou considerações relevantes sobre organização de dados, interpretação dos resultados, limitações técnicas e maturidade das ferramentas envolvidas.

6.2.1. AVALIAÇÃO DO FLUXO PROPOSTO E FERRAMENTAS UTILIZADAS

A criação de ficheiros IDS com ferramentas dedicadas mostrou-se viável, estável e acessível. O processo de definição de requisitos, organização das especificações e exportação dos ficheiros PDF e XML ocorreu sem grandes dificuldades, cumprindo seu papel de orientar as equipas modeladoras (no caso do PDF) e alimentar os softwares de verificação (no caso do XML). O uso do editor gratuito usBIM IDS reforça a viabilidade prática da metodologia, mesmo em contextos com restrições orçamentárias, e contribui para a disseminação do método em cenários que valorizam a liberdade tecnológica e a adoção de padrões abertos. Durante a preparação dos ficheiros IDS, a verificação preliminar da estrutura dos modelos foi realizada manualmente, incluindo a identificação de *property sets* e classes IFC. Esse processo, embora eficaz, revelou um potencial claro para automação futura, especialmente com o uso de bibliotecas como o IfcOpenShell em scripts personalizados.

Um ponto ainda não resolvido pelo fluxo é a ausência de critérios de verificação baseados no *spatial container* dos elementos. Embora o Blender identifique corretamente essa associação espacial, o editor IDS utilizado não permite que o utilizador defina regras com base nessa informação. Essa limitação, somada à ausência de convenções claras nos próprios modelos em relação à localização vertical dos elementos, dificultou a verificação de requisitos ligados ao piso ou compartimento dos objetos.

Quanto à agilidade do processo de verificação, os ganhos foram evidentes. No total, foram realizadas 58261 verificações automáticas em segundos. Esse nível de desempenho evidencia o contraste entre o processo automatizado e os métodos tradicionais de verificação manual baseados em planilhas e revisões visuais, mais sujeitos a erros humanos e inconsistências operacionais.

Além da eficiência, a liberdade estrutural permitida pelas ferramentas de edição de IDS revelou-se um diferencial importante. Cada empresa possui métodos, padrões e clientes distintos, com exigências específicas. Ter a possibilidade de estruturar os ficheiros IDS conforme essas particularidades garante maior aderência ao uso real no setor. Essa flexibilidade também se mostrou útil para ajustar o fluxo proposto, como foi feito neste estudo, visando facilitar etapas subsequentes, como a gestão de *issues* ou a organização de relatórios por tipo de elemento.

6.2.2. FLEXIBILIDADE, INTERPRETAÇÃO E ADAPTAÇÃO À PRÁTICA

Durante a execução da verificação automática, ficou evidente o caráter interpretativo dos relatórios gerados. Em muitos casos, as não conformidades identificadas estavam associadas a elementos secundários ou excepcionais, que não demandavam, na prática, o mesmo grau de informação que os elementos principais. Essa constatação reforça que a automatização da verificação não substitui o julgamento técnico: ela organiza, prioriza e evidencia o que deve ser analisado, mas ainda depende da experiência e do contexto da equipa coordenadora.

Do ponto de vista informacional, os modelos analisados apresentaram variação significativa. Os modelos arquitetónico e HVAC continham maior densidade de *property sets*, permitindo regras mais

robustas. Já o modelo de sistemas hidráulicos apresentou pouca informação verificável, o que limitou o impacto do fluxo sobre ele. Esses contrastes evidenciam a importância de definir requisitos claros e padronizados já nas etapas iniciais de modelação, reforçando o papel dos ficheiros IDS como guias práticos para o correto preenchimento da informação — uma função já reconhecida na literatura (M. Valinejadshoubi et al., 2024).

A etapa de exportação das não conformidades em formato BCF também trouxe contribuições e desafios. Embora a criação de *issues* no Bonsai funcione, o processo é inteiramente manual. A ausência de menus suspensos, validações automáticas ou preenchimento assistido torna o sistema vulnerável a erros de digitação, principalmente em campos como *Type*, *Status* e *Priority*. Isso pode comprometer a organização dos *issues* no *software* de gestão e gera um entrave crítico dentro do fluxo. Trata-se, possivelmente, de um dos maiores pontos de limitação da proposta.

Apesar disso, os *issues* exportados foram corretamente interpretados pelo BIMcollab Zoom, demonstrando que, com cuidado e atenção, é possível integrar ferramentas *open source* com plataformas proprietárias consolidadas do mercado. A navegação entre os *issues*, sua categorização e rastreabilidade ocorreram como esperado, e o ciclo de identificação de não conformidades pôde ser concluído. Ainda que não tenha havido uma equipa modeladora designada para corrigir os modelos nesta pesquisa, os *issues* ficaram disponíveis e poderiam ser utilizados para guiar uma etapa real de retrabalho.

6.2.3. BARREIRAS À INTEGRAÇÃO E CONTRIBUTOS DO ESTUDO

Do ponto de vista técnico, tanto o Blender quanto o Bonsai apresentaram desempenho sólido. A visualização dos dados, a geração dos relatórios HTML e a estabilidade geral do sistema atenderam às expectativas. No entanto, após o lançamento da versão 4.4 do Blender (março de 2025), foram observadas instabilidades, como fechamentos inesperados durante a exportação de ficheiros BCF, o que reforça a importância de acompanhar de perto a compatibilidade entre versões e a maturação contínua das ferramentas utilizadas.

Um aspeto estrutural importante discutido durante a pesquisa, e também apontado em conversa junto a empresa é a limitação relacionada à integração do fluxo. O ciclo proposto nesta dissertação não está conectado a um ambiente comum de dados (CDE), o que representa uma dificuldade prática em contextos de projeto reais. O uso de ficheiros IFC implica trocas constantes entre ambientes, e a falta de sincronização automática ou gestão centralizada pode gerar confusões, duplicidades de versões ou incertezas quanto ao estado atual do modelo. Essa desconexão anula parte dos ganhos de produtividade esperados e representa uma das maiores barreiras para a adoção desse tipo de fluxo em escala organizacional.

Enquanto não houver uma forma de integrar verificadores IFC, como o Bonsai, a ambientes colaborativos e centralizados, o fluxo permanecerá como uma alternativa externa ao processo principal, de adoção limitada. A iniciativa *OpenCDE*, atualmente em desenvolvimento pela *buildingSMART*, pode vir a suprir essa lacuna no futuro. No entanto, até que soluções práticas e estáveis sejam incorporadas ao mercado, esse continuará a ser um dos obstáculos da adoção de soluções baseadas em *OpenBIM*.

Ainda assim, mesmo com essas limitações, a estrutura proposta demonstrou ser funcional, aplicável e escalável. Os ganhos em rastreabilidade, clareza de requisitos, organização das verificações e controle de erros são evidentes — especialmente se comparados aos processos manuais ainda utilizados por muitas empresas.

A nível académico, esta dissertação contribui ao sistematizar, testar e documentar um fluxo de verificação informacional em BIM com base em princípios abertos e ferramentas acessíveis. O estudo

confirmou a viabilidade da proposta e destacou o valor de unir interoperabilidade, clareza de requisitos e rastreabilidade num processo prático e adaptável. Mais do que uma solução técnica, este trabalho aponta para uma construção digital mais transparente e colaborativa, onde a qualidade da informação é o que sustenta a confiança.

6.3 TRABALHOS FUTUROS

Este estudo explorou um fluxo de verificação automatizada com base em ficheiros IDS, utilizando ferramentas interoperáveis e acessíveis. No entanto, diversas oportunidades de aprofundamento e melhoria foram identificadas ao longo do percurso, as quais podem servir de base para investigações futuras.

Uma das possibilidades mais promissoras diz respeito à **automatização parcial das etapas iniciais do fluxo**, hoje realizadas manualmente. A identificação de *property sets*, classes IFC e outros metadados básicos necessários à construção dos ficheiros IDS poderia ser agilizada por meio de scripts personalizados em linguagens como Python, com apoio de bibliotecas como o *IfcOpenShell*. Essa abordagem não apenas reduziria o tempo necessário para estruturar os ficheiros, como também aumentaria a consistência e a confiabilidade dos dados de entrada.

Além disso, propõe-se o desenvolvimento de uma **plataforma de apoio à coordenação BIM em Power BI**, capaz de consolidar os relatórios HTML gerados pelo Bonsai para múltiplos modelos ou projetos. Isso permitiria uma visualização mais integrada dos resultados de verificação, possibilitando análises comparativas e decisões estratégicas com base em dados centralizados.

Outras sugestões incluem a **aprimoramento da criação de issues**, atualmente limitada por preenchimento manual e suscetível a erros. A criação de funcionalidades de preenchimento assistido, validação automática de campos e integração direta com ambientes comuns de dados (CDE) seriam passos importantes para tornar o processo mais fluido e confiável.

REFERÊNCIAS BIBLIOGRÁFICAS

- (ISO), T. I. O. f. S. (2018). ISO 19650-1:2018 Organization of information about construction works - Information management using building information modelling - Part 1: Concepts and Principles. In.
- Amor, R., & Dimyadi, J. (2021). The promise of automated compliance checking. *Developments in the Built Environment*, 5, 100039. <https://doi.org/https://doi.org/10.1016/j.dibe.2020.100039>
- Arditi, D., & Gunaydin, H. M. (1998). Factors That Affect Process Quality in the Life Cycle of Building Projects. *Journal of Construction Engineering and Management*, 124(3), 194-203. [https://doi.org/doi:10.1061/\(ASCE\)0733-9364\(1998\)124:3\(194\)](https://doi.org/doi:10.1061/(ASCE)0733-9364(1998)124:3(194))
- BIM Dictionary. (2025, July 31st, 2019). *Quality Assurance definition*. Retrieved May 21st from <https://bimdictionary.com/en/quality-assurance/1>
- buildingSMART, I. (2025a). Retrieved 12th of June from <https://www.buildingsmart.org/about/what-we-do/>
- buildingSMART, I. (2025b). *Global openBIM Mandates*.
- Chen, L., & Luo, H. (2014). A BIM-based construction quality management model and its applications. *Automation in Construction*, 46, 64-73. <https://doi.org/https://doi.org/10.1016/j.autcon.2014.05.009>
- Choi, J., Lee, S., & Kim, I. (2020). Development of Quality Control Requirements for Improving the Quality of Architectural Design Based on BIM. *Applied Sciences*, 10(20), 7074. <https://www.mdpi.com/2076-3417/10/20/7074>
- Donato, V., Lo Turco, M., & Bocconcino, M. M. (2018). BIM-QA/QC in the architectural design process. *Architectural Engineering and Design Management*, 14(3), 239-254. <https://doi.org/10.1080/17452007.2017.1370995>
- Eastman, C., Lee, J.-m., Jeong, Y.-s., & Lee, J.-k. (2009). Automatic rule-based checking of building designs. *Automation in Construction*, 18(8), 1011-1033. <https://doi.org/https://doi.org/10.1016/j.autcon.2009.07.002>
- Eastman, C., Teicholz, P., Sacks, R., & Liston, K. (2011). *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors* (Vol. 2). NJ: John Wiley & Sons. <https://doi.org/doi:10.1002/9780470261309>
- Ferguson, H., & Cleyton, L. (1988). *Quality in the constructed project: A guideline for owners, designers and constructors* (Vol. 1). ASCE.
- Hamledari, H., Azar, E., & McCabe, B. (2018). IFC-Based Development of As-Built and As-Is BIMs Using Construction and Facility Inspection Data: Site-to-BIM Data Transfer Automation. *Journal of Computing in Civil Engineering*, 32. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000727](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000727)
- Lee, H. W., Oh, H., Kim, Y., & Choi, K. (2015). Quantitative analysis of warnings in building information modeling (BIM). *Automation in Construction*, 51, 23-31. <https://doi.org/https://doi.org/10.1016/j.autcon.2014.12.007>
- Lu, W., Fung, A., Peng, Y., Liang, C., & Rowlinson, S. (2014). Cost-benefit analysis of Building Information Modeling implementation in building projects through demystification of time-effort distribution curves. *Building and Environment*, 82, 317-327. <https://doi.org/https://doi.org/10.1016/j.buildenv.2014.08.030>

