Towards BIM/GIS interoperability: A theoretical framework and practical generation of spaces to support infrastructure Asset Management

Gareth Boyes

Thesis submitted for the degree of

Doctor of Philosophy

Department of Civil, Environmental and Geomatic Engineering

University College London

Supervisors:

Dr Claire Ellul, UCL

Dr Jan Boehm, UCL

Mr Daniel Irwin, Crossrail Ltd

30 November 2021

I, Gareth Boyes, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

The past ten years have seen the widespread adoption of Building Information Modelling (BIM) among both the Architectural, Engineering and Construction (AEC) and the Asset Management/ Facilities Management (AM/FM) communities. This has been driven by the use of digital information to support collaborative working and a vision for more efficient reuse of data. Within this context, spatial information is either held in a Geographic Information Systems (GIS) or as Computer-Aided Design (CAD) models in a Common Data Environment (CDE). However, these being heterogeneous systems, there are inevitable interoperability issues that result in poor integration.

For this thesis, the interoperability challenges were investigated within a case study to ask: **Can a better understanding of the conceptual and technical challenges to the integration of BIM and GIS provide improved support for the management of asset information in the context of a major infrastructure project?**

Within their respective fields, the terms BIM and GIS have acquired a range of accepted meanings, that do not align well with each other. A seven-level socio-technical framework is developed to harmonise concepts in spatial information systems. This framework is used to explore the interoperability gaps that must be resolved to enable design and construction information to be joined up with operational asset information.

The Crossrail GIS and BIM systems were used to investigate some of the interoperability challenges that arise during the design, construction and operation of an infrastructure asset. One particular challenge concerns a missing link between AM-based information and CAD-based geometry which hinders engineering assets from being located within the geometric model and preventing geospatial analysis.

A process is developed to link these CAD-based elements with AM-based assets using defined 3D spaces to locate assets. However, other interoperability challenges must first be overcome; firstly, the extraction, transformation and loading of geometry from CAD to GIS; secondly, the creation of an explicit representation of each 3D space from the implicit enclosing geometry. This thesis develops an implementation of the watershed transform algorithm to use real-world Crossrail geometry to generate voxelated interior spaces that can then be converted into a B-Rep mesh for use in 3D GIS.

The issues faced at the technical level in this case study provide insight into the differences that must also be addressed at the conceptual level. With this in mind, this thesis develops a Spatial Information System Framework to classify the nature of differences between BIM, GIS and other spatial information systems.

Impact Statement

This doctoral studentship was jointly funded by the Engineering and Physical Sciences Research Council (EPSRC) and Crossrail Limited.

The work of this thesis makes the following contributions:

C1 - Spatial Information System Framework – A requirement was identified for a standard framework that spans the Building Information Modelling (BIM) and Geographic Information Systems (GIS) domains. A comparative study was made of the existing frameworks found in general information systems theory alongside the definitions and standards that exist in BIM and GIS. A seven-level Spatial Information System Framework was developed and applied to the full range of socio-technical levels found across the fields of BIM and GIS. The Framework is useful for comparing BIM-based and GIS-based systems at individual levels with the aim of identifying system heterogeneities. Use of the Framework is enhanced by using diagrams to make graphical illustrations. It was found that a systematic understanding of the technical differences leads to a better understanding of the higher-level conceptual differences that may cause technical challenges.

C2 - Extraction, Transformation and Loading – A practical Extract-Transform-Load (ETL) workflow was developed for extracting CAD elements from *MicroStation*, transforming them into GIS-compliant geometry, and uploading them to geospatial database. Writing the workflow identified various bugs and idiosyncrasies in the tools and shortcomings in the data that required the development of workarounds solutions. It was identified that ETL operations must be accompanied with rigorous quality assurance procedures to audit the extraction of elements and monitor any deformations in geometry.

C3 - Spaces – A requirement for explicit watertight spaces was identified as a means of performing spatial queries on asset features. A practical workflow for creating watertight spaces from rudimentary CAD models using *watershed segmentation* was implemented in *Python*, including an algorithm for converting voxel-based spatial enumeration into Boundary Representation (B-Rep). The workflow was performed on real-world CAD models and observations made concerning choice of parameters and the suitability of the method when compared to the *floor plan extrusion* method. The suitability of the segmented spaces for performing spatial queries was also assessed in comparison to spaces created using the *floor plan extrusion* method.

C4 - Linking – UK and international standards require that information in Asset Information Models (AIMs) is linked with external systems such as GIS and Asset Information Management System (AIMS). A method for reducing the enormity of manual linking is proposed that would involve linking features based upon the name of the space in which they are located in conjunction with their semantic classification, although it has not yet been possible to implement or test the proposed method.

C5 - Asset Information Management Prior to Handover – The Spatial Information System Framework was used to review the Crossrail Technical Information Systems, including AIMS. From this review, it is can be observed that there are technical and conceptual differences between the CAD-based AIM and the AIMS database, including a technical difference between the Work Breakdown Structure (WBS) and Asset Breakdown Structure (ABS) caused by differences in requirements and a conceptual difference on how spaces are interpreted. The work in this thesis contributes towards a better understanding of these differences and the development of applicable solutions.

R1-R6 - Recommendations – This work is able to make recommendations to design engineers and asset managers for implementation in future infrastructure projects. It is recommended that the future owner/operator of infrastructure expand their Employer's Information Requirements (EIRs) to specify the need for quality assurance procedures for ETL operations, the need for explicitly represented 3D spaces and the need to resolve incompatibilities between WBS and ABS. Employers should be mindful of the relatively short shelf life of information stored in proprietary information formats when compared to the life of a built asset that may last centuries. It is also recommended that the Spatial Information System Framework be adopted as a means to describe the range of socio-technical levels that make up project information systems.

During the period of my doctoral studies, I have presented on the following occasions:

- Previous work from my MSc dissertation on BIM/GIS Integration was presented at the 23rd GIS Research UK Conference, 15-17 April 2015, Leeds, UK (Boyes, et al. 2015)
- Previous work from my MSc dissertation on BIM/GIS Integration was included in a chapter published in Advances in 3D Geoinformation, Lecture Notes in Geoinformation and Cartography (Ellul, et al. 2017)
- An early draft of the Spatial Information System Framework (C1) was presented to the 3D Information Management Domain Working Group at the 99th Open Geospatial Consortium Technical Committee, 22 June 2016, Dublin, Ireland
- An initial implementation of the ETL Workflow (C2) was presented to the London meeting of the FME World Tour 2017, 23 March 2017, London, UK (Boyes 2017)
- The proposed method for asset linking (C4) and initial results and observations made from the ETL workflow (C2) were presented to the 12th 3D GeoInfo Conference, 26-27 October 2017, Melbourne, Australia, which was published in Volume IV-4/W5 of the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences (Boyes, et al. 2017)

Acknowledgments

This doctorate would not have been possible without the generous support of Crossrail throughout the course of this research. Crossrail has kindly granted controlled access to its project information, thus providing the opportunity to study a real-world living Building Information Model. This access has included the technical training and support to access the technical information systems that hold the model. I am also grateful for their financial support in conjunction with co-funding from the Engineering and Physical Sciences Research Council (EPSRC).

I am incredibly thankful to Daniel Irwin, the Crossrail GIS Manager, who was instrumental in setting up this Collaborative Research Agreement with Crossrail Ltd. I want to thank him for his patience and good humour in answering my many technical questions regarding the *MicroStation*, *ProjectWise*, *FME Workbench* and *Oracle Spatial*. It has been a privilege to sit alongside the Crossrail GIS team and experience their enthusiasm and professionalism.

I cannot begin to express the magnitude of my debt to my principal supervisor, Claire Ellul, for her confidence in me over these last few years. Her supervision and tutelage have been invaluable as I have learned the skills required to read, research and write for a doctorate. We have had many exciting discussions about all things GIS, BIM and 3D. I am grateful for her patience in reading through my drafts and providing many pointers for discussion.

These acknowledgements must also include a big thank you to everyone in the open-source community who write and develop amazing applications and libraries. Where would this world be without the likes of Guido van Rossum (*ex-Python*), Travis Oliphant (*numpy*), Thomas Paviot (*pythonOCC*), Sean Gillies (*shapely*/*rtree*) and Mike Dawson-Haggerty (*Trimesh*), not to mention the thousands who dedicate themselves to making just one more commit as they strive towards perfection. Although we have never met, I feel as though we know each other.

To my parents, Neil and Barbara, to Danny and Suzie and my wonderful niblings, to Linda and the rest of my wider family and friends with special mention to Debbie Spain - thank you all for your encouragement which has made all the difference when things got tough.