- Marco, G., Slongo, C., & Siegele, D. (2024). Enriching Building Information Modeling Models through Information Delivery Specification. *Buildings*, 14, 2206. <https://doi.org/10.3390/buildings14072206>
- Mirarchi, C., & Pavan, A. (2019). *Building information models are dirty*. <https://doi.org/10.35490/EC3.2019.180>
- Moço, R. M. M. (2015). *Verificação Automática de Modelos Bim - Aplicação à Avaliação de Qualidade de Projetos de Edifícios de Habitação* (Publication Number 30835539) [M.C.E., Universidade do Porto (Portugal)]. Dissertations & Theses @ University of Porto. Portugal. <https://www.proquest.com/dissertations-theses/verificacao-automatica-de-modelos-bim-aplicacao-a/docview/2925383733/se-2?accountid=43623>
- Nawari, N. O. (2018). *Building Information Modeling: Automated Code Checking and Compliance Processes*. <https://doi.org/https://doi.org/10.1201/9781351200998>
- Noardo, F., Ohori, K., Krijnen, T., & Stoter, J. (2021). An Inspection of IFC Models from Practice. *Applied Sciences*, 11, 2232. <https://doi.org/10.3390/app11052232>
- Nuyts, E., Bonduel, M., & Verstraeten, R. (2024). Comparative analysis of approaches for automated compliance checking of construction data. *Advanced Engineering Informatics*, 60, 102443. <https://doi.org/https://doi.org/10.1016/j.aei.2024.102443>
- O'Brien, J. J. (1989). *Construction Inspection Handbook: Quality assurance/quality Control*. Springer New York, NY. <https://doi.org/https://doi.org/10.1007/978-1-4757-1191-2>
- Pazlar, T., & Turk, Ž. (2008). Interoperability in practice: Geometric data exchange using the IFC standard. *Electronic Journal of Information Technology in Construction*, 13, 362-380.
- Salaberri Martin, J. J. (2022). *Automated BIM QA/QC procedures with practical application : master thesis No.: 241/II. GR-BIM* [Magistrsko delo, Fakulteta za gradbeništvo in geodezijo]. Repozitorij Univerze v Ljubljani. Ljubljana. <https://repozitorij.uni-lj.si/IzpisGradiva.php?lang=slv&id=141482>
- Silva, L. G. G. B. (2024). Verificação de atendimento a requisitos de informação por elemento utilizando information delivery specification em projetos de edificações residenciais apoiados em building information modeling. <https://repositorio.ufsc.br/handle/123456789/262339>
- Solihin, W., Eastman, C., & Lee, Y.-C. (2015). Toward robust and quantifiable automated IFC quality validation. *Advanced Engineering Informatics*, 29(3), 739-756. <https://doi.org/https://doi.org/10.1016/j.aei.2015.07.006>
- Succar, B., Sher, W., & Williams, A. (2012). Measuring BIM Performance: Five Metrics. *Architectural Engineering and Design Management*, 8, 120-142. <https://doi.org/10.1080/17452007.2012.659506>
- Temel, B., & Başağa, H. (2020). Investigation of IFC file format for BIM based automated code compliance checking. *Journal of Construction Engineering, Management & Innovation*, 3, 113-130. <https://doi.org/10.31462/jcemi.2020.02113130>
- Tomczak, A., van Berlo, L., Bolpagni, M., Krijnen, T., & Borrmann, A. (2022). A review of methods to specify information requirements in digital construction projects. *IOP Conference Series Earth and Environmental Science*, 1101. <https://doi.org/10.1088/1755-1315/1101/9/092024>
- Valinejadshoubi, M., Moselhi, O., Iordanova, I., Valdivieso, F., & Bagchi, A. (2024). Automated system for high-accuracy quantity takeoff using BIM. *Automation in Construction*, 157, 105155. <https://doi.org/https://doi.org/10.1016/j.autcon.2023.105155>
- Valinejadshoubi, M., Moselhi, O., Iordanova, I., Valdivieso, F., Shakibabarough, A., & Bagchi, A. (2024). The Development of an Automated System for a Quality Evaluation of Engineering

BIM Models: A Case Study [Article]. *Applied Sciences (Switzerland)*, 14(8), Article 3244.
<https://doi.org/10.3390/app14083244>

Xu, Z., Huang, T., Li, B., Li, H., & Li, Q. (2018). Developing an IFC-Based Database for Construction Quality Evaluation. *Advances in Civil Engineering*, 2018(1), 3946051.
<https://doi.org/https://doi.org/10.1155/2018/3946051>

ANEXOS

A1 FICHEIROS IDS CRIADOS

Specifications

Beams

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCBEAM**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material
- MUST HAVE property **Reference Level** of PSet **Constraints**

Columns

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCCOLUMN**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE any material
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Base Level** of PSet **Constraints**

Covering

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCCOVERING**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material
- MUST HAVE property **Base Level** of PSet **Constraints**

Curtain Wall

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCCURTAINWALL**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material
- MUST HAVE property **Base Constraint** of PSet **Constraints**

Figura 38 - Relatório IDS do Modelo Arquitetónico (Parte 1)

Door

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCDOOR**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material
- MUST HAVE property **Level** of PSet **Constraints**

Flow Terminal

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWTERMINAL**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE any material
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**

Furnishing Element

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFURNISHINGELEMENT**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE any material
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**

Member

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCMEMBER**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material
- MUST HAVE property **Level** of PSet **Constraints**

Figura 39 - Relatório IDS do Modelo Arquitetónico (Parte 2)

Plate

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCPLATE**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE any material

Railing

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCRAILING**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE any material

Roof

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCROOF**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **IsExternal** of PSet **Pset_RoofCommon (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE any material
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Base Level** of PSet **Constraints**

Slab

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCSLAB**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE any material

Figura 40 - Relatório IDS do Modelo Arquitetônico (Parte 3)

Stair

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCSTAIR**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Base Level** of PSet **Constraints**
- MUST HAVE any material

Stair flight

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCSTAIRFLIGHT**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Wall

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCWALL**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material
- MUST HAVE property **Base Constraint** of PSet **Constraints**

Wall Standard Case

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCWALLSTANDARDCASE**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Base Constraint** of PSet **Constraints**
- MUST HAVE any material

Window

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCWINDOW**

that MEET the following requirements

- MUST HAVE property **Fire Rating** of PSet **BioNTech**
- MUST HAVE property **Is External** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Load Bearing** of PSet **BioNTech (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE any material

Figura 41 - Relatório IDS do Modelo Arquitetônico (Parte 4)

Specifications

Beams

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCBEAM**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **LoadBearing** of PSet **Pset_BeamCommon (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Reference Level** of PSet **Constraints**
- MUST HAVE any material

Columns

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCCOLUMN**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **LoadBearing** of PSet **Pset_ColumnCommon (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Base Level** of PSet **Constraints**
- MUST HAVE any material

Footing

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFOOTING**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Slabs

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCSLAB**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **LoadBearing** of PSet **Pset_SlabCommon (IFCBOOLEAN)**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Figura 42 - Relatório IDS do Modelo Estrutural (Parte 1)

Stairs

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCSTAIR**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

StairFlights

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCSTAIRFLIGHT**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Wall

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCWALL (or: IFCWALLSTANDARDCASE)**

that MEET the following requirements

- MUST HAVE property **GEG_5D_Article** of PSet **Identity Data**
- MUST HAVE property **GEG_5D_Description** of PSet **Identity Data**
- MUST HAVE property **LoadBearing** of PSet **Pset_WallCommon (IFCBOOLEAN)**
- MUST HAVE property **Base Constraint** of PSet **Constraints**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Figura 43 - Relatório IDS do Modelo Estrutural (Parte 2)

Specifications

Flow Controller

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWCONTROLLER**

that MEET the following requirements

- MUST HAVE attribute **Name**
- MUST HAVE attribute **ObjectType**

Flow Fitting

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWFITTING**

that MEET the following requirements

- MUST HAVE attribute **Name**
- MUST HAVE attribute **ObjectType**

Flow Segment

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWSEGMENT**

that MEET the following requirements

- MUST HAVE attribute **Name**
- MUST HAVE attribute **ObjectType**

Flow Terminal

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWTERMINAL**

that MEET the following requirements

- MUST HAVE attribute **Name**
- MUST HAVE attribute **ObjectType**

Figura 44 - Relatório IDS do Modelo de Sistemas Hidráulicos (*Plumbing*)

Specifications

Covering

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCCOVERING**

that MEET the following requirements

- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **System Type** of PSet **BioNTech**
- MUST HAVE property **System Name** of PSet **Mechanical**
- MUST HAVE any material

Energy Conversion Device

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCENERGYCONVERSIONDEVICE**

that MEET the following requirements

- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE any material

Flow Controller

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWCONTROLLER**

that MEET the following requirements

- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE property **System Type** of PSet **BioNTech**
- MUST HAVE property **System Name** of PSet **Mechanical**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Flow Fitting

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWFITTING**

that MEET the following requirements

- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE property **System Type** of PSet **BioNTech**
- MUST HAVE property **System Name** of PSet **Mechanical**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Figura 45 - Relatório IDS do Modelo HVAC (Parte 1)

Flow Segment

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWSEGMENT**

that MEET the following requirements

- MUST HAVE property **Reference Level** of PSet **Constraints**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **System Type** of PSet **BioNTech**
- MUST HAVE property **System Name** of PSet **Mechanical**
- MUST HAVE any material

Flow Terminal

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWTERMINAL**

that MEET the following requirements

- MUST HAVE property **Reference Level** of PSet **Constraints**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **System Type** of PSet **BioNTech**
- MUST HAVE property **System Name** of PSet **Mechanical**
- MUST HAVE any material

Mechanical Fastener

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCMECHANICALFASTENER**

that MEET the following requirements

- MUST HAVE property **Level** of PSet **Constraints**
- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **System Name** of PSet **Mechanical**
- MUST HAVE any material

Figura 46 - Relatório IDS do Modelo HVAC (Parte 2)

Specifications

Energy Conversion Device

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCENERGYCONVERSIONDEVICE**

that MEET the following requirements

- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE any material

Flow Fitting

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWFITTING**

that MEET the following requirements

- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Service Type** of PSet **BioNTech**
- MUST HAVE any material

Flow Segment

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWSEGMENT**

that MEET the following requirements

- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Service Type** of PSet **BioNTech**

Flow Terminal

IFC versions: **IFC2X3**

The model MUST contain entities that have

- IFC class **IFCFLOWTERMINAL**

that MEET the following requirements

- MUST HAVE property **Name** of PSet **BioNTech**
- MUST HAVE property **Power** of PSet **BioNTech**
- MUST HAVE any material

Figura 47 – Relatório IDS do Modelo Elétrico

A2 **FICHEIROS XML CRIADOS**

```

3 <ids:info>
4 <ids:title>verificação IFC Architecture</ids:title>
5 <ids:version>1.0</ids:version>
6 <ids:date>2025-05-05</ids:date>
7 <ids:purpose>Dissertação de Mestrado</ids:purpose>
8 </ids:info>
9 <ids:specifications>
10 <ids:specification ifcVersion="IFC2X3" name="Beams">
11 <ids:applicability minOccurs="1" maxOccurs="unbounded">
12 <ids:entity>
13 <ids:name>
14 <ids:simpleValue>IFCBEAM</ids:simpleValue>
15 </ids:name>
16 </ids:entity>
17 </ids:applicability>
18 <ids:requirements>
19 <ids:property cardinality="required">
20 <ids:propertySet>
21 <ids:simpleValue>BionTech</ids:simpleValue>
22 </ids:propertySet>
23 <ids:baseName>
24 <ids:simpleValue>Fire Rating</ids:simpleValue>
25 </ids:baseName>
26 </ids:property>
27 <ids:property dataType="IFCBOOLEAN" cardinality="required">
28 <ids:propertySet>
29 <ids:simpleValue>BionTech</ids:simpleValue>
30 </ids:propertySet>
31 <ids:baseName>
32 <ids:simpleValue>Is External</ids:simpleValue>
33 </ids:baseName>
34 </ids:property>
35 <ids:property dataType="IFCBOOLEAN" cardinality="required">
36 <ids:propertySet>
37 <ids:simpleValue>BionTech</ids:simpleValue>
38 </ids:propertySet>
39 <ids:baseName>
40 <ids:simpleValue>Load Bearing</ids:simpleValue>
41 </ids:baseName>
42 </ids:property>
43 <ids:property cardinality="required">
44 <ids:propertySet>
45 <ids:simpleValue>BionTech</ids:simpleValue>
46 </ids:propertySet>
47 <ids:baseName>
48 <ids:simpleValue>Name</ids:simpleValue>
49 </ids:baseName>
50 </ids:property>
51 <ids:property cardinality="required">
52 <ids:propertySet>
53 <ids:simpleValue>Constraints</ids:simpleValue>
54 </ids:propertySet>
55 <ids:baseName>
56 <ids:simpleValue>Reference Level</ids:simpleValue>
57 </ids:baseName>
58 </ids:property>
59 <ids:material cardinality="required" />
60 </ids:requirements>
61 </ids:specification>
62 <ids:specification ifcVersion="IFC2X3" name="Columns">
63 <ids:applicability minOccurs="1" maxOccurs="unbounded">
64 <ids:entity>
65 <ids:name>
66 <ids:simpleValue>IFCCOLUMN</ids:simpleValue>
67 </ids:name>
68 </ids:entity>
69 </ids:applicability>
70 <ids:requirements>
71 <ids:property cardinality="required">
72 <ids:propertySet>
73 <ids:simpleValue>BionTech</ids:simpleValue>
74 </ids:propertySet>
75 <ids:baseName>
76 <ids:simpleValue>Fire Rating</ids:simpleValue>
77 </ids:baseName>
78 </ids:property>
79 <ids:property dataType="IFCBOOLEAN" cardinality="required">
80 <ids:propertySet>
81 <ids:simpleValue>BionTech</ids:simpleValue>
82 </ids:propertySet>
83 <ids:baseName>
84 <ids:simpleValue>Is External</ids:simpleValue>
85 </ids:baseName>
86 </ids:property>
87 <ids:property dataType="IFCBOOLEAN" cardinality="required">
88 <ids:propertySet>
89 <ids:simpleValue>BionTech</ids:simpleValue>
90 </ids:propertySet>
91 <ids:baseName>
92 <ids:simpleValue>Load Bearing</ids:simpleValue>
93 </ids:baseName>
94 </ids:property>
95 <ids:property cardinality="required">
96 <ids:propertySet>
97 <ids:simpleValue>BionTech</ids:simpleValue>
98 </ids:propertySet>
99 <ids:baseName>
100 <ids:simpleValue>Name</ids:simpleValue>
101 </ids:baseName>
102 </ids:property>
103 <ids:property cardinality="required">
104 <ids:propertySet>
105 <ids:simpleValue>Constraints</ids:simpleValue>
106 </ids:propertySet>
107 <ids:baseName>
108 <ids:simpleValue>Base Level</ids:simpleValue>
109 </ids:baseName>
110 </ids:property>
111 <ids:material cardinality="required" />
112 </ids:requirements>
113 </ids:specification>
114 <ids:specification ifcVersion="IFC2X3" name="Covering">
115 <ids:applicability minOccurs="1" maxOccurs="unbounded">
116 <ids:entity>
117 <ids:name>
118 <ids:simpleValue>IFCCOVERING</ids:simpleValue>
119 </ids:name>
120 </ids:entity>
121 </ids:applicability>
122 <ids:requirements>
123 <ids:property cardinality="required">
124 <ids:propertySet>
125 <ids:simpleValue>BionTech</ids:simpleValue>
126 </ids:propertySet>
127 <ids:baseName>
128 <ids:simpleValue>Fire Rating</ids:simpleValue>
129 </ids:baseName>
130 </ids:property>
131 <ids:property dataType="IFCBOOLEAN" cardinality="required">
132 <ids:propertySet>
133 <ids:simpleValue>BionTech</ids:simpleValue>
134 </ids:propertySet>
135 <ids:baseName>
136 <ids:simpleValue>Is External</ids:simpleValue>
137 </ids:baseName>
138 </ids:property>
139 <ids:property dataType="IFCBOOLEAN" cardinality="required">
140 <ids:propertySet>
141 <ids:simpleValue>BionTech</ids:simpleValue>
142 </ids:propertySet>
143 <ids:baseName>
144 <ids:simpleValue>Load Bearing</ids:simpleValue>
145 </ids:baseName>
146 </ids:property>
147 <ids:property cardinality="required">
148 <ids:propertySet>
149 <ids:simpleValue>BionTech</ids:simpleValue>
150 </ids:propertySet>
151 <ids:baseName>
152 <ids:simpleValue>Name</ids:simpleValue>