To Alister - thank you for your love and patience and most of all for believing in me.

Contents

List of Figures

List of Tables

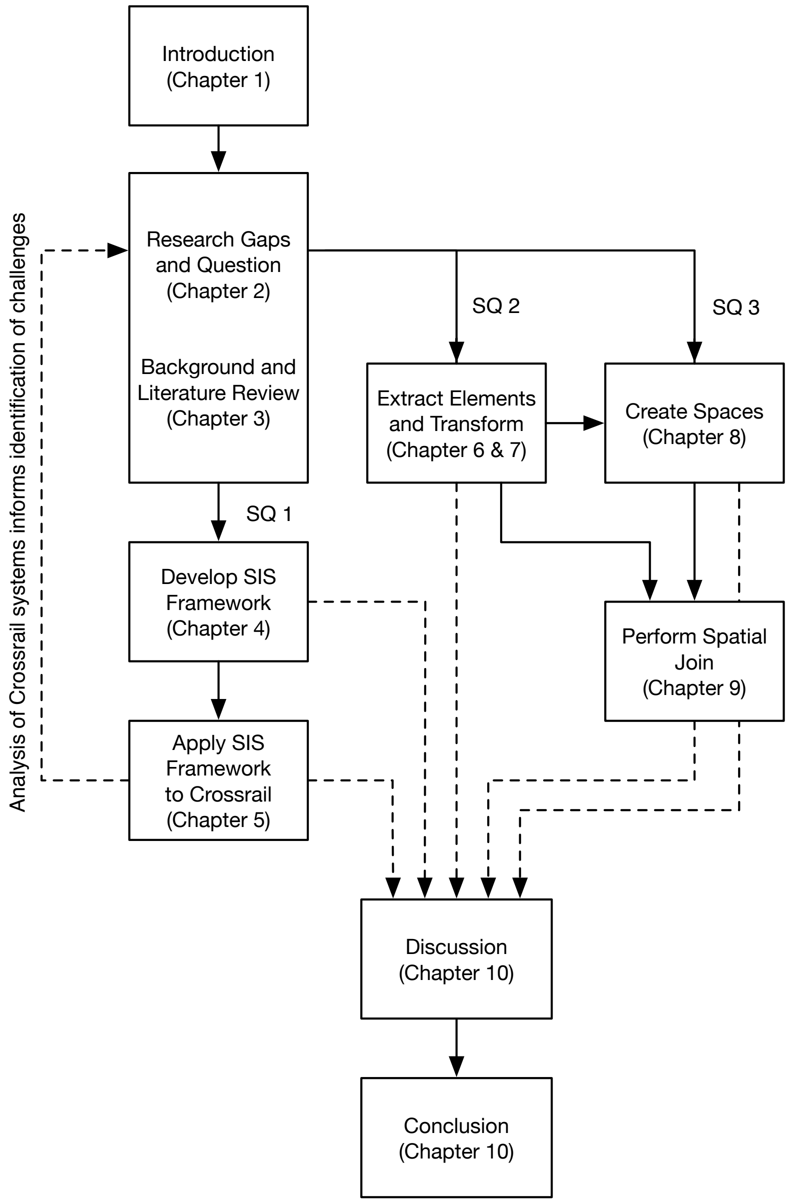
Acronyms

2.5D	Two-and-a-half-dimensional
2D	Two-dimensional
3D	Three-dimensional
ABS	Asset Breakdown Structure
AD4	Asset Data Dictionary Definition Document
ADE	Application Domain Extension
ADMM	Asset Data Management Manual
AEC	Architectural, Engineering and Construction
AIM	Asset Information Model
AIMS	Asset Information Management System
AIR	Asset Information Requirement
AM	Asset Management
AM/FM	Asset/Facilities Management
API	Application Programming Interface
AR	Augmented Reality
B-Rep	Boundary Representation
BCS	British Computer Society
BIM	Building Information Modelling
BNG	British National Grid
BSI	British Standards International
BTH	Broadgate Ticket Hall
CAD	Computer Aided Design
CAPEX	Capital Expenditure
CDE	Common Data Environment

CEN	Comité Européen de Normalisation (European Committee for Standardization)
CGAL	Computational Geometry Algorithms Library
CityGML	City Geographical Mark-up Language
COBie	Construction Operations Building Information Exchange
CPD	Continuing Professional Development
CPG	Cell and Portal Graph
CRS	Coordinate Reference System
CSG	Constructive Solid Geometry
CSV	Comma Space Value File
DEM	Digital Elevation Model
DF	Discretisation Factor
DGN	MicroStation Drawing File
DIHAS	Data, Information and Human Activity Systems
DWG	MicroStation Drawing File
EAM	Engineering Asset Management
EAMS	Enterprise Asset Management System
ECMS	Electronic CAD Management System
EDMS	Electronic Document Management System
EIR	Employer's Information Requirement
EM	Element Mapping
ETL	Extract-Transform-Load
FIFO	First-In, First-Out
FM	Facilities Management
FU	Functional Unit
GI System	Geographic Information System
GIS	Geographic Information Systems
GML	Geographical Markup Language
GNSS	Global Navigation Satellite System
GPU	Graphics Processing Unit
GUI	Graphical User Interface
GUID	Globally Unique Identifier

HAS	Human Activity System
HCI	Human Computer Interface
HDD	Hard Disk Drive
HVAC	Heating, Ventilation and Air Conditioning
ID	Identifier
IDBE	Integrated Digital Built Environment
IDM	Information Delivery Manual
IFC	Industry Foundation Classes
IGDS	Interactive Graphics Design System
IM	Infrastructure Manager
ISO	International Standards Organisation
ISO/TC	International Standards Organisation Technical Committee
JSON	Javascript Object Notation
LandInfra	Land and Infrastructure Conceptual Model Standard
LCIM	Levels of Conceptual Interoperability Model
LFB	London Fire Brigade
LM	Level of Development Mapping
LoD	Level of Detail
LU	London Underground
MDM	Master Data Model
MEP	Mechanical, Electrical and Plumbing
MES	Mile End Shaft
MVBA	MicroStation Visual Basic Application
MVD	Model View Definition
NURBS	Non-Uniform Rational B-Spline
OED	Oxford English Dictionary
OGC	Open Geospatial Consortium
OPEX	Operational Expenditure
ORDBMS	Object Relational Database Management System
PD	Perspective Definition
PFU	Primary Functional Unit
PIM	Project Information Model

PPP	Public-Private Partnership
RAM	Random Access Memory
RANSAC	Random Sample Consensus
RDBMS	Relational Database Management System
RDF	Resource Description Framework
RfL	Rail for London
RGS	Royal Geographic Society
RM-ODP	Reference Model - Open Distributed Processing
SDI	Spatial Data Infrastructure
SDK	Software Developer Kit
SDW	Spatial Data Warehouse
SKIZ	Skeleton of Influence Zone
SKP	SketchUp Model File
SOA	Service Orientated Architecture
SoS	System-of-Systems
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Sequential Query Language
SSD	Solid State Drive
STEP	STEP file format
STL	STL file format
SWG	Standards Working Group
SWOT	Strengths, Weaknesses, Opportunities and Threats
TCP/IP	Transmission Control Protocol/Internet Protocol
TfL	Transport for London
UML	Unified Modeling Language
URI	Uniform Resource Identifier
VR	Virtual Reality
WBS	Work Breakdown Structure
WFS	Web Feature Service
WKT	Well Known Text
XML	Extensible Markup Language
XSD	XML Schema Definition



Summary diagram of thesis chapter flow (Section 2.3)

1 Introduction

Over the past half-century, the world has become increasingly digital with electronic information ubiquitously woven into the fabric of civilisation. This digital revolution has had a transformative effect on every aspect of modern society, disrupting the traditional behaviours of individuals, businesses, organisations and communities.

The Architectural, Engineering and Construction (AEC) and Asset/Facilities Management (AM/FM) communities have traditionally been resistant to change and are still adapting to reap the full benefits of this digital revolution (Ahmed 2018). Perversely, although it is probably unfair to say that the implementation of information technology has only increased the capacity to generate larger volumes of paper-based documentation, this phenomenon has no doubt been an unfortunate outcome within the AEC and AM/FM sector.

To benefit from this digital revolution, the AEC community must use intelligent information models and processes that serve the needs of decision-makers in the design, construction and operation of infrastructure (Jernigan 2007). Only then can the right information get to the right people so that they can make better decisions with the potential to deliver more efficient construction and operation of infrastructure (HM Government 2015).