```

Figura 48 - XML do IDS do Modelo Arquitetônico (Parte 1)

```

153     </ids:baseName>
154   </ids:property>
155   <ids:property cardinality="required">
156     <ids:propertySet>
157       <ids:simpleValue>Constraints</ids:simpleValue>
158     </ids:propertySet>
159     <ids:baseName>
160       <ids:simpleValue>Base Level</ids:simpleValue>
161     </ids:baseName>
162   </ids:property>
163   <ids:material cardinality="required" />
164 </ids:requirements>
165 </ids:specification>
166 <ids:specification ifcVersion="IFC2X3" name="Curtain Wall">
167   <ids:applicability minOccurs="1" maxOccurs="unbounded">
168     <ids:entity>
169       <ids:name>
170         <ids:simpleValue>IFCCURTAINWALL</ids:simpleValue>
171       </ids:name>
172     </ids:entity>
173   </ids:applicability>
174   <ids:requirements>
175     <ids:property cardinality="required">
176       <ids:propertySet>
177         <ids:simpleValue>BionTech</ids:simpleValue>
178       </ids:propertySet>
179       <ids:baseName>
180         <ids:simpleValue>Fire Rating</ids:simpleValue>
181       </ids:baseName>
182     </ids:property>
183     <ids:property dataType="IFCBOOLEAN" cardinality="required">
184       <ids:propertySet>
185         <ids:simpleValue>BionTech</ids:simpleValue>
186       </ids:propertySet>
187       <ids:baseName>
188         <ids:simpleValue>Is External</ids:simpleValue>
189       </ids:baseName>
190     </ids:property>
191     <ids:property dataType="IFCBOOLEAN" cardinality="required">
192       <ids:propertySet>
193         <ids:simpleValue>BionTech</ids:simpleValue>
194       </ids:propertySet>
195       <ids:baseName>
196         <ids:simpleValue>Load Bearing</ids:simpleValue>
197       </ids:baseName>
198     </ids:property>
199     <ids:property cardinality="required">
200       <ids:propertySet>
201         <ids:simpleValue>BionTech</ids:simpleValue>
202       </ids:propertySet>
203       <ids:baseName>
204         <ids:simpleValue>Name</ids:simpleValue>
205       </ids:baseName>
206     </ids:property>
207     <ids:property cardinality="required">
208       <ids:propertySet>
209         <ids:simpleValue>Constraints</ids:simpleValue>
210       </ids:propertySet>
211       <ids:baseName>
212         <ids:simpleValue>Base Constraint</ids:simpleValue>
213       </ids:baseName>
214     </ids:property>
215     <ids:material cardinality="required" />
216   </ids:requirements>
217 </ids:specification>
218 <ids:specification ifcVersion="IFC2X3" name="Door">
219   <ids:applicability minOccurs="1" maxOccurs="unbounded">
220     <ids:entity>
221       <ids:name>
222         <ids:simpleValue>IFCDOOR</ids:simpleValue>
223       </ids:name>
224     </ids:entity>
225   </ids:applicability>
226   <ids:requirements>
227     <ids:property cardinality="required">
228       <ids:propertySet>
229         <ids:simpleValue>BionTech</ids:simpleValue>
230       </ids:propertySet>
231       <ids:baseName>
232         <ids:simpleValue>Fire Rating</ids:simpleValue>
233       </ids:baseName>
234     </ids:property>
235     <ids:property dataType="IFCBOOLEAN" cardinality="required">
236       <ids:propertySet>
237         <ids:simpleValue>BionTech</ids:simpleValue>
238       </ids:propertySet>
239       <ids:baseName>
240         <ids:simpleValue>Is External</ids:simpleValue>
241       </ids:baseName>
242     </ids:property>
243     <ids:property dataType="IFCBOOLEAN" cardinality="required">
244       <ids:propertySet>
245         <ids:simpleValue>BionTech</ids:simpleValue>
246       </ids:propertySet>
247       <ids:baseName>
248         <ids:simpleValue>Load Bearing</ids:simpleValue>
249       </ids:baseName>
250     </ids:property>
251     <ids:property cardinality="required">
252       <ids:propertySet>
253         <ids:simpleValue>BionTech</ids:simpleValue>
254       </ids:propertySet>
255       <ids:baseName>
256         <ids:simpleValue>Name</ids:simpleValue>
257       </ids:baseName>
258     </ids:property>
259     <ids:property cardinality="required">
260       <ids:propertySet>
261         <ids:simpleValue>Constraints</ids:simpleValue>
262       </ids:propertySet>
263       <ids:baseName>
264         <ids:simpleValue>Level</ids:simpleValue>
265       </ids:baseName>
266     </ids:property>
267     <ids:material cardinality="required" />
268   </ids:requirements>
269 </ids:specification>
270 <ids:specification ifcVersion="IFC2X3" name="Flow Terminal">
271   <ids:applicability minOccurs="1" maxOccurs="unbounded">
272     <ids:entity>
273       <ids:name>
274         <ids:simpleValue>IFCFLOWTERMINAL</ids:simpleValue>
275       </ids:name>
276     </ids:entity>
277   </ids:applicability>
278   <ids:requirements>
279     <ids:property cardinality="required">
280       <ids:propertySet>
281         <ids:simpleValue>BionTech</ids:simpleValue>
282       </ids:propertySet>
283       <ids:baseName>
284         <ids:simpleValue>Fire Rating</ids:simpleValue>
285       </ids:baseName>
286     </ids:property>
287     <ids:property dataType="IFCBOOLEAN" cardinality="required">
288       <ids:propertySet>
289         <ids:simpleValue>BionTech</ids:simpleValue>
290       </ids:propertySet>
291       <ids:baseName>
292         <ids:simpleValue>Is External</ids:simpleValue>
293       </ids:baseName>
294     </ids:property>
295     <ids:property dataType="IFCBOOLEAN" cardinality="required">
296       <ids:propertySet>
297         <ids:simpleValue>BionTech</ids:simpleValue>

```

Figura 49 - XML do IDS do Modelo Arquitetónico (Parte 2)

```

328 </ids:propertySet>
329 <ids:baseName>
330 <ids:simpleValue>Load Bearing</ids:simpleValue>
331 </ids:baseName>
332 </ids:property>
333 <ids:property cardinality="required">
334 <ids:propertySet>
335 <ids:simpleValue>BionTech</ids:simpleValue>
336 </ids:propertySet>
337 <ids:baseName>
338 <ids:simpleValue>Name</ids:simpleValue>
339 </ids:baseName>
340 </ids:property>
341 <ids:property cardinality="required">
342 <ids:propertySet>
343 <ids:simpleValue>Constraints</ids:simpleValue>
344 </ids:propertySet>
345 <ids:baseName>
346 <ids:simpleValue>Level</ids:simpleValue>
347 </ids:baseName>
348 </ids:property>
349 <ids:material cardinality="required" />
350 </ids:requirements>
351 </ids:specification>
352 <ids:specification ifcVersion="IFC2X3" name="Furnishing Element">
353 <ids:applicability minOccurs="1" maxOccurs="unbounded">
354 <ids:entity>
355 <ids:name>
356 <ids:simpleValue>IFCFURNISHINGELEMENT</ids:simpleValue>
357 </ids:name>
358 </ids:entity>
359 </ids:applicability>
360 <ids:requirements>
361 <ids:property cardinality="required">
362 <ids:propertySet>
363 <ids:simpleValue>BionTech</ids:simpleValue>
364 </ids:propertySet>
365 <ids:baseName>
366 <ids:simpleValue>Fire Rating</ids:simpleValue>
367 </ids:baseName>
368 </ids:property>
369 <ids:property datatype="IFCBOOLEAN" cardinality="required">
370 <ids:propertySet>
371 <ids:simpleValue>BionTech</ids:simpleValue>
372 </ids:propertySet>
373 <ids:baseName>
374 <ids:simpleValue>Is External</ids:simpleValue>
375 </ids:baseName>
376 </ids:property>
377 <ids:property datatype="IFCBOOLEAN" cardinality="required">
378 <ids:propertySet>
379 <ids:simpleValue>BionTech</ids:simpleValue>
380 </ids:propertySet>
381 <ids:baseName>
382 <ids:simpleValue>Load Bearing</ids:simpleValue>
383 </ids:baseName>
384 </ids:property>
385 <ids:property cardinality="required">
386 <ids:propertySet>
387 <ids:simpleValue>BionTech</ids:simpleValue>
388 </ids:propertySet>
389 <ids:baseName>
390 <ids:simpleValue>Name</ids:simpleValue>
391 </ids:baseName>
392 </ids:property>
393 <ids:property cardinality="required">
394 <ids:propertySet>
395 <ids:simpleValue>Constraints</ids:simpleValue>
396 </ids:propertySet>
397 <ids:baseName>
398 <ids:simpleValue>Level</ids:simpleValue>
399 </ids:baseName>
400 </ids:property>
401 <ids:material cardinality="required" />
402 </ids:requirements>
403 </ids:specification>
404 <ids:specification ifcVersion="IFC2X3" name="Plate">
405 <ids:applicability minOccurs="1" maxOccurs="unbounded">
406 <ids:entity>
407 <ids:name>
408 <ids:simpleValue>IFCPLATE</ids:simpleValue>
409 </ids:name>
410 </ids:entity>
411 </ids:applicability>
412 <ids:requirements>
413 <ids:property cardinality="required">
414 <ids:propertySet>
415 <ids:simpleValue>BionTech</ids:simpleValue>
416 </ids:propertySet>
417 <ids:baseName>
418 <ids:simpleValue>Fire Rating</ids:simpleValue>
419 </ids:baseName>
420 </ids:property>
421 <ids:property datatype="IFCBOOLEAN" cardinality="required">
422 <ids:propertySet>
423 <ids:simpleValue>BionTech</ids:simpleValue>
424 </ids:propertySet>
425 <ids:baseName>
426 <ids:simpleValue>Is External</ids:simpleValue>
427 </ids:baseName>
428 </ids:property>

```

Figura 50 - XML do IDS do Modelo Arquitetônico (Parte 3)

```

450 </ids:property>
451 <ids:property dataType="IFCBOOLEAN" cardinality="required">
452   <ids:propertySet>
453     <ids:simpleValue>BionTech</ids:simpleValue>
454   </ids:propertySet>
455   <ids:baseName>
456     <ids:simpleValue>Load Bearing</ids:simpleValue>
457   </ids:baseName>
458 </ids:property>
459 <ids:property cardinality="required">
460   <ids:propertySet>
461     <ids:simpleValue>BionTech</ids:simpleValue>
462   </ids:propertySet>
463   <ids:baseName>
464     <ids:simpleValue>Name</ids:simpleValue>
465   </ids:baseName>
466 </ids:property>
467 <ids:property cardinality="required">
468   <ids:propertySet>
469     <ids:simpleValue>Constraints</ids:simpleValue>
470   </ids:propertySet>
471   <ids:baseName>
472     <ids:simpleValue>Level</ids:simpleValue>
473   </ids:baseName>
474 </ids:property>
475 <ids:material cardinality="required" />
476 </ids:requirements>
477 </ids:specification>
478 <ids:specification ifcVersion="IFC2X3" name="Railing">
479   <ids:applicability minOccurs="1" maxOccurs="unbounded">
480     <ids:entity>
481       <ids:name>
482         <ids:simpleValue>IFCRAILING</ids:simpleValue>
483       </ids:name>
484     </ids:entity>
485   </ids:applicability>
486   <ids:requirements>
487     <ids:property cardinality="required">
488       <ids:propertySet>
489         <ids:simpleValue>BionTech</ids:simpleValue>
490       </ids:propertySet>
491       <ids:baseName>
492         <ids:simpleValue>Fire Rating</ids:simpleValue>
493       </ids:baseName>
494     </ids:property>
495     <ids:property dataType="IFCBOOLEAN" cardinality="required">
496       <ids:propertySet>
497         <ids:simpleValue>BionTech</ids:simpleValue>
498       </ids:propertySet>
499       <ids:baseName>
500         <ids:simpleValue>Is External</ids:simpleValue>
501       </ids:baseName>
502     </ids:property>
503     <ids:property dataType="IFCBOOLEAN" cardinality="required">
504       <ids:propertySet>
505         <ids:simpleValue>BionTech</ids:simpleValue>
506       </ids:propertySet>
507       <ids:baseName>
508         <ids:simpleValue>Load Bearing</ids:simpleValue>
509       </ids:baseName>
510     </ids:property>
511     <ids:property cardinality="required">
512       <ids:propertySet>
513         <ids:simpleValue>BionTech</ids:simpleValue>
514       </ids:propertySet>
515       <ids:baseName>
516         <ids:simpleValue>Name</ids:simpleValue>
517       </ids:baseName>
518     </ids:property>
519     <ids:property cardinality="required">
520       <ids:propertySet>
521         <ids:simpleValue>Constraints</ids:simpleValue>
522       </ids:propertySet>
523       <ids:baseName>
524         <ids:simpleValue>Level</ids:simpleValue>
525       </ids:baseName>
526     </ids:property>
527   </ids:applicability>
528   <ids:requirements>
529     <ids:property cardinality="required">
530       <ids:propertySet>
531         <ids:simpleValue>BionTech</ids:simpleValue>
532       </ids:propertySet>
533       <ids:baseName>
534         <ids:simpleValue>Fire Rating</ids:simpleValue>
535       </ids:baseName>
536     </ids:property>
537     <ids:property dataType="IFCBOOLEAN" cardinality="required">
538       <ids:propertySet>
539         <ids:simpleValue>Pset_RoofCommon</ids:simpleValue>
540       </ids:propertySet>
541       <ids:baseName>
542         <ids:simpleValue>Is External</ids:simpleValue>
543       </ids:baseName>
544     </ids:property>
545     <ids:property dataType="IFCBOOLEAN" cardinality="required">
546       <ids:propertySet>
547         <ids:simpleValue>BionTech</ids:simpleValue>
548       </ids:propertySet>
549       <ids:baseName>
550         <ids:simpleValue>Load Bearing</ids:simpleValue>
551       </ids:baseName>
552     </ids:property>
553     <ids:property cardinality="required">
554       <ids:propertySet>
555         <ids:simpleValue>BionTech</ids:simpleValue>
556       </ids:propertySet>
557       <ids:baseName>
558         <ids:simpleValue>Name</ids:simpleValue>
559       </ids:baseName>
560     </ids:property>
561     <ids:property cardinality="required">
562       <ids:propertySet>
563         <ids:simpleValue>Constraints</ids:simpleValue>
564       </ids:propertySet>
565       <ids:baseName>
566         <ids:simpleValue>Base Level</ids:simpleValue>
567       </ids:baseName>
568     </ids:property>
569     <ids:material cardinality="required" />
570   </ids:requirements>
571 </ids:specification>
572 <ids:specification ifcVersion="IFC2X3" name="Slab">
573   <ids:applicability minOccurs="1" maxOccurs="unbounded">
574     <ids:entity>
575       <ids:name>
576         <ids:simpleValue>IFCSLAB</ids:simpleValue>
577       </ids:name>
578     </ids:entity>
579   </ids:applicability>
580   <ids:requirements>
581     <ids:property cardinality="required">
582       <ids:propertySet>
583         <ids:simpleValue>BionTech</ids:simpleValue>
584       </ids:propertySet>
585       <ids:baseName>
586         <ids:simpleValue>Fire Rating</ids:simpleValue>
587       </ids:baseName>
588     </ids:property>
589     <ids:property dataType="IFCBOOLEAN" cardinality="required">
590       <ids:propertySet>
591     
```