As part of this digital revolution, the development of intelligent building product models has been advancing for the best part of 50 years (Eastman 1975). Concurrent to this, the increased ability to process digital information has revolutionised the study into the complex spatial relationships that exist between objects (Eastman, et al. 2011). Although this technology has been freely available for some time, there have been non-technical barriers preventing wholesale adoption of technology. Indeed, convincing people to change is often the most significant challenge (Halttula, et al. 2016).

Writing a blog article entitled "*Comparing Pommes and Naranjas"*, Laiserin (2002) called upon the AEC community to abandon the term *Computer Aided Design (CAD)* and adopt in its place the phrase Building Information Modelling. He was able to convince *Autodesk*, *Bentley* and *Graphisoft*, between them the three major CAD software houses, to market their architectural modelling software using a common term to describe their products. The showcasing of BIM as a new product triggered the AEC community into understanding that BIM was no longer just a fancy demonstration of three-dimensional (3D) graphics but a tool that the industry needed to take seriously (Laiserin 2008).

The essence of BIM is that it provides a collaborative working environment with which to share intelligent, structured information. This collaborative environment enables stakeholders to communicate with each other using a 3D information model as a common reference (BIM Task Group 2014). The driving force behind the adoption of BIM is the expectation that digital technologies will foster more efficient collaboration resulting in financial savings. As an example, in 2015, the UK government made savings of £840m on construction projects, a significant proportion of which is attributed to adopting BIM (HM Government 2015). Furthermore, the adoption of BIM is a central pillar of the industry's strategy to reduce carbon emissions.

BIM is not the only software product capable of representing intelligent 3D information in a collaborative model. GIS is a spatial information technology that pre-dates the first building product modelling initiative by a decade (Longley, et al. 2005). Unlike the precursors to BIM, GIS quickly established itself among government and academia as a practical tool for managing geographic information. GIS developed to meet a different set of requirements to those for which BIM was developed. As a consequence, the information between the two systems is not immediately interoperable (Bishr 1998). As will be explained in Section 1.1.2, GIS has advantages over BIM for some tasks in the design, construction and operation of civil infrastructure.

The savings promised by BIM (in conjunction with other digital technologies such as GIS) are expected to manifest themselves in one of three ways (Eastman, et al. 2011):

a. Collaborative working and information sharing makes the design and construction process more efficient due to better communication and fewer mistakes delivering savings within the Capital Expenditure (CAPEX), or delivery, phase.

b. Secondly, these collaborative working methods enable the design team to construct more efficient buildings and infrastructure that will be more energy-efficient, demand less maintenance, and will be inherently better designed for future refitting and ultimate disposal. This approach delivers savings throughout the Operational Expenditure (OPEX) phase.

c. Finally, design and construction information will be handed over to the owner of the building. This information will then be used to populate the asset register to be used in support of the owner's asset management strategy. This approach delivers savings not just at initial stages of OPEX, but better information integration is expected to deliver savings throughout the life of the building or infrastructure asset.

Whereas it is apparent that BIM is working for the first two items in this list, the challenge of handing over information is preventing owner-operators from realising the benefit of BIM (López, et al. 2017; van Nederveen, et al. 2014). In the next section, this chapter will provide a brief introduction to BIM, GIS and Asset Management (AM) prior to carrying out an in-depth analysis of BIM and GIS using a specially developed Spatial Information System Framework in Chapter 4. This Framework will be used to explore the interoperability gaps that must be resolved to enable design and construction information to be joined up with operational asset information.

1.1 Introduction to BIM, GIS and Asset Management

1.1.1 Building Information Modelling

The term *Building Information Modelling* and its acronym *BIM*, as proposed by Laiserin (2002) in his seminal blog post, were only intended to refer to the next generation of architectural design software. However, as the use of BIM software advanced, it became apparent that traditional project management methods were holding back the true potential of using BIM.

Traditional design protocols focus on the delivery of 2D drawings; they, therefore, do not sanction the handing over the wealth of intelligent 3D data embedded in the model at the various project milestones. Valuable information is lost that invariably has to be created again. Furthermore, traditional project management was just not set up to manage the collaborative sharing of information between stakeholders.

With this advance in technology, the term *BIM* has evolved to mean the practice of designing and constructing built assets (Laiserin 2008) while using AEC modelling software. However, it is fair to say that this rapid evolution of terminology has fostered a degree of confusion as to its precise meaning.

As a consequence, the internationally accepted definition of *Building Information Modelling* has now evolved to mean "*the use of a shared digital representation of a built asset to facilitate design construction and operation processes to form a reliable basis for decisions"* (from ISO 19650-1:2018 (ISO 2018)). Following this definition, the term *Building Information Modelling* has been extended to include the range of human and technical activities required to represent a built asset as digital information and then the use of that information to design, construct and operate the built asset.

This shift in understanding of what BIM entails is best explained by the phrase "*BIG BIM, little bim"* which is taken from the title of the book written by Jernigan (2007). *Little bim* refers to the use of architectural modelling software to achieve a particular task; whereas, *BIG BIM* refers to the management of information to achieve strategic aims. BIM is a socio-technical system that extends from the hardware to the people who exploit it.

Within the practice of BIM, the digital representation of the built asset is referred to as the *information model*. Among organisations that follow the guidance in the UK and ISO specifications for information management using BIM (PAS 1192-3 and ISO 19650-1), the information model used in the delivery phase is distinguished from the one used in the operational phase, referring to the former as the Project Information Model (PIM) and the latter as the Asset Information Model (AIM) (BSI 2014; ISO 2018). However, many authorities and references (General Services Administration 2007; Isikdag, et al. 2008; Demian, et al. 2016) refer to the information model as the *Building Information Model*, colloquially referring to it as "*the BIM"*, thus adding to the confusion surrounding the definition of the term *BIM*.

The PIM and AIM contain both graphical and non-graphical, structured and unstructured information. Structured graphical information can be stored in either proprietary or open information formats. *Autodesk Revit* and the *Bentley MicroStation* BIM extensions are popular examples of proprietary formats for structured graphical information; while Industry Foundation Classes (IFC) provides an open format for the exchange of information between BIM applications.

1.1.2 Geographic Information Systems

The classic definition of a Geographic Information System (GI System) is that it is "*a computer-based information system that enables capture, modelling, storage, retrieval, sharing, manipulation, analysis and presentation of geographically referenced data"* (Worboys and Duckham 2004). A GI System is a particular case of a general information system that has two additional characteristics (Maguire 1991). The first requirement of a GI System is that it must be capable of storing object geometry, a characteristic that is shared with CAD applications such as *Bentley MicroStation*. The second, and arguably the fundamental, requirement of a GI System is that it must be able to integrate disparate data sources using spatial relationships thus permitting overlay operations and spatial analysis (Cowen 1988; Maguire 1991).

A third characteristic of a GI System is that it can handle geographic distances (i.e. distances that need to factor in the curvature of the earth). This characteristic gives a GI System an advantage over BIM software when used for civil infrastructure projects that extend over larger distances, as they do not need to be constrained by the limitations of using a projected coordinate system.

The Worboys and Duckham (2004) definition used above is a narrow description and emphasises the computer-based information system. However, as with all information systems, a GIS is more than just the hardware, software and information. Like BIM-based systems, a GIS is a socio-technical system that requires people to drive the hardware and the software and manage the information within (Reeve and Petch 1999).

In order to research the interaction between BIM and GIS, a more precise definition of the terms and associated jargon is required. Chapter 4 of this thesis will explore a generalised framework for understanding the socio-technical parts of the BIM and GIS. This exercise aims to identify similarities and differences and thus understand how they might be more closely integrated, which may then offer solutions to the closer integration of design and construction information and operational asset information.

1.1.3 Asset Management

Asset Management (AM) is a strategy used by senior management to realise the full value of an organisation's assets. The term is formally defined in the ISO standard for AM (ISO 55000:2014) as the "*coordinated activity of an organisation to realise value from its assets"* (ISO 2014), where an asset is further defined as any "*item, thing or entity that provides value to the organisation"* (ISO 2014).

The principles of AM can be applied to any asset, irrespective of whether the asset is tangible (i.e. physical) or intangible (e.g. financial investments). However, the handling and preservation of physical assets constitute a particular set of activities requiring a shift in managerial emphasis. References to AM in this thesis are intended as a reference to Engineering Asset Management (EAM), the branch of AM that concerns the management of physical assets (Amadi-Echendu, et al. 2010). This sub-division of asset management activities into Engineering Asset Management and Financial Asset Management does not diminish the principle that all physical assets are capital investments that require financial planning.