Figura 51 - XML do IDS do Modelo Arquitetônico (Parte 4)

```

601      <ids:simpleValue>BionTech</ids:simpleValue>
602    </ids:propertySet>
603    <ids:baseName>
604      <ids:simpleValue>Is External</ids:simpleValue>
605    </ids:baseName>
606  </ids:property>
607  <ids:property dataType="IFCBOOLEAN" cardinality="required">
608    <ids:propertySet>
609      <ids:simpleValue>BionTech</ids:simpleValue>
610    </ids:propertySet>
611    <ids:baseName>
612      <ids:simpleValue>Load Bearing</ids:simpleValue>
613    </ids:baseName>
614  </ids:property>
615  <ids:property cardinality="required">
616    <ids:propertySet>
617      <ids:simpleValue>BionTech</ids:simpleValue>
618    </ids:propertySet>
619    <ids:baseName>
620      <ids:simpleValue>Name</ids:simpleValue>
621    </ids:baseName>
622  </ids:property>
623  <ids:property cardinality="required">
624    <ids:propertySet>
625      <ids:simpleValue>Constraints</ids:simpleValue>
626    </ids:propertySet>
627    <ids:baseName>
628      <ids:simpleValue>Level</ids:simpleValue>
629    </ids:baseName>
630  </ids:property>
631  <ids:material cardinality="required" />
632 </ids:requirements>
633 </ids:specification>
634 <ids:specification ifcVersion="IFC2X3" name="Stair">
635   <ids:applicability minOccurs="1" maxOccurs="unbounded">
636     <ids:entity>
637       <ids:name>
638         <ids:simpleValue>IFCSTAIR</ids:simpleValue>
639       </ids:name>
640     </ids:entity>
641   </ids:applicability>
642   <ids:requirements>
643     <ids:property cardinality="required">
644       <ids:propertySet>
645         <ids:simpleValue>BionTech</ids:simpleValue>
646       </ids:propertySet>
647       <ids:baseName>
648         <ids:simpleValue>Fire Rating</ids:simpleValue>
649       </ids:baseName>
650     </ids:property>
651   <ids:property dataType="IFCBOOLEAN" cardinality="required">
652     <ids:propertySet>
653       <ids:simpleValue>BionTech</ids:simpleValue>
654     </ids:propertySet>
655     <ids:baseName>
656       <ids:simpleValue>Is External</ids:simpleValue>
657     </ids:baseName>
658   </ids:property>
659   <ids:property dataType="IFCBOOLEAN" cardinality="required">
660     <ids:propertySet>
661       <ids:simpleValue>BionTech</ids:simpleValue>
662     </ids:propertySet>
663     <ids:baseName>
664       <ids:simpleValue>Load Bearing</ids:simpleValue>
665     </ids:baseName>
666   </ids:property>
667   <ids:property cardinality="required">
668     <ids:propertySet>
669       <ids:simpleValue>BionTech</ids:simpleValue>
670     </ids:propertySet>
671     <ids:baseName>
672       <ids:simpleValue>Name</ids:simpleValue>
673     </ids:baseName>
674   </ids:property>
675   <ids:property cardinality="required">
676     <ids:propertySet>
677       <ids:simpleValue>Constraints</ids:simpleValue>
678     </ids:propertySet>
679     <ids:baseName>
680       <ids:simpleValue>Base Level</ids:simpleValue>
681     </ids:baseName>
682   </ids:property>
683   <ids:material cardinality="required" />
684 </ids:requirements>
685 </ids:specification>
686 <ids:specification ifcVersion="IFC2X3" name="Stair flight">
687   <ids:applicability minOccurs="1" maxOccurs="unbounded">
688     <ids:entity>
689       <ids:name>
690         <ids:simpleValue>IFCSTAIRFLIGHT</ids:simpleValue>
691       </ids:name>
692     </ids:entity>
693   </ids:applicability>
694   <ids:requirements>
695     <ids:property cardinality="required">
696       <ids:propertySet>
697         <ids:simpleValue>BionTech</ids:simpleValue>
698       </ids:propertySet>
699       <ids:baseName>
700         <ids:simpleValue>Fire Rating</ids:simpleValue>
701       </ids:baseName>
702     </ids:property>
703   <ids:property dataType="IFCBOOLEAN" cardinality="required">
704     <ids:propertySet>
705       <ids:simpleValue>BionTech</ids:simpleValue>
706     </ids:propertySet>
707     <ids:baseName>
708       <ids:simpleValue>Is External</ids:simpleValue>
709     </ids:baseName>
710   </ids:property>
711   <ids:property dataType="IFCBOOLEAN" cardinality="required">
712     <ids:propertySet>
713       <ids:simpleValue>BionTech</ids:simpleValue>
714     </ids:propertySet>
715     <ids:baseName>
716       <ids:simpleValue>Load Bearing</ids:simpleValue>
717     </ids:baseName>
718   </ids:property>
719   <ids:property cardinality="required">
720     <ids:propertySet>
721       <ids:simpleValue>BionTech</ids:simpleValue>
722     </ids:propertySet>
723     <ids:baseName>
724       <ids:simpleValue>Name</ids:simpleValue>
725     </ids:baseName>
726   </ids:property>
727   <ids:material cardinality="required" />
728 </ids:requirements>
729 </ids:specification>
730 <ids:specification ifcVersion="IFC2X3" name="Wall">
731   <ids:applicability minOccurs="1" maxOccurs="unbounded">
732     <ids:entity>
733       <ids:name>
734         <ids:simpleValue>IFCWALL</ids:simpleValue>
735       </ids:name>
736     </ids:entity>
737   </ids:applicability>
738   <ids:requirements>
739     <ids:property cardinality="required">
740       <ids:propertySet>
741         <ids:simpleValue>BionTech</ids:simpleValue>
742       </ids:propertySet>
743       <ids:baseName>
744         <ids:simpleValue>Fire Rating</ids:simpleValue>
745       </ids:baseName>
746     </ids:property>
747   <ids:property dataType="IFCBOOLEAN" cardinality="required">
748     <ids:propertySet>
749       <ids:simpleValue>BionTech</ids:simpleValue>
750     </ids:propertySet>
751     <ids:baseName>
752       <ids:simpleValue>Is External</ids:simpleValue>

```

Figura 52 - XML do IDS do Modelo Arquitetônico (Parte 5)

```

753     </ids:baseName>
754 </ids:property>
755 <ids:property dataType="IFCBOOLEAN" cardinality="required">
756   <ids:propertySet>
757     <ids:simpleValue>BionTech</ids:simpleValue>
758   </ids:propertySet>
759   <ids:baseName>
760     <ids:simpleValue>Load Bearing</ids:simpleValue>
761   </ids:baseName>
762 </ids:property>
763 <ids:property cardinality="required">
764   <ids:propertySet>
765     <ids:simpleValue>BionTech</ids:simpleValue>
766   </ids:propertySet>
767   <ids:baseName>
768     <ids:simpleValue>Name</ids:simpleValue>
769   </ids:baseName>
770 </ids:property>
771 <ids:property cardinality="required">
772   <ids:propertySet>
773     <ids:simpleValue>Constraints</ids:simpleValue>
774   </ids:propertySet>
775   <ids:baseName>
776     <ids:simpleValue>Base Constraint</ids:simpleValue>
777   </ids:baseName>
778 </ids:property>
779 <ids:material cardinality="required" />
780 </ids:requirements>
781 </ids:specification>
782 <ids:specification ifcVersion="IFC2X3" name="Wall Standard Case">
783   <ids:applicability minOccurs="1" maxOccurs="unbounded">
784     <ids:entity>
785       <ids:name>
786         <ids:simpleValue>IFCWALLSTANDARDCASE</ids:simpleValue>
787       </ids:name>
788     </ids:entity>
789   </ids:applicability>
790   <ids:requirements>
791     <ids:property cardinality="required">
792       <ids:propertySet>
793         <ids:simpleValue>BionTech</ids:simpleValue>
794       </ids:propertySet>
795       <ids:baseName>
796         <ids:simpleValue>Fire Rating</ids:simpleValue>
797       </ids:baseName>
798     </ids:property>
799     <ids:property dataType="IFCBOOLEAN" cardinality="required">
800       <ids:propertySet>
801         <ids:simpleValue>BionTech</ids:simpleValue>
802       </ids:propertySet>
803       <ids:baseName>
804         <ids:simpleValue>Is External</ids:simpleValue>
805       </ids:baseName>
806     </ids:property>
807     <ids:property dataType="IFCBOOLEAN" cardinality="required">
808       <ids:propertySet>
809         <ids:simpleValue>BionTech</ids:simpleValue>
810       </ids:propertySet>
811       <ids:baseName>
812         <ids:simpleValue>Load Bearing</ids:simpleValue>
813       </ids:baseName>
814     </ids:property>
815     <ids:property cardinality="required">
816       <ids:propertySet>
817         <ids:simpleValue>BionTech</ids:simpleValue>
818       </ids:propertySet>
819       <ids:baseName>
820         <ids:simpleValue>Name</ids:simpleValue>
821       </ids:baseName>
822     </ids:property>
823     <ids:property cardinality="required">
824       <ids:propertySet>
825         <ids:simpleValue>Constraints</ids:simpleValue>
826       </ids:propertySet>
827       <ids:baseName>
828         <ids:simpleValue>Base Constraint</ids:simpleValue>
829       </ids:baseName>
830     </ids:property>
831   </ids:requirements>
832 </ids:specification>
833 </ids:specifications>
834 </ids:ids>