The importance of AM becomes manifestly apparent in safety-critical industries such as rail and other transport infrastructure. Following the privatisation of British Rail, the failure by the new Infrastructure Manager (IM), Railtrack, to establish proper oversight of EAM led to a spate of catastrophic accidents (Crompton and Jupe 2008). Under new contractual relationships, maintenance contractors were left to maintain the track and signalling infrastructure. Although in theory, the contractors assumed liability for substandard performance, the legal reality was less than concrete. Consequently, Railtrack lost control of the condition of the railway, while still remaining liable for the safety of the passengers who travelled upon it. This loss of control not only had disastrous consequences in terms of safety, but it would also lead to financial disaster. Blind to the actual condition of their assets, Railtrack was unable to raise the capital required to reinstate the assets to permit full operations. Railtrack collapsed and was taken into administration, necessitating the effective renationalisation of the infrastructure by Network Rail (Crompton and Jupe 2008).

On taking over from Railtrack, Network Rail adopted an AM-based approach to assess the condition of the national network. A national asset management policy oversaw the creation of a national database of all assets, enabling senior management to draw up a strategic plan that could target investment to where it was most needed to restore the railway to full operational strength (Scott 2015).

At about the same time, London Underground (LU) entered into a set of Public-Private Partnership (PPP) agreements with two private companies for the maintenance and renewal of its lines. Learning lessons from Network Rail, the service agreements required the companies to implement an AM strategy and make regular reports providing LU with oversight of its assets (Lloyd 2015). Despite this, one of the companies struggled to implement an effective reporting system and consequently was unable to provide the documentary evidence that was a precondition to drawing down funding from LU. Without the necessary funding, the company was unable to meet its obligations and was subsequently taken into administration. The functions of the contractor were then taken on by LU, who adopted the AM tools and systems and integrating them with its established decision-making procedures (Lloyd 2015).

At first sight, it would appear that the contracting out of AM responsibilities has been a failed experiment but these failures can alternatively be seen as an opportunity to learn from mistakes. Events have forced the IMs to adopt the principles of AM into their corporate cultures so that senior management now have a critical awareness of the condition of their estate, as well as an ability to make a realistic forecast of maintenance costs. Being able to have confidence in AM enables an organisation to target investment, knowing that it will directly improve operational performance (Lloyd 2015).

In summary, AM is critical to both operational safety and the financial viability of public transport infrastructure. Although AM principally concerns financial planning and management of risk, the philosophy is ultimately dependent upon being able to track every individual asset and know about its material condition. These fundamental tasks are referred to in Section 1.1.4 as *Managing Assets*.

1.1.4 Managing Assets

A distinction is often made between the day-to-day activity of *Managing Assets* and the management strategy that is formally referred to *Asset Management* (Dempsey 2017). The activity of *Managing Assets* includes tracking asset location, performance monitoring, maintenance, repair and replacement; maintaining databases and IT systems; it also includes the training of staff or contracting in qualified people to maintain, assess and operate assets. (Dempsey 2017)

The role of AM is to ask how assets are being effectively utilised to contribute to the organisational value. It adopts a holistic approach to funding mechanisms and manages risks that might devalue the organisation. AM asks whether collaboration within the organisation increases the value of the organisation or whether certain activities can be outsourced to sub-contractors or the supply chain. Whichever way the distinction is made, *Managing Assets* is an essential operational activity encompassed within the strategic aims of *Asset Management* (Zach 2017). Individuals at every level of an organisation have a role to perform with regard to AM.

1.1.5 Facilities Management

The AM activities of *Managing Assets* often overlap with the functions of Facilities Management (FM). So as to focus on core business functions, it is common for an organisation to outsource routine cleaning and maintenance tasks to an expert Facilities Management (FM) contractor. FM is defined as "*the organisational function which integrates people, place and process with the built environment with the purpose of improving the quality of life of people and the productivity of the core business"* (ISO 2017). This is in contrast to AM which has a broader focus on all assets that belong to organisation that provide value. FM is only focussed on maintaining those assets that support the primary business function by maintaining a quality environment where employees and others carry out their business (Kavrakov 2015; Mason 2017).

Within the literature on the use of BIM and GIS to support the activities of AM and FM (Section 3.8), it is evident that BIM and GIS provide a common set tools for *Managing Assets*. It is reasonable to assume that where a conclusion has been reached in the literature concerning the use of these common tools for FM, the same conclusion is likely to also apply to AM. It is for this reason that literature concerning FM is included in the literature review despite AM being the focus of this thesis.

Challenges relating to AM information handover have also prevailed at Crossrail, a major infrastructure construction project to build an underground metro system underneath the streets of London. The experience of preparing information for handover from construction to operation makes Crossrail an excellent case study for investigation with the potential to extend the observations made to future infrastructure projects.

1.2 The Crossrail Project

Crossrail is an ambitious civil engineering project being built to upgrade London's public transport infrastructure and make it fit for the 21st century (Taylor 2017). Although London was the first city in the world to build a metropolitan underground railway, much of the Victorian infrastructure lacks the capacity and reliability that is required for a modern city.

Crossrail aims to address these issues with a much-needed addition to the existing network.

The issues concerning capacity and reliability are not new. Indeed, planners and politicians have argued for better transport links across London for the best part of the last century (Crossrail 2016). More than 40 years ago, the London Rail Study recommended a direct route to connect the western suburban lines, that run into Paddington station, with the eastern suburban lines running into Liverpool Street station. (HM Government and Greater London Council 1974). From this proposal, the vision for what is now known as Crossrail was initially conceived. Over the next 30 years, the project was a victim of political deliberation and a reluctance to invest in infrastructure using public funds. Eventually in January 2002, the necessary political and economic conditions came into alignment, and the notional ambitions of the 1974 study became a tangible prospect. The Strategic Rail Authority and Transport for London (TfL) were given the authority to establish Crossrail with the remit to scope the project and steer the planning legislation through Parliament (Crossrail 2016).

Following a lengthy passage through Parliament, royal assent was received in July 2008, and the current route became established in law. With the green light given, contracts were put out to tender, and initial plans were worked up into detailed designs. Design quickly gave way to construction, and ground was first broken at Canary Wharf station on 15 May 2009, followed by the commencement of tunnelling in 2012 (Crossrail 2016).

The Crossrail project involves the design, construction and commissioning of 10 new stations, such as Bond Street Station in Figure 1.2 and 42 km of underground railway tunnels from Reading in the west, running underneath central London, and then branching out to Shenfield and Abbey Wood, located north and south of the Thames respectively (Crossrail 2015). The new railway line, to be called the *Elizabeth Line*, is illustrated as a *London Underground* diagram in Figure 1.1; this diagram has been annotated with the locations of the *Broadgate Ticket Hall (BTH)* and the Mile End Shaft, two sites that will be used as part of the case study.

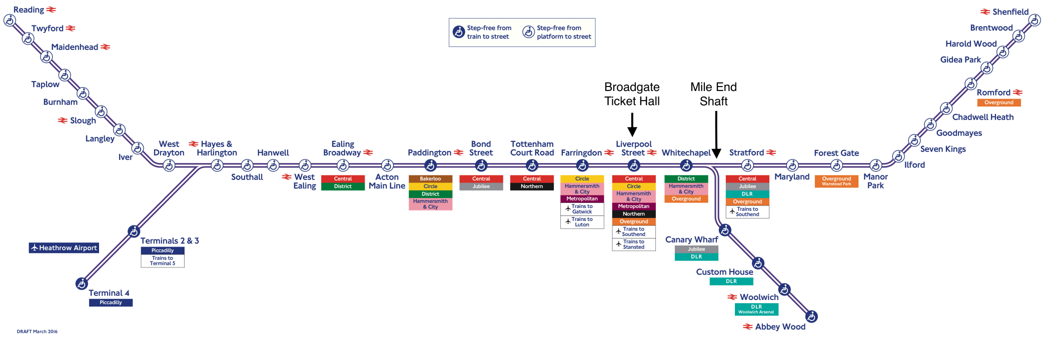


Figure 1.1: The Elizabeth Line with locations of case study sites - Broadgate Ticket Hall (BTH) and Mile End Shaft (MES)