```

Figura 53 - XML do IDS do Modelo Arquitetônico (Parte 6)

```

3  <ids:info>
4    <ids:title>Verificação IFC Structural</ids:title>
5    <ids:description>IDS para o Modelo Estrutural</ids:description>
6  </ids:info>
7  <ids:specifications>
8    <ids:specification ifcVersion="IFC2X3" name="Beams">
9      <ids:applicability minOccurs="1" maxOccurs="unbounded">
10        <ids:entity>
11          <ids:name>
12            <ids:simpleValue>IFCBEAM</ids:simpleValue>
13          </ids:name>
14        </ids:entity>
15      </ids:applicability>
16      <ids:requirements>
17        <ids:property cardinality="required">
18          <ids:propertySet>
19            <ids:simpleValue>Identity Data</ids:simpleValue>
20          </ids:propertySet>
21          <ids:baseName>
22            <ids:simpleValue>GEG_5D_Article</ids:simpleValue>
23          </ids:baseName>
24        </ids:property>
25        <ids:property cardinality="required">
26          <ids:propertySet>
27            <ids:simpleValue>Identity Data</ids:simpleValue>
28          </ids:propertySet>
29          <ids:baseName>
30            <ids:simpleValue>GEG_5D_Description</ids:simpleValue>
31          </ids:baseName>
32        </ids:property>
33        <ids:property dataType="IFCBOOLEAN" cardinality="required">
34          <ids:propertySet>
35            <ids:simpleValue>Pset_BeamCommon</ids:simpleValue>
36          </ids:propertySet>
37          <ids:baseName>
38            <ids:simpleValue>LoadBearing</ids:simpleValue>
39          </ids:baseName>
40        </ids:property>
41        <ids:property cardinality="required">
42          <ids:propertySet>
43            <ids:simpleValue>BioNTech</ids:simpleValue>
44          </ids:propertySet>
45          <ids:baseName>
46            <ids:simpleValue>Name</ids:simpleValue>
47          </ids:baseName>
48        </ids:property>
49        <ids:property cardinality="required">
50          <ids:propertySet>
51            <ids:simpleValue>Constraints</ids:simpleValue>
52          </ids:propertySet>
53          <ids:baseName>
54            <ids:simpleValue>Reference Level</ids:simpleValue>
55          </ids:baseName>
56        </ids:property>
57      <ids:material cardinality="required" />
58    </ids:requirements>
59  </ids:specification>
60  <ids:specification ifcVersion="IFC2X3" name="Columns">
61    <ids:applicability minOccurs="1" maxOccurs="unbounded">
62      <ids:entity>
63        <ids:name>
64          <ids:simpleValue>IFCCOLUMN</ids:simpleValue>
65        </ids:name>
66      </ids:entity>
67    </ids:applicability>
68    <ids:requirements>
69      <ids:property cardinality="required">
70        <ids:propertySet>
71          <ids:simpleValue>Identity Data</ids:simpleValue>
72        </ids:propertySet>
73        <ids:baseName>
74          <ids:simpleValue>GEG_5D_Article</ids:simpleValue>
75        </ids:baseName>
76      </ids:property>
77      <ids:property cardinality="required">
78        <ids:propertySet>
79          <ids:simpleValue>Identity Data</ids:simpleValue>
80        </ids:propertySet>
81        <ids:baseName>
82          <ids:simpleValue>GEG_5D_Description</ids:simpleValue>
83        </ids:baseName>
84      </ids:property>
85      <ids:property dataType="IFCBOOLEAN" cardinality="required">
86        <ids:propertySet>
87          <ids:simpleValue>Pset_ColumnCommon</ids:simpleValue>
88        </ids:propertySet>
89        <ids:baseName>
90          <ids:simpleValue>LoadBearing</ids:simpleValue>
91        </ids:baseName>
92      </ids:property>
93      <ids:property cardinality="required">
94        <ids:propertySet>
95          <ids:simpleValue>BioNTech</ids:simpleValue>
96        </ids:propertySet>
97        <ids:baseName>
98          <ids:simpleValue>Name</ids:simpleValue>
99        </ids:baseName>
100      </ids:property>
101      <ids:property cardinality="required">
102        <ids:propertySet>
103          <ids:simpleValue>Constraints</ids:simpleValue>
104        </ids:propertySet>
105        <ids:baseName>
106          <ids:simpleValue>Base Level</ids:simpleValue>
107        </ids:baseName>
108      </ids:property>
109    <ids:material cardinality="required" />
110  </ids:requirements>
111 </ids:specification>
112 <ids:specification ifcVersion="IFC2X3" name="Footing">
113   <ids:applicability minOccurs="1" maxOccurs="unbounded">
114     <ids:entity>
115       <ids:name>
116         <ids:simpleValue>IFCFOOTING</ids:simpleValue>
117       </ids:name>
118     </ids:entity>
119   </ids:applicability>
120   <ids:requirements>
121     <ids:property cardinality="required">
122       <ids:propertySet>
123         <ids:simpleValue>Identity Data</ids:simpleValue>
124       </ids:propertySet>
125       <ids:baseName>
126         <ids:simpleValue>GEG_5D_Article</ids:simpleValue>
127       </ids:baseName>
128     </ids:property>
129     <ids:property cardinality="required">
130       <ids:propertySet>
131         <ids:simpleValue>Identity Data</ids:simpleValue>
132       </ids:propertySet>
133       <ids:baseName>
134         <ids:simpleValue>GEG_5D_Description</ids:simpleValue>
135       </ids:baseName>

```

Figura 54 - XML do IDS do Modelo Estrutural (Parte 1)

<pre> 136 </ids:property> 137 <ids:property cardinality="required"> 138 <ids:propertySet> 139 <ids:simpleValue>BioNTech</ids:simpleValue> 140 </ids:propertySet> 141 <ids:baseName> 142 <ids:simpleValue>Name</ids:simpleValue> 143 </ids:baseName> 144 </ids:property> 145 <ids:material cardinality="required" /> 146 </ids:requirements> 147 </ids:specification> 148 <ids:specification ifcVersion="IFC2X3" name="Slabs"> 149 <ids:applicability minOccurs="1" maxOccurs="unbounded"> 150 <ids:entity> 151 <ids:name> 152 <ids:simpleValue>IFCSLAB</ids:simpleValue> 153 </ids:name> 154 </ids:entity> 155 </ids:applicability> 156 <ids:requirements> 157 <ids:property cardinality="required"> 158 <ids:propertySet> 159 <ids:simpleValue>Identity Data</ids:simpleValue> 160 </ids:propertySet> 161 <ids:baseName> 162 <ids:simpleValue>GEG_5D_Article</ids:simpleValue> 163 </ids:baseName> 164 </ids:property> 165 <ids:property cardinality="required"> 166 <ids:propertySet> 167 <ids:simpleValue>Identity Data</ids:simpleValue> 168 </ids:propertySet> 169 <ids:baseName> 170 <ids:simpleValue>GEG_5D_Description</ids:simpleValue> 171 </ids:baseName> 172 </ids:property> 173 <ids:property dataType="IFCBOOLEAN" cardinality="required"> 174 <ids:propertySet> 175 <ids:simpleValue>Pset_SlabCommon</ids:simpleValue> 176 </ids:propertySet> 177 <ids:baseName> 178 <ids:simpleValue>loadBearing</ids:simpleValue> 179 </ids:baseName> 180 </ids:property> 181 <ids:property cardinality="required"> 182 <ids:propertySet> 183 <ids:simpleValue>BioNTech</ids:simpleValue> 184 </ids:propertySet> 185 <ids:baseName> 186 <ids:simpleValue>Name</ids:simpleValue> 187 </ids:baseName> 188 </ids:property> 189 <ids:material cardinality="required" /> 190 </ids:requirements> 191 </ids:specification> 192 <ids:specification ifcVersion="IFC2X3" name="Stairs"> 193 <ids:applicability minOccurs="1" maxOccurs="unbounded"> 194 <ids:entity> 195 <ids:name> 196 <ids:simpleValue>IFCSTAIR</ids:simpleValue> 197 </ids:name> </pre>	<pre> 198 </ids:entity> 199 </ids:applicability> 200 </ids:requirements> 201 <ids:property cardinality="required"> 202 <ids:propertySet> 203 <ids:simpleValue>Identity Data</ids:simpleValue> 204 </ids:propertySet> 205 <ids:baseName> 206 <ids:simpleValue>GEG_5D_Article</ids:simpleValue> 207 </ids:baseName> 208 </ids:property> 209 <ids:property cardinality="required"> 210 <ids:propertySet> 211 <ids:simpleValue>Identity Data</ids:simpleValue> 212 </ids:propertySet> 213 <ids:baseName> 214 <ids:simpleValue>GEG_5D_Description</ids:simpleValue> 215 </ids:baseName> 216 </ids:property> 217 <ids:property cardinality="required"> 218 <ids:propertySet> 219 <ids:simpleValue>BioNTech</ids:simpleValue> 220 </ids:propertySet> 221 <ids:baseName> 222 <ids:simpleValue>Name</ids:simpleValue> 223 </ids:baseName> 224 </ids:property> 225 <ids:material cardinality="required" /> 226 </ids:requirements> 227 </ids:specification> 228 <ids:specification ifcVersion="IFC2X3" name="Stairflights&#xA;"> 229 <ids:applicability minOccurs="1" maxOccurs="unbounded"> 230 <ids:entity> 231 <ids:name> 232 <ids:simpleValue>IFCSTAIRFLIGHT</ids:simpleValue> 233 </ids:name> 234 </ids:entity> 235 </ids:applicability> 236 <ids:requirements> 237 <ids:property cardinality="required"> 238 <ids:propertySet> 239 <ids:simpleValue>Identity Data</ids:simpleValue> 240 </ids:propertySet> 241 <ids:baseName> 242 <ids:simpleValue>GEG_5D_Article</ids:simpleValue> 243 </ids:baseName> 244 </ids:property> 245 <ids:property cardinality="required"> 246 <ids:propertySet> 247 <ids:simpleValue>Identity Data</ids:simpleValue> 248 </ids:propertySet> 249 <ids:baseName> 250 <ids:simpleValue>GEG_5D_Description</ids:simpleValue> 251 </ids:baseName> 252 </ids:property> 253 <ids:property cardinality="required"> 254 <ids:propertySet> 255 <ids:simpleValue>BioNTech</ids:simpleValue> 256 </ids:propertySet> 257 <ids:baseName> 258 <ids:simpleValue>Name</ids:simpleValue> 259 </ids:baseName> 260 </ids:property> 261 <ids:material cardinality="required" /> </pre>
--	---

Figura 55 - XML do IDS do Modelo Estrutural (Parte 2)

```

261     <ids:material cardinality="required" />
262   </ids:requirements>
263 </ids:specification>
264 <ids:specification ifcVersion="IFC2X3" name="Wall">
265   <ids:applicability minOccurs="1" maxOccurs="unbounded">
266     <ids:entity>
267       <ids:name>
268         <xs:restriction base="xs:string">
269           <xs:enumeration value="IFCWALL" />
270           <xs:enumeration value="IFCWALLSTANDARDCASE" />
271         </xs:restriction>
272       </ids:name>
273     </ids:entity>
274   </ids:applicability>
275   <ids:requirements>
276     <ids:property cardinality="required">
277       <ids:propertySet>
278         <ids:simpleValue>Identity Data</ids:simpleValue>
279       </ids:propertySet>
280       <ids:baseName>
281         <ids:simpleValue>GEG_5D_Article</ids:simpleValue>
282       </ids:baseName>
283     </ids:property>
284     <ids:property cardinality="required">
285       <ids:propertySet>
286         <ids:simpleValue>Identity Data</ids:simpleValue>
287       </ids:propertySet>
288       <ids:baseName>
289         <ids:simpleValue>GEG_5D_Description</ids:simpleValue>
290       </ids:baseName>
291     </ids:property>
292     <ids:property dataType="IFCBOOLEAN" cardinality="required">
293       <ids:propertySet>
294         <ids:simpleValue>Pset_WallCommon</ids:simpleValue>
295       </ids:propertySet>
296       <ids:baseName>
297         <ids:simpleValue>LoadBearing</ids:simpleValue>
298       </ids:baseName>
299     </ids:property>
300     <ids:property cardinality="required">
301       <ids:propertySet>
302         <ids:simpleValue>Constraints</ids:simpleValue>
303       </ids:propertySet>
304       <ids:baseName>
305         <ids:simpleValue>Base Constraint</ids:simpleValue>
306       </ids:baseName>
307     </ids:property>
308     <ids:property cardinality="required">
309       <ids:propertySet>
310         <ids:simpleValue>BioNTech</ids:simpleValue>
311       </ids:propertySet>
312       <ids:baseName>
313         <ids:simpleValue>Name</ids:simpleValue>
314       </ids:baseName>
315     </ids:property>
316   </ids:material cardinality="required" />
317 </ids:requirements>
318 </ids:specification>
319 </ids:specifications>
320 </ids:ids>

```

Figura 56 - XML do IDS do Modelo Estrutural (Parte 3)

```

3 <ids:info>
4   <ids:title>Verificação IFC Plumbing</ids:title>
5 </ids:info>
6 <ids:specifications>
7   <ids:specification ifcVersion="IFC2X3" name="Flow Controller">
8     <ids:applicability minOccurs="1" maxOccurs="unbounded">
9       <ids:entity>
10        <ids:name>
11          <ids:simpleValue>IFCFLOWCONTROLLER</ids:simpleValue>
12        </ids:name>
13      </ids:entity>
14    </ids:applicability>
15    <ids:requirements>
16      <ids:attribute cardinality="required">
17        <ids:name>
18          <ids:simpleValue>Name</ids:simpleValue>
19        </ids:name>
20      </ids:attribute>
21      <ids:attribute cardinality="required">
22        <ids:name>
23          <ids:simpleValue>ObjectType</ids:simpleValue>
24        </ids:name>
25      </ids:attribute>
26    </ids:requirements>
27  </ids:specification>
28  <ids:specification ifcVersion="IFC2X3" name="Flow Fitting">
29    <ids:applicability minOccurs="1" maxOccurs="unbounded">
30      <ids:entity>
31        <ids:name>
32          <ids:simpleValue>IFCFLOWFITTING</ids:simpleValue>
33        </ids:name>
34      </ids:entity>
35    </ids:applicability>
36    <ids:requirements>
37      <ids:attribute cardinality="required">
38        <ids:name>
39          <ids:simpleValue>Name</ids:simpleValue>
40        </ids:name>
41      </ids:attribute>
42      <ids:attribute cardinality="required">
43        <ids:name>
44          <ids:simpleValue>ObjectType</ids:simpleValue>
45        </ids:name>
46      </ids:attribute>
47    </ids:requirements>
48  </ids:specification>
49  <ids:specification ifcVersion="IFC2X3" name="Flow Segment">
50    <ids:applicability minOccurs="1" maxOccurs="unbounded">
51      <ids:entity>
52        <ids:name>
53          <ids:simpleValue>IFCFLOWSEGMENT</ids:simpleValue>
54        </ids:name>
55      </ids:entity>
56    </ids:applicability>
57    <ids:requirements>
58      <ids:attribute cardinality="required">
59        <ids:name>
60          <ids:simpleValue>Name</ids:simpleValue>
61        </ids:name>
62      </ids:attribute>
63      <ids:attribute cardinality="required">
64        <ids:name>
65          <ids:simpleValue>ObjectType</ids:simpleValue>
66        </ids:name>
67      </ids:attribute>
68    </ids:requirements>
69  </ids:specification>
70  <ids:specification ifcVersion="IFC2X3" name="Flow Terminal">
71    <ids:applicability minOccurs="1" maxOccurs="unbounded">
72      <ids:entity>
73        <ids:name>
74          <ids:simpleValue>IFCFLOWTERMINAL</ids:simpleValue>
75        </ids:name>
76      </ids:entity>
77    </ids:applicability>
78    <ids:requirements>
79      <ids:attribute cardinality="required">
80        <ids:name>
81          <ids:simpleValue>Name</ids:simpleValue>
82        </ids:name>
83      </ids:attribute>
84      <ids:attribute cardinality="required">
85        <ids:name>
86          <ids:simpleValue>ObjectType</ids:simpleValue>
87        </ids:name>
88      </ids:attribute>
89    </ids:requirements>
90  </ids:specification>
91 </ids:specifications>
92 </ids:ids>

```

Figura 57 - XML do IDS do Modelo de Sistemas Hidráulicos (Plumbing)

```

3 <ids:info>
4 <ids:title>Verificação IFC HVAC</ids:title>
5 </ids:info>
6 <ids:specifications>
7 <ids:specification ifcVersion="IFC2X3" name="Covering">
8 <ids:applicability minOccurs="1" maxOccurs="unbounded">
9 <ids:entity>
10 <ids:name>
11 <ids:simpleValue>IFCCOVERING</ids:simpleValue>
12 </ids:name>
13 </ids:entity>
14 </ids:applicability>
15 <ids:requirements>
16 <ids:property cardinality="required">
17 <ids:propertySet>
18 <ids:simpleValue>BionTech</ids:simpleValue>
19 </ids:propertySet>
20 <ids:baseName>
21 <ids:simpleValue>Name</ids:simpleValue>
22 </ids:baseName>
23 </ids:property>
24 <ids:property cardinality="required">
25 <ids:propertySet>
26 <ids:simpleValue>BionTech</ids:simpleValue>
27 </ids:propertySet>
28 <ids:baseName>
29 <ids:simpleValue>System Type</ids:simpleValue>
30 </ids:baseName>
31 </ids:property>
32 <ids:property cardinality="required">
33 <ids:propertySet>
34 <ids:simpleValue>Mechanical</ids:simpleValue>
35 </ids:propertySet>
36 <ids:baseName>
37 <ids:simpleValue>System Name</ids:simpleValue>
38 </ids:baseName>
39 </ids:property>
40 <ids:material cardinality="required" />
41 </ids:requirements>
42 </ids:specification>
43 <ids:specification ifcVersion="IFC2X3" name="Energy Conversion Device">
44 <ids:applicability minOccurs="1" maxOccurs="unbounded">
45 <ids:entity>
46 <ids:name>
47 <ids:simpleValue>IFCENERGYCONVERSIONDEVICE</ids:simpleValue>
48 </ids:name>
49 </ids:entity>
50 </ids:applicability>
51 <ids:requirements>
52 <ids:property cardinality="required">
53 <ids:propertySet>
54 <ids:simpleValue>BionTech</ids:simpleValue>
55 </ids:propertySet>
56 <ids:baseName>
57 <ids:simpleValue>Name</ids:simpleValue>
58 </ids:baseName>
59 </ids:property>
60 <ids:property cardinality="required">
61 <ids:propertySet>
62 <ids:simpleValue>Constraints</ids:simpleValue>
63 </ids:propertySet>
64 <ids:baseName>
65 <ids:simpleValue>Level</ids:simpleValue>
66 </ids:baseName>
67 </ids:property>
68 <ids:material cardinality="required" />
69 </ids:requirements>
70 </ids:specification>
71 <ids:specification ifcVersion="IFC2X3" name="Flow Controller">
72 <ids:applicability minOccurs="1" maxOccurs="unbounded">
73 <ids:entity>
74 <ids:name>
75 <ids:simpleValue>IFCFLOWCONTROLLER</ids:simpleValue>
76 </ids:name>
77 </ids:entity>
78 </ids:applicability>
79 <ids:requirements>
80 <ids:property cardinality="required">
81 <ids:propertySet>
82 <ids:simpleValue>Constraints</ids:simpleValue>
83 </ids:propertySet>
84 <ids:baseName>
85 <ids:simpleValue>Level</ids:simpleValue>
86 </ids:baseName>
87 </ids:property>
88 <ids:property cardinality="required">
89 <ids:propertySet>
90 <ids:simpleValue>BionTech</ids:simpleValue>
91 </ids:propertySet>
92 <ids:baseName>
93 <ids:simpleValue>System Type</ids:simpleValue>
94 </ids:baseName>
95 </ids:property>
96 <ids:property cardinality="required">
97 <ids:propertySet>
98 <ids:simpleValue>Mechanical</ids:simpleValue>
99 </ids:propertySet>
100 <ids:baseName>
101 <ids:simpleValue>System Name</ids:simpleValue>
102 </ids:baseName>
103 </ids:property>
104 <ids:property cardinality="required">
105 <ids:propertySet>
106 <ids:simpleValue>BionTech</ids:simpleValue>
107 </ids:propertySet>
108 <ids:baseName>
109 <ids:simpleValue>Name</ids:simpleValue>
110 </ids:baseName>
111 </ids:property>
112 <ids:material cardinality="required" />
113 </ids:requirements>
114 </ids:specification>
115 <ids:specification ifcVersion="IFC2X3" name="Flow Fitting">
116 <ids:applicability minOccurs="1" maxOccurs="unbounded">
117 <ids:entity>
118 <ids:name>
119 <ids:simpleValue>IFCFLOWFITTING</ids:simpleValue>
120 </ids:name>
121 </ids:entity>
122 </ids:applicability>
123 <ids:requirements>
124 <ids:property cardinality="required">
125 <ids:propertySet>
126 <ids:simpleValue>Constraints</ids:simpleValue>
127 </ids:propertySet>
128 <ids:baseName>
129 <ids:simpleValue>Level</ids:simpleValue>
130 </ids:baseName>
131 </ids:property>
132 <ids:property cardinality="required">
133 <ids:propertySet>
134 <ids:simpleValue>BionTech</ids:simpleValue>
135 </ids:propertySet>
136 <ids:baseName>
137 <ids:simpleValue>System Type</ids:simpleValue>
138 </ids:baseName>
139 </ids:property>
140 <ids:property cardinality="required">
141 <ids:propertySet>
142 <ids:simpleValue>Mechanical</ids:simpleValue>
143 </ids:propertySet>
144 <ids:baseName>
145 <ids:simpleValue>System Name</ids:simpleValue>
146 </ids:baseName>
147 </ids:property>
148 <ids:property cardinality="required">
149 <ids:propertySet>
150 <ids:simpleValue>BionTech</ids:simpleValue>
151 </ids:propertySet>
152 <ids:baseName>
153 <ids:simpleValue>Name</ids:simpleValue>
154 </ids:baseName>
155 </ids:property>
156 <ids:material cardinality="required" />

```

Figura 58 - XML do IDS do Modelo HVAC (Parte 1)

```

157 </ids:requirements>
158 </ids:specification>
159 <ids:specification ifcVersion="IFC2X3" name="Flow Segment">
160 <ids:applicability minOccurs="1" maxOccurs="unbounded">
161 <ids:entity>
162 <ids:name>
163 <ids:simpleValue>IFCFLOWSEGMENT</ids:simpleValue>
164 </ids:name>
165 </ids:entity>
166 </ids:applicability>
167 <ids:requirements>
168 <ids:property cardinality="required">
169 <ids:propertySet>
170 <ids:simpleValue>Constraints</ids:simpleValue>
171 </ids:propertySet>
172 <ids:baseName>
173 <ids:simpleValue>Reference Level</ids:simpleValue>
174 </ids:baseName>
175 </ids:property>
176 <ids:property cardinality="required">
177 <ids:propertySet>
178 <ids:simpleValue>BionTech</ids:simpleValue>
179 </ids:propertySet>
180 <ids:baseName>
181 <ids:simpleValue>Name</ids:simpleValue>
182 </ids:baseName>
183 </ids:property>
184 <ids:property cardinality="required">
185 <ids:propertySet>
186 <ids:simpleValue>BionTech</ids:simpleValue>
187 </ids:propertySet>
188 <ids:baseName>
189 <ids:simpleValue>System Type</ids:simpleValue>
190 </ids:baseName>
191 </ids:property>
192 <ids:property cardinality="required">
193 <ids:propertySet>
194 <ids:simpleValue>Mechanical</ids:simpleValue>
195 </ids:propertySet>
196 <ids:baseName>
197 <ids:simpleValue>System Name</ids:simpleValue>
198 </ids:baseName>
199 </ids:property>
200 <ids:material cardinality="required" />
201 </ids:requirements>
202 </ids:specification>
203 <ids:specification ifcVersion="IFC2X3" name="Flow Terminal">
204 <ids:applicability minOccurs="1" maxOccurs="unbounded">
205 <ids:entity>
206 <ids:name>
207 <ids:simpleValue>IFCFLOWTERMINAL</ids:simpleValue>
208 </ids:name>
209 </ids:entity>
210 </ids:applicability>
211 <ids:requirements>
212 <ids:property cardinality="required">
213 <ids:propertySet>
214 <ids:simpleValue>Constraints</ids:simpleValue>
215 </ids:propertySet>
216 <ids:baseName>
217 <ids:simpleValue>Reference Level</ids:simpleValue>
218 </ids:baseName>
219 </ids:property>
220 <ids:property cardinality="required">
221 <ids:propertySet>
222 <ids:simpleValue>BionTech</ids:simpleValue>
223 </ids:propertySet>
224 <ids:baseName>
225 <ids:simpleValue>Name</ids:simpleValue>
226 </ids:baseName>
227 </ids:property>
228 <ids:property cardinality="required">
229 <ids:propertySet>
230 <ids:simpleValue>BionTech</ids:simpleValue>
231 </ids:propertySet>
232 <ids:baseName>
233 <ids:simpleValue>System Type</ids:simpleValue>
234 </ids:baseName>
235 </ids:property>
236 <ids:property cardinality="required">
237 <ids:propertySet>
238 <ids:simpleValue>Mechanical</ids:simpleValue>
239 </ids:propertySet>
240 <ids:baseName>
241 <ids:simpleValue>System Name</ids:simpleValue>
242 </ids:baseName>
243 </ids:property>
244 <ids:material cardinality="required" />
245 </ids:requirements>
246 </ids:specification>
247 <ids:specification ifcVersion="IFC2X3" name="Mechanical Fastener">
248 <ids:applicability minOccurs="1" maxOccurs="unbounded">
249 <ids:entity>
250 <ids:name>
251 <ids:simpleValue>IFCMECHANICALFASTENER</ids:simpleValue>
252 </ids:name>
253 </ids:entity>
254 </ids:applicability>
255 <ids:requirements>
256 <ids:property cardinality="required">
257 <ids:propertySet>
258 <ids:simpleValue>constraints</ids:simpleValue>
259 </ids:propertySet>
260 <ids:baseName>
261 <ids:simpleValue>Level</ids:simpleValue>
262 </ids:baseName>
263 </ids:property>
264 <ids:property cardinality="required">
265 <ids:propertySet>
266 <ids:simpleValue>BionTech</ids:simpleValue>
267 </ids:propertySet>
268 <ids:baseName>
269 <ids:simpleValue>Name</ids:simpleValue>
270 </ids:baseName>
271 </ids:property>
272 <ids:property cardinality="required">
273 <ids:propertySet>
274 <ids:simpleValue>Mechanical</ids:simpleValue>
275 </ids:propertySet>
276 <ids:baseName>
277 <ids:simpleValue>System Name</ids:simpleValue>
278 </ids:baseName>
279 </ids:property>
280 <ids:material cardinality="required" />
281 </ids:requirements>
282 </ids:specification>
283 </ids:specifications>
284 </ids:ids>

```

Figura 59 - XML do IDS do Modelo HVAC (Parte 2)

```

3 <ids:info>
4 <ids:title>Verificação IFC Electrical</ids:title>
5 </ids:info>
6 <ids:specifications>
7 <ids:specification ifcVersion="IFC2X3" name="Energy Conversion Device">
8 <ids:applicability minOccurs="1" maxOccurs="unbounded">
9 <ids:entity>
10 <ids:name>
11 <ids:simpleValue>IFCENERGYCONVERSIONDEVICE</ids:simpleValue>
12 </ids:name>
13 </ids:entity>
14 </ids:applicability>
15 <ids:requirements>
16 <ids:property cardinality="required">
17 <ids:propertySet>
18 <ids:simpleValue>BioNTech</ids:simpleValue>
19 </ids:propertySet>
20 <ids:baseName>
21 <ids:simpleValue>Name</ids:simpleValue>
22 </ids:baseName>
23 </ids:property>
24 <ids:material cardinality="required" />
25 </ids:requirements>
26 </ids:specification>
27 <ids:specification ifcVersion="IFC2X3" name="Flow Fitting">
28 <ids:applicability minOccurs="1" maxOccurs="unbounded">
29 <ids:entity>
30 <ids:name>
31 <ids:simpleValue>IFCFLOWFITTING</ids:simpleValue>
32 </ids:name>
33 </ids:entity>
34 </ids:applicability>
35 <ids:requirements>
36 <ids:property cardinality="required">
37 <ids:propertySet>
38 <ids:simpleValue>BioNTech</ids:simpleValue>
39 </ids:propertySet>
40 <ids:baseName>
41 <ids:simpleValue>Name</ids:simpleValue>
42 </ids:baseName>
43 </ids:property>
44 <ids:property cardinality="required">
45 <ids:propertySet>
46 <ids:simpleValue>BioNTech</ids:simpleValue>
47 </ids:propertySet>
48 <ids:baseName>
49 <ids:simpleValue>Service Type</ids:simpleValue>
50 </ids:baseName>
51 </ids:property>
52 <ids:material cardinality="required" />
53 </ids:requirements>
54 </ids:specification>
55 <ids:specification ifcVersion="IFC2X3" name="Flow Segment">
56 <ids:applicability minOccurs="1" maxOccurs="unbounded">
57 <ids:entity>
58 <ids:name>
59 <ids:simpleValue>IFCFLOWSEGMENT</ids:simpleValue>
60 </ids:name>
61 </ids:entity>
62 </ids:applicability>
63 <ids:requirements>
64 <ids:property cardinality="required">
65 <ids:propertySet>
66 <ids:simpleValue>BioNTech</ids:simpleValue>
67 </ids:propertySet>
68 <ids:baseName>
69 <ids:simpleValue>Name</ids:simpleValue>
70 </ids:baseName>
71 </ids:property>
72 <ids:property cardinality="required">
73 <ids:propertySet>
74 <ids:simpleValue>BioNTech</ids:simpleValue>
75 </ids:propertySet>
76 <ids:baseName>
77 <ids:simpleValue>Service Type</ids:simpleValue>
78 </ids:baseName>
79 </ids:property>
80 </ids:requirements>
81 </ids:specification>
82 <ids:specification ifcVersion="IFC2X3" name="Flow Terminal">
83 <ids:applicability minOccurs="1" maxOccurs="unbounded">
84 <ids:entity>
85 <ids:name>
86 <ids:simpleValue>IFCFLOWTERMINAL</ids:simpleValue>
87 </ids:name>
88 </ids:entity>
89 </ids:applicability>
90 <ids:requirements>
91 <ids:property cardinality="required">
92 <ids:propertySet>
93 <ids:simpleValue>BioNTech</ids:simpleValue>
94 </ids:propertySet>
95 <ids:baseName>
96 <ids:simpleValue>Name</ids:simpleValue>
97 </ids:baseName>
98 </ids:property>
99 <ids:property cardinality="required">
100 <ids:propertySet>
101 <ids:simpleValue>BioNTech</ids:simpleValue>
102 </ids:propertySet>
103 <ids:baseName>
104 <ids:simpleValue>Power</ids:simpleValue>
105 </ids:baseName>
106 </ids:property>
107 <ids:material cardinality="required" />
108 </ids:requirements>
109 </ids:specification>
110 </ids:specifications>
111 </ids:ids>