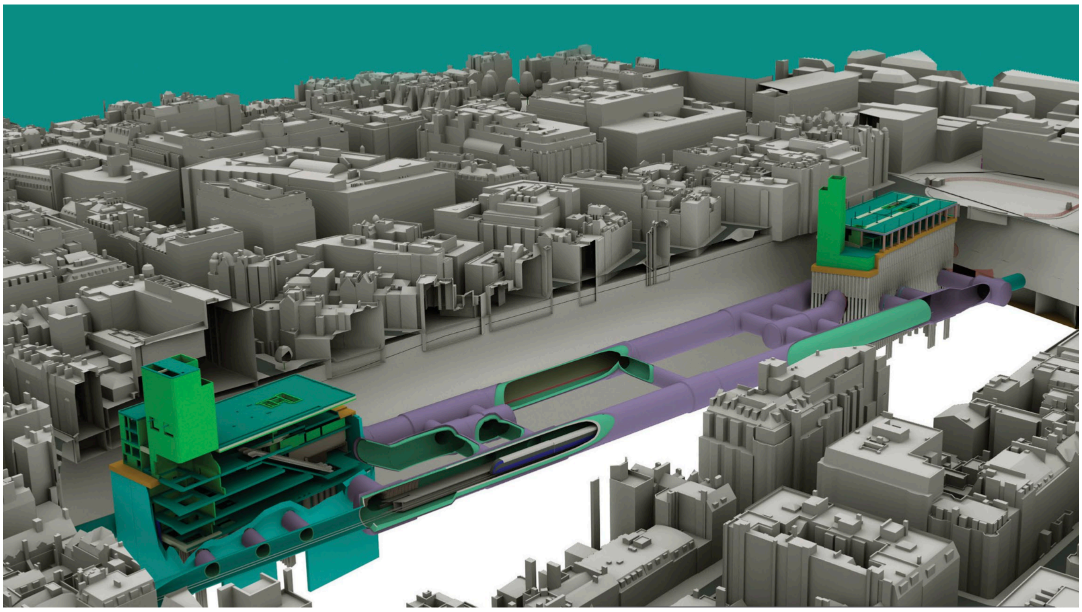


Figure 1.2: Bond Street Station (ESRI 2015)

To achieve an endeavour on this scale, the project has required the collaboration of over 100 contractors, typically represented by many of the global civil engineering companies such as Laing O'Rourke, BAM Nuttall, Kier, and Atkins, to name a few (Crossrail 2010). The majority of design work and planning has, therefore, been contracted out, with the role of the Crossrail central staff being to provide strategic direction and coordination. In addition, Crossrail acts as the technical design authority and a review board, with the task of approving design proposals.

Initially expected to be ready for 2018, the project has suffered major setbacks. The current expectation is for the railway to commence passenger operations through the central section sometime in 2022.

On completion of construction and commissioning, Crossrail will hand over ownership to Rail for London (RfL), a company wholly owned by TfL, who will take on the statutory role of an Infrastructure Manager (IM). TfL will then delegate the day-to-day operations of trains and stations to MTR Crossrail, the train operating company (Crossrail 2014).

Crossrail is committed to technological innovation at every level of the project. Consequently, despite the project pre-dating the UK Government BIM mandate, the Crossrail board has set a BIM objective that directs the project to "*set a world-class standard in creating and maintaining data for constructing, operating and maintaining railways by: exploiting the use of BIM by Crossrail, its contractors and suppliers; and the adoption of Crossrail information into future IM and operator systems*". With this, Crossrail has had the vision to build a virtual railway to exist in parallel with the physical railway (Taylor 2017).

As such, a single information model would be established at the start of the project to hold the planning information and provide a foundation with which to design and construct the railway. This information system would then be handed over to the operator to use during the operational life of the railway.

As part of this commitment to technological innovation, Crossrail has championed the establishment of a Common Data Environment (CDE) to provide a central location to access all CAD, GIS and AM information following the fundamental principles of BIM (see Chapter 5). The CAD system, GI System and AIMS are referred to as the Crossrail Technical Information Systems (Figure 1.3).

Ideally, each element in the CAD would share a common Globally Unique Identifier (GUID) with the corresponding element in AIMS. However, for reasons that will be explained in Section 1.2.1, the CAD system and AIMS were developed independently. As a consequence, there is a disconnect between the two information systems. It is not just a case of objects not sharing a common GUID but also a disconnect also arises in schemes of classification and schemes of aggregation, although the extent of the problem is not precisely known.

Despite all this, Crossrail hold an aspiration to link the information that they hold in their CAD systems with their GIS-based systems, and vice versa. In doing so, an integrated CDE will enable models created in design and construction phases to be used for different applications throughout the life of an infrastructure asset.

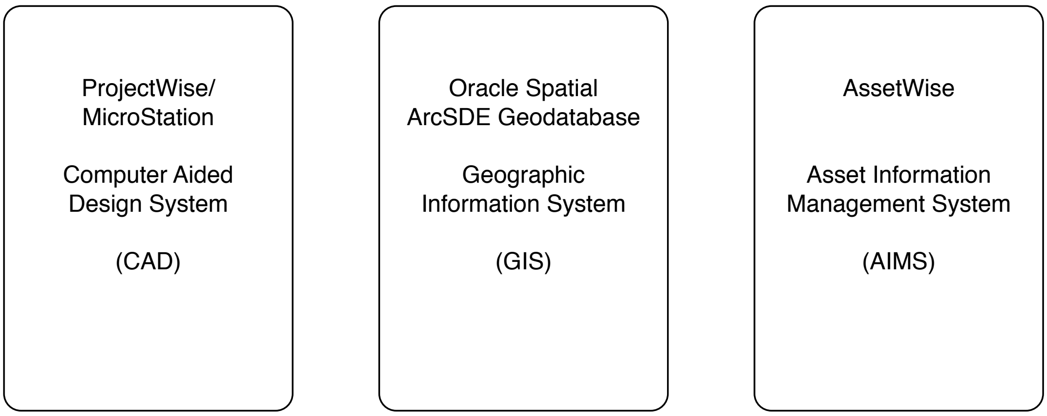


Figure 1.3: Crossrail Technical Information Systems

1.2.1 Asset Linking

Initially Crossrail had hoped that the information held in the *ProjectWise/MicroStation* CAD systems (Section 5.1.3.1), could be linked to the information in the asset register implemented in AIMS (Section 5.1.3.3). This linking would provide geographic and engineering context as assets would be cross-referenced back to drawings and models used in the design and construction phase. Despite best intentions, AIMS was populated with asset information without the opportunity to reference the CAD information. The decision to move away from storing asset information in 3D CAD models was chosen to avoid bloated model files and facilitate interoperability between CAD formats (Taylor 2017). Consequently, interoperability and integration issues prevent Crossrail from linking the information locked away in CAD-based systems with information in the AM-based systems. Section 5.1.3.3 provides further elaboration on this decision.

The IM has not provided Crossrail with any requirements of how this linked information will be used (Crossrail 2013). In the absence of a specific requirement, Crossrail chose to follow the UK *specification for information management for the operational phase of assets using BIM* (PAS 1192-3) which requires a relationship between the two information systems to be established (BSI 2014). PAS 1192-3 was part of the UK suite of standards which was published to support the UK BIM mandate (Section 3.4.2) and has now been adopted by International Standards Organisation (ISO) as the ISO 19650 suite. Although not mandated to abide by this requirement, Crossrail has set an aspirational benchmark to comply with PAS 1192-3 as closely as possible.

Crossrail had been expecting to use the tool that was being developed by *Bentley* called *Asset Painter* (Bentley Systems 2012). This tool was to be available for asset managers to select CAD elements within *MicroStation* and tag them with the corresponding Asset Identifier (ID) in AIMS; however, development of the tool was dropped. It was explained that although the tool was capable of linking together elements that resided in the same file, the software development team encountered technical difficulties in cases where an asset consisted of parts saved in more than one file. Without the commercially-developed tool, Crossrail looked to developing an in-house solution that might assist in establishing explicit relationships between AIMS assets and the CAD elements in *MicroStation*.

Four approaches for linking asset items in AIMS with geometry elements (Figure 1.4) were developed in conjunction with the Crossrail technical information team[1]. The suitability of each method is summarised in Table 1.1 considering issues relating to the time required to input links into the system, the link quality, and the potential for practical implementation.

[1] An initial report on these four approaches is published in a preceding paper within the scope of this thesis (Boyes, et al. 2017).