```

Figura 60 - XML do IDS do Modelo Elétrico

A3 **RELATÓRIOS HTML**

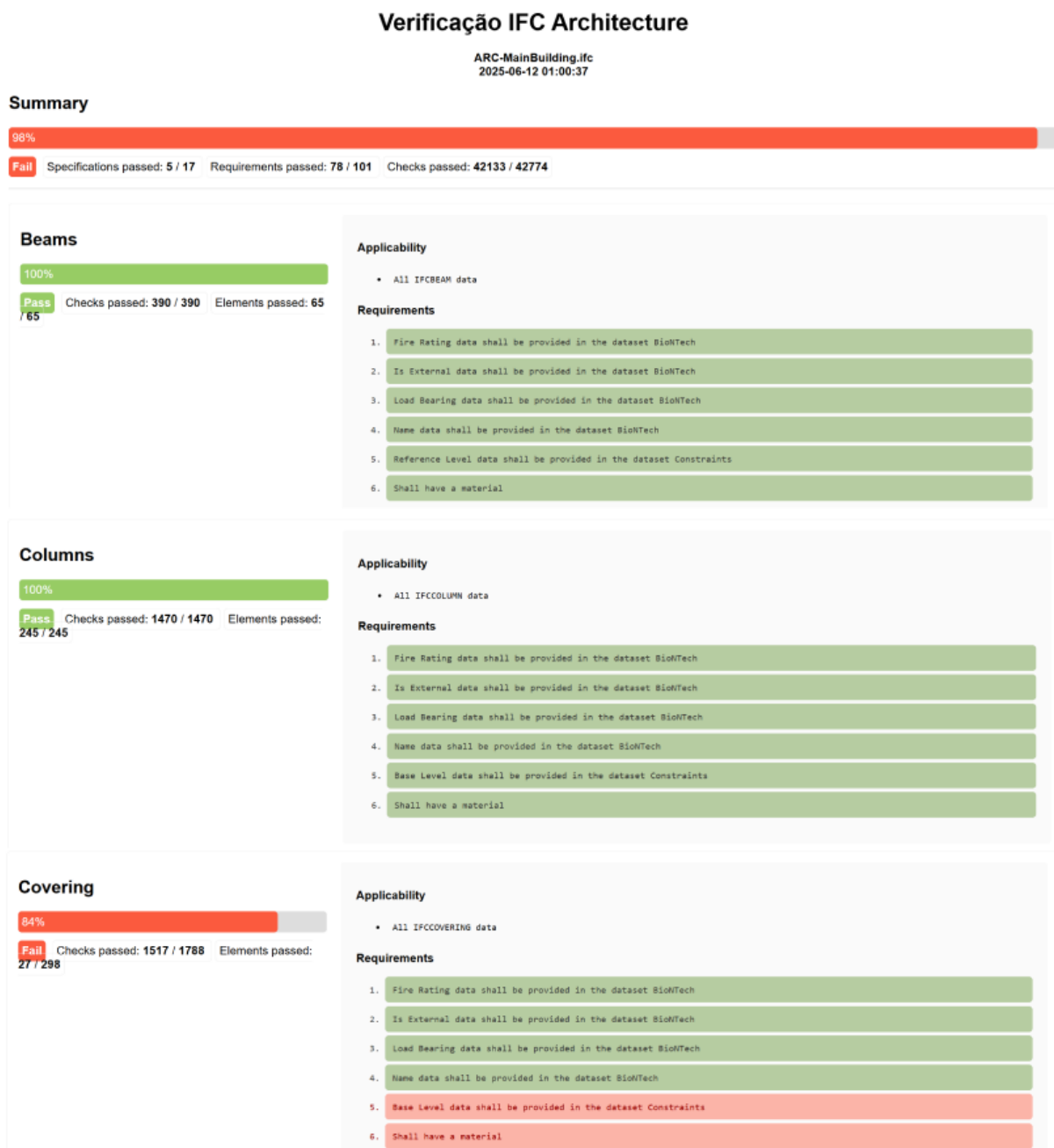


Figura 61 - Relatório HTML do Modelo Arquitetônico (Parte 1)

Curtain Wall

83%
Fail Checks passed: 820 / 984 Elements passed: 0 / 164

Applicability

- All IFCURTAINWALL data

Requirements

- Fire Rating data shall be provided in the dataset BioNTech
- Is External data shall be provided in the dataset BioNTech
- Load Bearing data shall be provided in the dataset BioNTech
- Name data shall be provided in the dataset BioNTech
- Base Constraint data shall be provided in the dataset Constraints
- Shall have a material

Door

100%
Pass Checks passed: 1716 / 1716 Elements passed: 286 / 286

Applicability

- All IFCDOOR data

Requirements

- Fire Rating data shall be provided in the dataset BioNTech
- Is External data shall be provided in the dataset BioNTech
- Load Bearing data shall be provided in the dataset BioNTech
- Name data shall be provided in the dataset BioNTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Flow Terminal

97%
Fail Checks passed: 1962 / 2004 Elements passed: 292 / 334

Applicability

- All IFCFLOWTERMINAL data

Requirements

- Fire Rating data shall be provided in the dataset BioNTech
- Is External data shall be provided in the dataset BioNTech
- Load Bearing data shall be provided in the dataset BioNTech
- Name data shall be provided in the dataset BioNTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Furnishing Element

94%
Fail Checks passed: 68 / 72 Elements passed: 8 / 12

Applicability

- All IFCFURNISHINGELEMENT data

Requirements

- Fire Rating data shall be provided in the dataset BioNTech
- Is External data shall be provided in the dataset BioNTech
- Load Bearing data shall be provided in the dataset BioNTech
- Name data shall be provided in the dataset BioNTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Figura 62 - Relatório HTML do Modelo Arquitetônico (Parte 2)

Member

99%
Fail Checks passed: 17458 / 17472 Elements passed: 2898 / 2912

Applicability

- All IFCMEMBER data

Requirements

- Fire Rating data shall be provided in the dataset \$IotTech
- Is External data shall be provided in the dataset \$IotTech
- Load Bearing data shall be provided in the dataset \$IotTech
- Name data shall be provided in the dataset \$IotTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Plate

100%
Pass Checks passed: 8940 / 8940 Elements passed: 1490 / 1490

Applicability

- All IFCPLATE data

Requirements

- Fire Rating data shall be provided in the dataset \$IotTech
- Is External data shall be provided in the dataset \$IotTech
- Load Bearing data shall be provided in the dataset \$IotTech
- Name data shall be provided in the dataset \$IotTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Railing

96%
Fail Checks passed: 838 / 870 Elements passed: 113 / 145

Applicability

- All IFCRAILING data

Requirements

- Fire Rating data shall be provided in the dataset \$IotTech
- Is External data shall be provided in the dataset \$IotTech
- Load Bearing data shall be provided in the dataset \$IotTech
- Name data shall be provided in the dataset \$IotTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Roof

94%
Fail Checks passed: 85 / 90 Elements passed: 10 / 15

Applicability

- All IFCROOF data

Requirements

- Fire Rating data shall be provided in the dataset \$IotTech
- Is External data shall be provided in the dataset Post_RoofCommon
- Load Bearing data shall be provided in the dataset \$IotTech
- Name data shall be provided in the dataset \$IotTech
- Base Level data shall be provided in the dataset Constraints
- Shall have a material

Slab

95%
Fail Checks passed: 887 / 930 Elements passed: 122 / 155

Applicability

- All IFCSLAB data

Requirements

- Fire Rating data shall be provided in the dataset \$IotTech
- Is External data shall be provided in the dataset \$IotTech
- Load Bearing data shall be provided in the dataset \$IotTech
- Name data shall be provided in the dataset \$IotTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Figura 63 - Relatório HTML do Modelo Arquitetônico (Parte 3)

Stair

74%
Fail Checks passed: 80 / 108 Elements passed: 0 / 18

Stair flight

100%
Pass Checks passed: 120 / 120 Elements passed: 24 / 24

Wall

98%
Fail Checks passed: 2468 / 2502 Elements passed: 400 / 417

Wall Standard Case

99%
Fail Checks passed: 2890 / 2892 Elements passed: 481 / 482

Window

99%
Fail Checks passed: 424 / 426 Elements passed: 69 / 71

Applicability

- All IPCSTAIR data

Requirements

- Fire Rating data shall be provided in the dataset BionTech
- Is External data shall be provided in the dataset BionTech
- Load Bearing data shall be provided in the dataset BionTech
- Name data shall be provided in the dataset BionTech
- Base Level data shall be provided in the dataset Constraints
- Shall have a material

Applicability

- All IPCSTAIRFLIGHT data

Requirements

- Fire Rating data shall be provided in the dataset BionTech
- Is External data shall be provided in the dataset BionTech
- Load Bearing data shall be provided in the dataset BionTech
- Name data shall be provided in the dataset BionTech
- Shall have a material

Applicability

- All IFOWALL data

Requirements

- Fire Rating data shall be provided in the dataset BionTech
- Is External data shall be provided in the dataset BionTech
- Load Bearing data shall be provided in the dataset BionTech
- Name data shall be provided in the dataset BionTech
- Base Constraint data shall be provided in the dataset Constraints
- Shall have a material

Applicability

- All IPCNALLSTANDARDCASE data

Requirements

- Fire Rating data shall be provided in the dataset BionTech
- Is External data shall be provided in the dataset BionTech
- Load Bearing data shall be provided in the dataset BionTech
- Name data shall be provided in the dataset BionTech
- Base Constraint data shall be provided in the dataset Constraints
- Shall have a material

Applicability

- All IFOWINDOW data

Requirements

- Fire Rating data shall be provided in the dataset BionTech
- Is External data shall be provided in the dataset BionTech
- Load Bearing data shall be provided in the dataset BionTech
- Name data shall be provided in the dataset BionTech
- Level data shall be provided in the dataset Constraints
- Shall have a material

Figura 64 - Relatório HTML do Modelo Arquitetônico (Parte 4)

Verificação IFC Structural

STR-MainBuilding.ifc
 2025-06-12 01:29:16

Summary

83%

Fail

Specifications passed: 0 / 7

Requirements passed: 20 / 35

Checks passed: 4230 / 5072

Beams

75%

Fail

Checks passed: 929 / 1224

Elements passed: 54 / 204

Applicability

- All IFCBEAM data

Requirements

- GEG_5D_Article data shall be provided in the dataset Identity Data
- GEG_5D_Description data shall be provided in the dataset Identity Data
- LoadBearing data shall be provided in the dataset Pset_BeamCommon
- Name data shall be provided in the dataset BioNTech
- Reference Level data shall be provided in the dataset Constraints
- Shall have a material

Columns

88%

Fail

Checks passed: 2342 / 2640

Elements passed: 288 / 440

Applicability

- All IFCCOLUMN data

Requirements

- GEG_5D_Article data shall be provided in the dataset Identity Data
- GEG_5D_Description data shall be provided in the dataset Identity Data
- LoadBearing data shall be provided in the dataset Pset_ColumnCommon
- Name data shall be provided in the dataset BioNTech
- Base Level data shall be provided in the dataset Constraints
- Shall have a material

Footing

75%

Fail

Checks passed: 36 / 48

Elements passed: 0 / 12

Applicability

- All IFCFOOTING data

Requirements

- GEG_5D_Article data shall be provided in the dataset Identity Data
- GEG_5D_Description data shall be provided in the dataset Identity Data
- Name data shall be provided in the dataset BioNTech
- Shall have a material

Figura 65 - Relatório HTML do Modelo Estrutural (Parte 1)

Slabs

72%

Fail Checks passed: 348 / 480
Elements passed: 40 / 96

Applicability

- All IFCSLAB data

Requirements

- GEG_SD_Article data shall be provided in the dataset Identity Data
- GEG_SD_Description data shall be provided in the dataset Identity Data
- LoadBearing data shall be provided in the dataset Pset_SlabCommon
- Name data shall be provided in the dataset BioNTech
- Shall have a material

Stairs

60%

Fail Checks passed: 58 / 96
Elements passed: 0 / 24

Applicability

- All IFCSTAIR data

Requirements

- GEG_SD_Article data shall be provided in the dataset Identity Data
- GEG_SD_Description data shall be provided in the dataset Identity Data
- Name data shall be provided in the dataset BioNTech
- Shall have a material