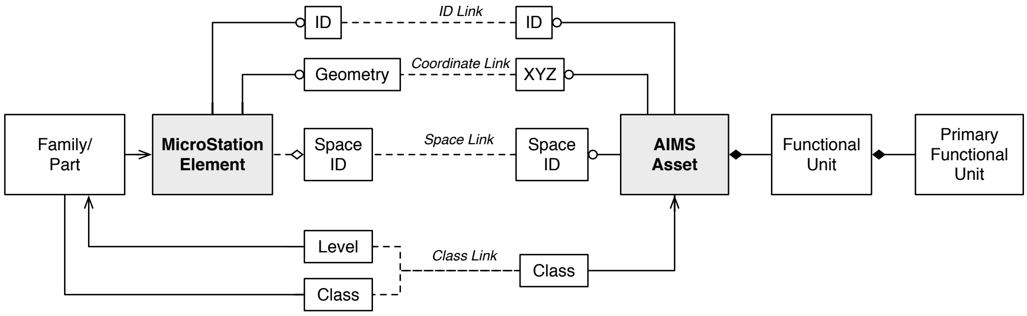


Figure 1.4: Relationships between AIMS assets and CAD elements

ID link approach – This approach establishes an explicit link between the IDs of AIMS assets and *MicroStation* elements, and it had been hoped to use the previously described *Asset Painter* tool to put it into effect. Without the tool, the work-hours required to input a defined link for every AIMS asset is prohibitively expensive, given that there are millions of assets. Additionally, such a link does not possess referential integrity and is susceptible to being broken should the element ID change, which may happen if elements are deleted and recreated during editing.

Coordinate link approach – This approach would provide an explicit spatial relationship between a fixed point on the asset and the geometry of the related *MicroStation* element, and involves attributing each asset in AIMS with an (*x,y,z*) coordinate to its approximate location in *London Survey Grid* as the asset is commissioned or inspected. The inputting of point coordinates representing each asset was trialled by the Crossrail technical team but was abandoned due to the unacceptably slow speed of entry. It was also found to be unreliable due to the introduction of human errors during the input process, and the trial was discontinued in preference for the next method of linking.

Space link approach – The third approach would establish an inferred spatial relationship based upon the identity of the space attributed to the asset and the space(s) enclosing the CAD element. Populating AIMS using this approach is faster because it is simpler for data-inputers to enter the name of a space rather than coordinates. It should be noted that long elements (e.g. pipes) may pass through multiple spaces, creating a *1:M* relationship between asset and spaces. Any practical method using this approach will need to handle such relationships.

Table 1.1: Summary of approaches to asset linking

Approach	Description	Cost / Quality / Practicality
All approaches		Requirement to identify suitable method for 1:N, M:1 and M:N relationships.
ID link	Manual attribution of AIMS asset with Element ID.	Susceptible to broken links if ID changes. Prohibitively time consuming.
Coord link	Manual attribution of AIMS asset with coord. Nearest neighbour spatial join on CAD element.	Human error reliability issues. Prohibitively time consuming.
Space link	Attribute asset with named space (Exisiting requirement). Identify space of CAD element using ”contains” join with space geometry. Join on space name.	Expensive to generate 3D space geometry. Dependent on quality of 3D representation and spatial join. Elements can belong to more than one space. Dependent on timely production of space names in project plan.
Class link	Join on class name.	Differences in classification schema. Impractical due to sheer volume of joins.
Combined approach	Combined join of space name and class name.	As per Space/Class link. Manual resolution of multiple joins still required.

Class link approach – The final approach is an inferred semantic relationship based on schemes of classification. For this approach to work, a method of mapping schema needs to be established because slightly different schemes of classification are used between the CAD model and the AIMS (Section 5.2.3.5). Establishing links using this method is impractical due to the sheer volume of returned joins when performing a query.

Combined approach – While these four approaches individually do not provide a workable method, it may be possible to develop a practical workflow for linking CAD elements and AIMS assets by combining approaches. By using the *Class link* approach in conjunction with the *Space link* approach, it may just be possible to reduce the number of unsolved links to a sufficiently manageable task for a human asset manager to perform manually.

The research in this thesis is motivated by a desire to identify relationships between different representations of the same asset where they are modelled in diverse information systems. It is proposed that the approaches described in this section can be used, in isolation or combination, to identify these desired relationships.

Initial investigations by Crossrail have revealed that there will be problems in establishing these relationships due to differences in how assets and elements are aggregated and in how they are classified within the various enterprise systems. These problems will arise because the systems have been built to achieve different objectives; i.e. the CAD model has been constructed to support design and construction, while the AIMS model has been created to support operations, maintenance and the management of assets. It is also the case that the true extent of the problems cannot be assessed without first conducting exploratory analysis.

The opportunity to explore the extent of the problem and application of these linking approaches was hindered within the period of this research due to sufficient data not then being available. The scope of this research is undeterred by these problems as there are still a variety of interoperability challenges that must first be overcome before investigation into the linking challenge can be started, as will be discussed in the next chapter.

1.3 Summary

In summary, having an effective AM strategy is hugely important within the rail industry to ensure that organisations have the capital to maintain their infrastructure and keep their passengers and staff safe. BIM and GIS promise to deliver considerable savings not only in designing and constructing infrastructure but also in establishing and maintaining the register of individual assets that supports the Asset Management Strategy. However, it is apparent that the savings promised by these information systems may not be realised due to there being insufficient interoperability between them, as is the case at Crossrail. Understanding why this information is trapped within the Crossrail systems, as well as what can be done to avoid future interoperability issues, would be a great benefit to the field of BIM.

2 Research Gaps and Questions

It is evident from Chapter 1 that Asset Management (AM) has a critical role to play in the construction and operation of infrastructure. For an effective AM strategy, it is important to keep an accurate and extensive register of assets in an Asset Information Management System (AIMS). The adoption of Building Information Modelling (BIM) has promised to deliver considerable savings in the Architectural, Engineering and Construction (AEC) and Asset/Facilities Management (AM/FM) sectors, not least in the handover of information from the delivery phase to the operational phase in a format that can be used to populate a register of assets. However, the transfer of asset information is fraught with interoperability issues.

This research was born out from a presumption that the potential of BIM may be unlocked if certain challenges relating to the integration of BIM and Geographic Information Systems (GIS) can first be overcome; these challenges may be conceptual, geometric, semantic or technical in nature. To address this presumption, this research has had unique access to the Crossrail Technical Information Systems and the opportunity to investigate how interoperability issues have affected a major infrastructure construction project, thus allowing both theoretical and practical issues to be explored.

This chapter will provide a high-level overview of the main challenges, with further detail of the gaps from which they are derived being given in later chapters.

2.1 Research Gaps

2.1.1 Socio-Technical Nature of Spatial Information Systems

The design, construction and operational management of built assets requires pulling spatial information (BIM, GIS, AM) together into a collaborative working environment. This ability to work collaboratively is described in Chapter 1 as the essence of BIM. The ability to access asset information from a single source not only informs operational managers on where to carry out maintenance and repair, but also provides senior management with a critical awareness of the condition of their infrastructure, as described in Section 1.1.3 and Section 1.1.4. For this to work, information must either be considered to be contained within a single information system or a System-of-Systems (SoS) (Section 3.2).

In Section 1.1.2, the definition of a Geographic Information System (GI System) was considered from both a technical perspective (Worboys and Duckham 2004) and a wider perspective that includes human interaction (Reeve and Petch 1999) referred to as the social-technical perspective. This differentiation was also considered in Section 1.1.1 with particular reference to the phrase "*BIG BIM, little bim*" (Jernigan 2007).

A review of the current literature on BIM/GIS integration (Section 3.6) reveals a sizeable collection of material published on the how to solve challenges from a technical perspective. However, the review struggled to unearth any studies into how integration works at the social end of the socio-technical spectrum.

There is a bounty of literature containing frameworks for describing the socio-technical make-up of BIM and GIS as separate entities (see Section 4.1 and Section 4.2). However, the terms used within these frameworks are mismatched (Section 4.2) which hinders the task of comparing BIM and GIS side-by-side. If a universal framework were to be available, it might be easier to consider the different levels of interoperability between BIM and GIS.

RG1 - The lack of a universal Spatial Information System framework and the missing literature on BIM/GIS integration across the full range of socio-technical levels is identified as a research gap.

2.1.2 Linking Asset Information

Initial conversations with Crossrail revealed three particular challenges hindering the integration of its Technical Information Systems. The principal challenge, initially referred to in Section 1.2.1, concerns the absence of any relationship between the Computer Aided Design (CAD) elements that have been created in *ProjectWise/MicroStation* and the assets to AIMS. The benefits of linking information will be discussed in Section 10.2.1.

The second challenge concerns the extraction of CAD elements from *ProjectWise/MicroStation* and loading them into a GI System while keeping their unique *MicroStation Element Identifiers (IDs)*. Maintaining *Element IDs* across the two systems is important for referencing back to the information source and is the basis of linking elements between CAD and AIMS.