StairFlights

50%

Fail Checks passed: 52 / 104
Elements passed: 0 / 26

Applicability

- All IFCSTAIRFLIGHT data

Requirements

- GEG_SD_Article data shall be provided in the dataset Identity Data
- GEG_SD_Description data shall be provided in the dataset Identity Data
- Name data shall be provided in the dataset BioNTech
- Shall have a material

Wall

96%

Fail Checks passed: 465 / 480
Elements passed: 65 / 80

Applicability

- All {'enumeration': ['IFCWALL', 'IFCWALLSTANDARDCASE']} data

Requirements

- GEG_SD_Article data shall be provided in the dataset Identity Data
- GEG_SD_Description data shall be provided in the dataset Identity Data
- LoadBearing data shall be provided in the dataset Pset_WallCommon
- Base Constraint data shall be provided in the dataset Constraints
- Name data shall be provided in the dataset BioNTech
- Shall have a material

Figura 66 - Relatório HTML do Modelo Estrutural (Parte 2)

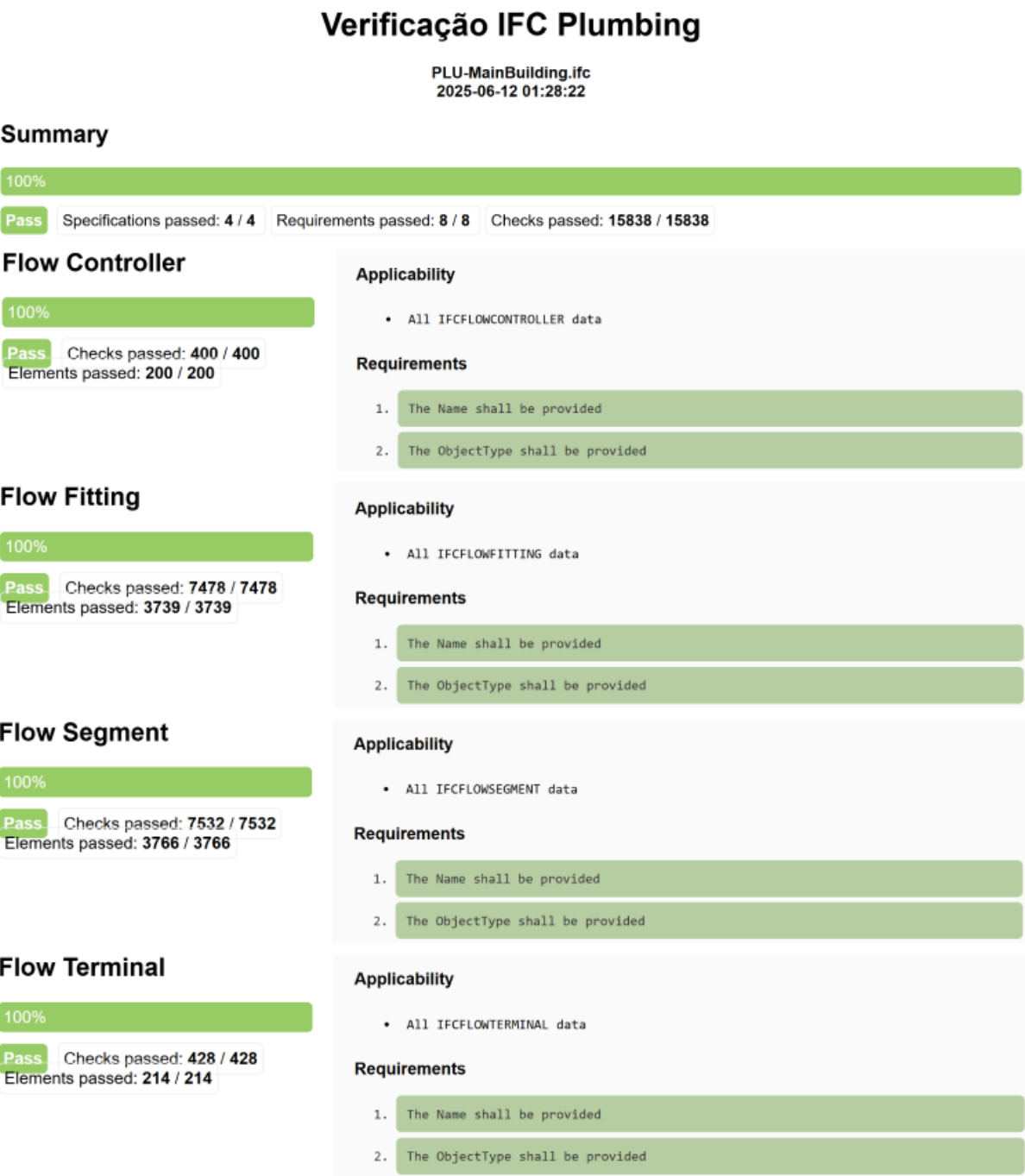


Figura 67 - Relatório HTML do Modelo de Sistemas Hidráulicos

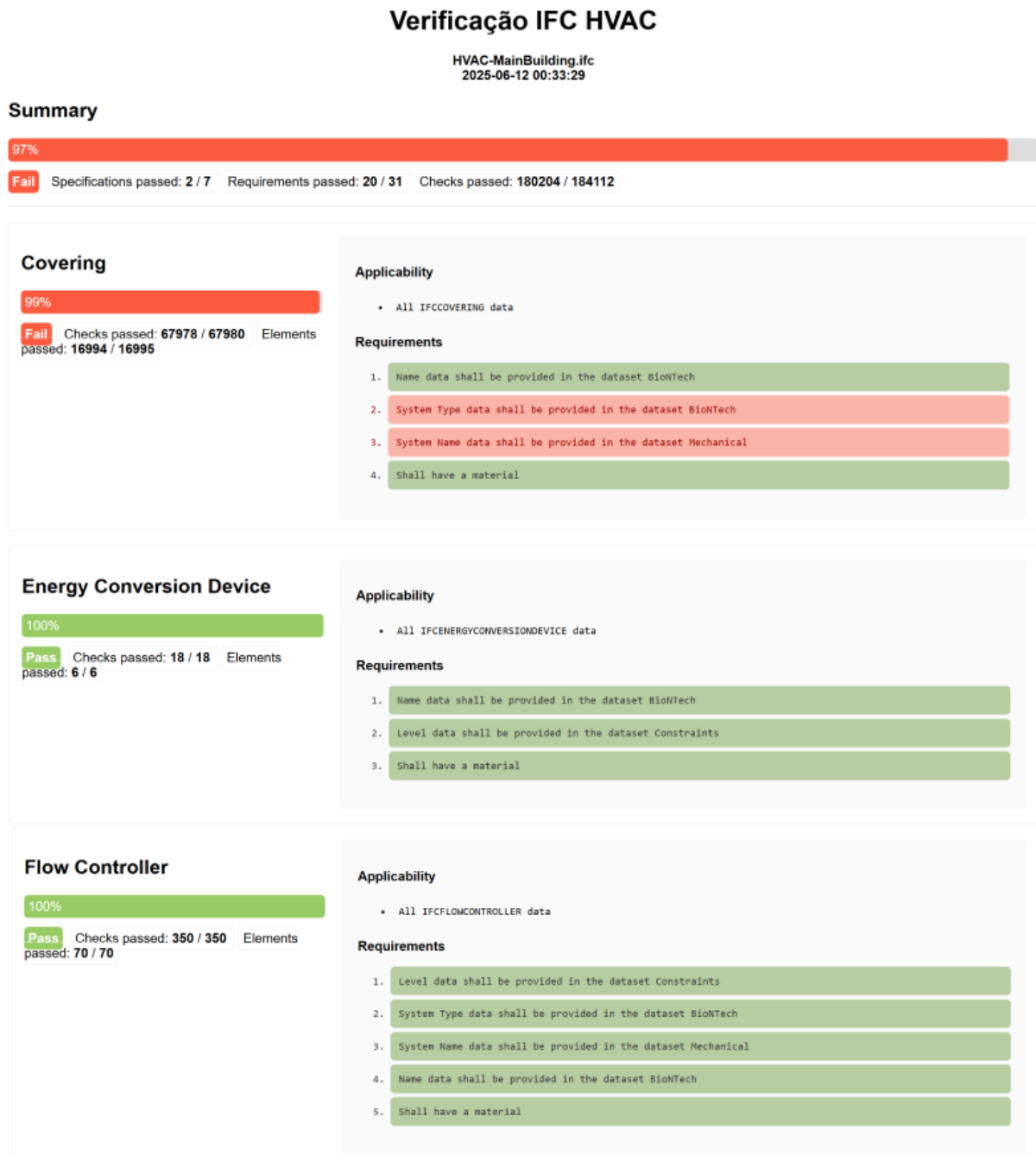


Figura 68 - Relatório HTML do Modelo HVAC (Parte 1)

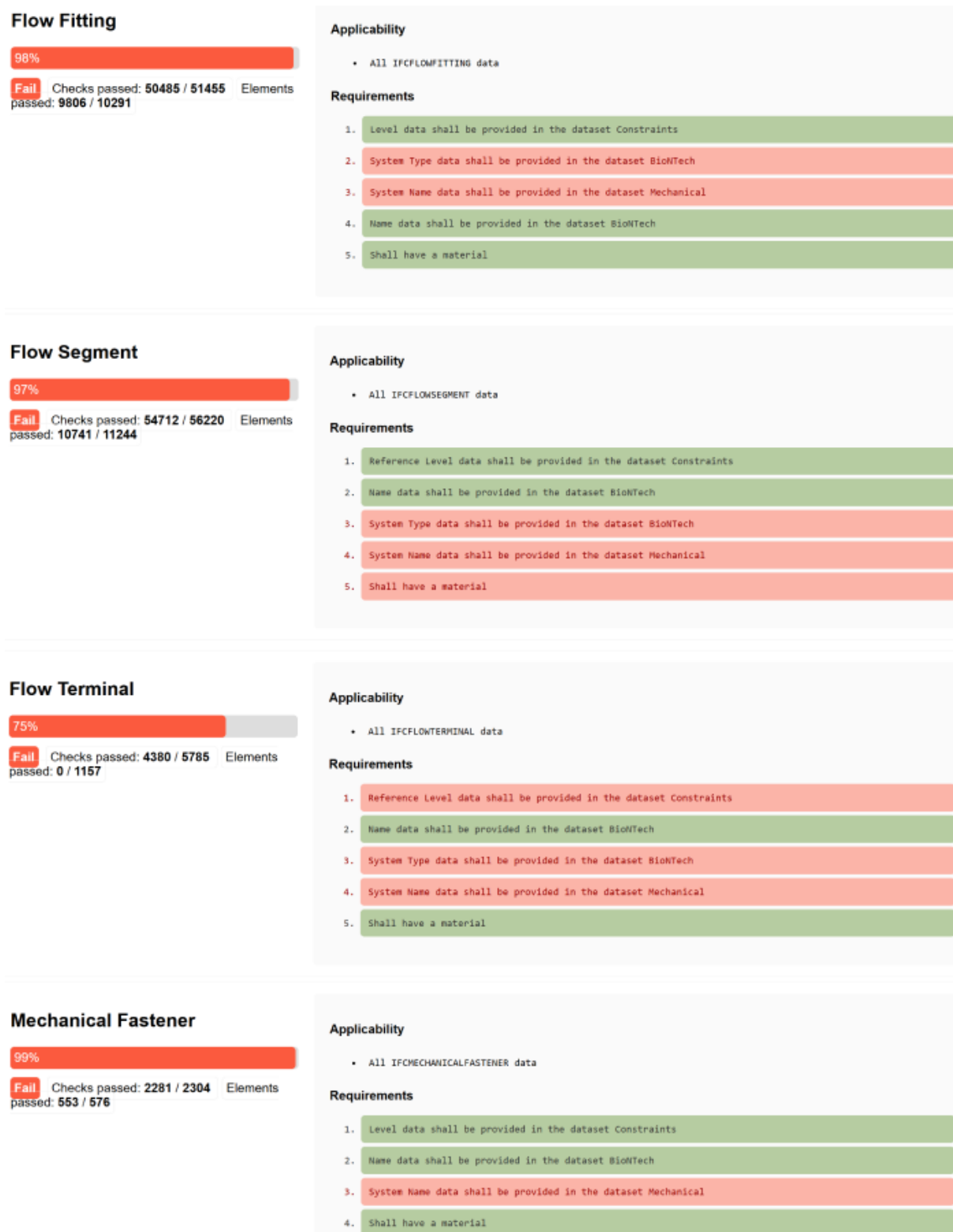


Figura 69 - Relatório HTML do Modelo HVAC (Parte 2)

Verificação IFC Electrical

ELE-MainBuilding.ifc
2025-06-12 01:36:17

Summary

98%

Fail

Specifications passed: 3 / 4

Requirements passed: 9 / 10

Checks passed: 5276 / 5352

Energy Conversion Device

100%

Pass

Checks passed: 60 / 60

Elements passed: 30 / 30

Applicability

- All IFCENERGYCONVERSIONDEVICE data

Requirements

- Name data shall be provided in the dataset BioNTech
- Shall have a material

Flow Fitting

100%

Pass

Checks passed: 1308 / 1308

Elements passed: 436 / 436

Applicability

- All IFCFLOWFITTING data

Requirements

- Name data shall be provided in the dataset BioNTech
- Service Type data shall be provided in the dataset BioNTech
- Shall have a material

Flow Segment

100%

Pass

Checks passed: 1104 / 1104

Elements passed: 552 / 552

Applicability

- All IFCFLOWSEGMENT data

Requirements

- Name data shall be provided in the dataset BioNTech
- Service Type data shall be provided in the dataset BioNTech

Flow Terminal

97%

Fail

Checks passed: 2804 / 2880

Elements passed: 884 / 960

Applicability

- All IFCFLOWTERMINAL data

Requirements

- Name data shall be provided in the dataset BioNTech
- Power data shall be provided in the dataset BioNTech
- Shall have a material

Figura 70 - Relatório HTML do Modelo Elétrico