The third challenge concerns a difference in the methods used to locate elements and assets. The positions of CAD elements in *MicroStation* are fully located using three-dimensional (3D) solid geometry described using a common Coordinate Reference System (CRS). In contrast, the assets in AIMS are only located in terms of the name of space in which they are located. This issue is further compounded by the absence of explicit 3D representations that describe the geometry of the named spaces. In the majority of built assets, named spaces can be adequately described using extruded two-dimensional (2D) floor plans. Extruded floor plans may be suitable for ordinary buildings such as offices, schools and hospitals, but the complex space geometry found within a typical underground metro system station may not be adequately represented in Two-and-a-half-dimensional (2.5D). This complex space geometry is illustrated in Figure 2.1. Figure (a) shows the side elevation of a tunnel underneath a staircase rise in which a control panel is located. It is ambiguous from the 2D floor plan (c) as to whether the control panel is located in the staircase or the tunnel. Figure (b) shows a control panel located on the concave side-wall of a tunnel, but the 2D floor plan (d) incorrectly locates the control panel outside of the tunnel.

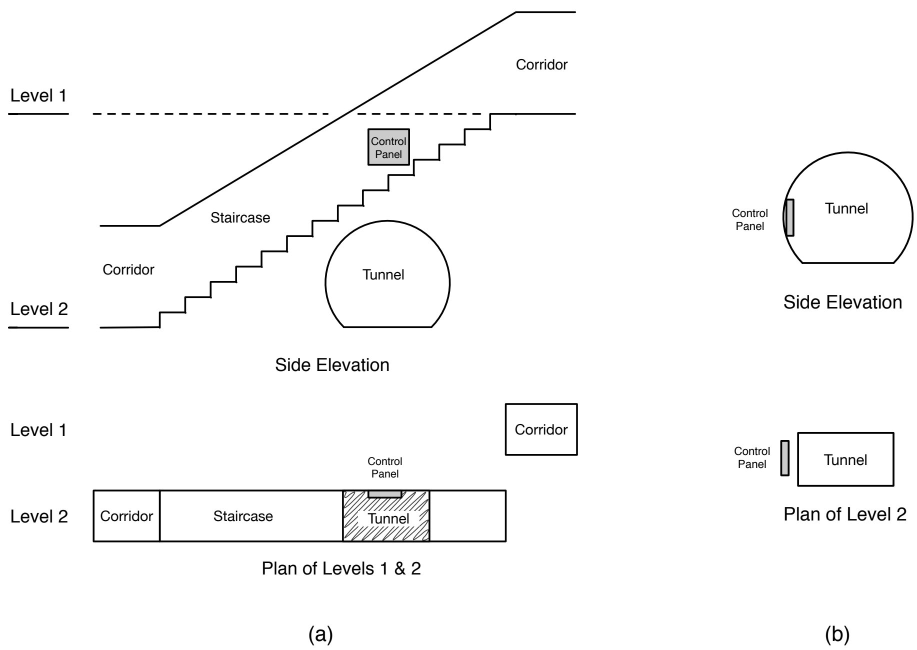


Figure 2.1: Extruded floor plan misrepresentations

It should be noted that the elements and assets in these two systems are structured differently. Elements in the federated CAD model are structured according to a Work Breakdown Structure (WBS) that supports the design and construction activities of the project. In contrast, assets in AIMS are structured according to an Asset Breakdown Structure (ABS) that supports the future IM's Asset Management Strategy. An example of this can be found in the Crossrail model of the Mile End Shaft head house (Irwin 2016). An exterior wall of the head house is modelled in MicroStation as individual bricks (Figure 2.2a), however this level of detail is unnecessary for the purposes of AM and so the wall is modelled as a single asset (Figure 2.2b). This example can be extended to illustrate a hypothetical situation whereby different owners are responsible for maintaining each side of the same wall, such that the wall is represented by two assets in AIMS (Figure 2.2c). This *m:n* relationship is explored further in Section 5.2.3.4.

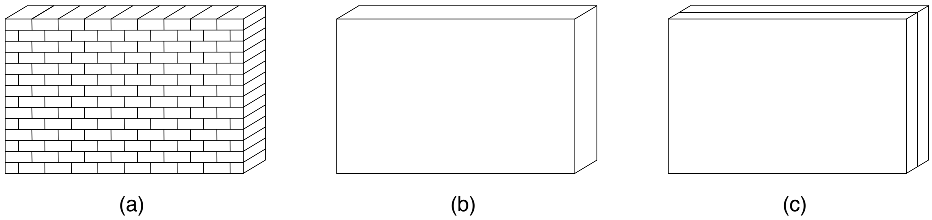


Figure 2.2: Representation of Mile End Shaft head house exterior wall (a) as individual bricks (b) as a single asset and (c) hypothetically as two separately maintained assets

The difference in the philosophy behind how assets and elements are hierarchically structured leads to incompatible systems for identifying individual components; this frustrated the development of a consistent method for linking components. One of the motivations of this research is to investigate the incompatibility of these hierarchical structures and its implications on establishing a practical linking protocol. Unfortunately, it is beyond the scope of this thesis to investigate these schemes of hierarchical aggregation fully due to a comprehensive asset dataset not being fully available within the timescale of this research project.

However, there are at least three prerequisite steps to be completed before attempting to implement a practical workflow based on the combined approach described in Section 1.2.1. These three prerequisite steps are illustrated in Figure 2.3.

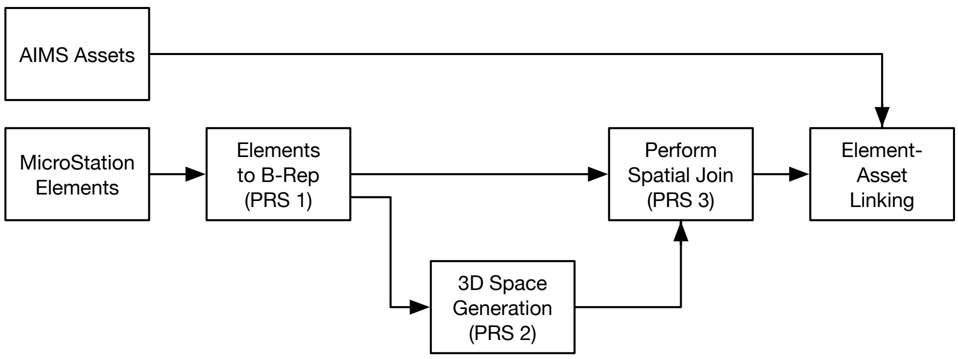


Figure 2.3: Prerequisite steps required for proposed linking method

Working backwards, the last prerequisite step (PRS 3) of a practical workflow is to determine the named location of every CAD element. The named location of every CAD element can be automatically determined by performing a spatial query with a 3D representation of the relevant spaces. Those elements contained within the geometry of a space will be attributed with the name of the space.

If not already provided by the design contractors, the penultimate prerequisite step (PRS 2) is, therefore, to generate explicit 3D representations of the named spaces in a format suitable for performing a spatial join. In order to perform a spatial join, the space must be geometrically represented as a valid watertight solid.

Before either of these prerequisite steps can be accomplished, it is necessary to extract the CAD elements from *MicroStation* and transform them into a geometry format suitable for further operations (PRS 1); in the case of this research the creation of 3D spaces and spatial operations require the elements to be in a Boundary Representation (B-Rep) planar mesh.

RG2 - The second research gap is identified as the scarcity of published methods for establishing links between geometric elements in federated CAD models with assets registered in independent enterprise information systems.

2.2 Research Questions

The gaps identified in Section 2.1 relate to the conceptual and technical barriers that are currently preventing the interoperation of information used for BIM with other information systems with particular reference to AM. These gaps should take in the organisational context in which these challenges are being addressed. This thesis will therefore ask the following question:

> **RQ - Can a better understanding of the conceptual and technical challenges to the integration of BIM and GIS provide improved support for the management of asset information in the context of a major infrastructure project?**

This question will be addressed in part by asking three supporting questions that arise from the identified gaps in Section 2.1.

Firstly, as BIM and GIS are both spatial information systems, the harmonisation of the various socio-technical levels by which they are constituted could be used to provide a better understanding of how interoperation occurs at each level. If the schemes of classification in Figure 3.18 are compared with the Levels of Conceptual Interoperability Model (LCIM) (see Section 3.3.1), there appears some correlation. From this pattern, a framework could be developed for the efficient exploration and articulation of similarities and differences between information systems at a range of levels from technical to conceptual.

This correlation could also provide the basis for exploring whether the socio-technical hierarchies in BIM, GIS and AIMS share a common structure. If this is the case, the correlation may be sufficient to develop a novel framework to guide future studies into information systems and the interoperability issues that arise between them as a result of heterogeneities in the modelling process. This potential use leads to asking the question:

> **SQ 1 - Can a novel Spatial Information System Framework be developed to identify and classify interoperability issues that currently hinder the management of asset information in the context of a major infrastructure project?**

This doctoral research is motivated by an aspiration not only to explore the integration of BIM and GIS for AM from a theoretical perspective but also to address the missing relationships between the federated CAD model and AIMS from a technical perspective. Reaching this goal is beyond the scope of this thesis due to the full datasets not being available during the period of this research. However, there is a great opportunity to research the prerequisite steps described in Section 2.1.2 needed to develop a practical workflow for linking elements in federated CAD models with assets in an AIMS using a combined approach (Section 1.2.1).

The second supporting question concerns the first of the prerequisite steps (PRS 1) described in Section 2.1.2. Practical attempts to extract CAD elements from *MicroStation* and load them as transformed features into a 3D GIS were found to be fraught with difficulties. As well as being a practical requirement for the next steps (PRS 2 and PRS 3), finding a workflow will provide a case study into the technical challenges that are hindering interoperability. A second supporting question asks:

> **SQ 2 - What challenges are frustrating reliable ETL (extract, transform, load) operations between CAD-based design models and a GIS-based spatial data warehouse? How can these be overcome?**

The second prerequisite step in the workflow (PRS 2) concerns the creation of 3D spaces. These spaces are represented as 2D floor plans in the federated *MicroStation* CAD model but they are not explicitly represented in 3D. 3D spaces can be created by extruding floor plans, however, the method does not extend to the diversity of space geometry found in an underground metro system railway stations, which contain tunnelled corridors, platforms, sloping floors and escalator rises as illustrated in Figure 2.1.

This lack of explicit 3D geometrical representations of named spaces that are suitable for spatial queries, combined with the scarcity of appropriate methods for generating suitable spaces for an underground metro system station, is also a technical challenge hindering interoperability. A third supporting question asks:

> **SQ 3 - What methods exist for modelling complex spaces to locate assets using 3D spatial analysis? Can these be implemented?**

2.3 Summary of Thesis Structure

These research questions will be addressed in the chapters that follow as illustrated in Section 2.3. Chapter 3 will start with a background account into the nature of systems in general, information systems in particular, and the interoperability issues that may arise between them. The literature review will start in earnest by reading into how BIM and AM are being used in the context of infrastructure. This will be followed by a study into how spaces are conceptualised within a built asset and how they can be modelled to describe the location of assets. The literature review will finish with an investigation into how the watershed transform algorithm can be used to partition spaces with internal boundaries located at the portals between spaces.

The development of a harmonised Spatial Information Systems framework, as proposed in Section 2.2, will be described in Chapter 4 and applied to the various approaches towards BIM/GIS integration identified in the literature review.

In Chapter 5, this novel Spatial Information Systems framework will be applied to the Technical Information Systems that have been put in place by Crossrail. This framework will first be used to provide a general description of the diverse systems that make up the Crossrail Technical Information Systems and then identify the interoperability issues at various levels. The chapter will expand on the issue initially identified above in Section 2.1.2 concerning the linking of elements in the federated CAD model and the assets in the AIMS.

Chapter 6 will consider the methods that exist for extracting information held in *MicroStation* and transforming it into a format suitable for use with GIS tools. The success of the Extract-Transform-Load (ETL) operation on the case study models will then be described in Chapter 7.

Chapter 8 will consider various methods for generating explicit 3D watertight spaces and will qualitatively assess the resulting output from implemented workflows. These resulting spaces will then be used for performing spatial queries using GIS tools and the results of these spatial operations will described in Chapter 9.

The results and findings from this experimental work will be considered in Chapter 10 in the context of the research gaps and questions identified above in Section 2.1 and Section 2.2. The contributions from this thesis will be drawn up in Chapter 11 in conjunction with a list of recommendations to be considered by practitioners, along with a list of recomendations for further research.

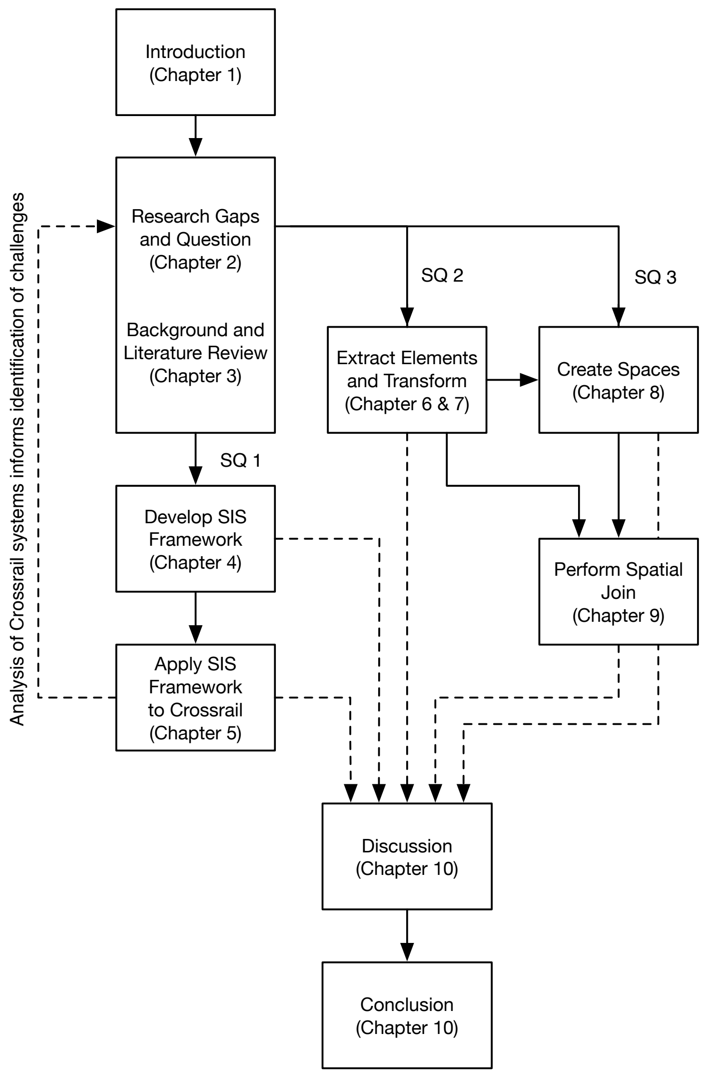


Figure 2.4: Summary diagram of thesis chapter flow

3 Background and Literature Review

Chapter 1 and Chapter 2 introduced the challenges faced by the Architectural, Engineering and Construction (AEC) and Asset/Facilities Management (AM/FM) communities that concern the linking of spatial information held in different information systems in the context of Building Information Modelling. Poor interoperability hinders integration and limits the potential value of information. Data must be manually adapted or recreated across different systems, or else it will forever be left in isolation.

The first half of this chapter (Section 3.1 to Section 3.3) will provide a background account into the general character of systems and the peculiar nature of information systems and modelling systems as well as literature on systems interoperability.

There are occasional references in this thesis to the UK BIM scene. Section 3.4 will provide some background into the UK Government BIM Mandate and the UK *Levels of BIM Maturity*. This section will also set out the 1192 suite of standards and the ISO 19650 suite that has replaced it.

This thesis also makes references to different types of solid geometry used in Building Information Modelling (BIM) and three-dimensional (3D) Geographic Information Systems (GIS). Section 3.5 will, therefore, explain the predominant methods for representing the geometry of solids and watertight spaces and comment on its advantages and disadvantages.

The second half of the chapter will review the current literature that is relevant to BIM/GIS interoperability and integration (Section 3.6), the use of BIM and GIS with regard to infrastructure (Section 3.7), the concept of spaces within the built environment

(Section 3.9) and review of the methods that can be used to generate spaces (Section 3.10), including a specific review of the *watershed transform* (Section 3.11).

3.1 Systems

Seeing the world as an infinite collection of systems is fundamental to observing its interconnectedness. General systems theory states that a system is a set of interconnected components (Meadows 2008) that work together to function in a way that the individual parts would not be able to achieve by themselves (Maier 1998). Systems can be observed not only in the physical realm such as nature and engineering but also in broader sociological contexts such as business and economics (Meadows 2008). Breaking down complex systems into interconnected components is beneficial as it enables a better understanding of how the system works, and therefore allows complex systems to be developed in a more manageable way (Succar 2009).

As an example, a building can be considered as a structural system with columns and beams working together to provide support (Sacks, et al. 2004). The structural system within the building interacts with the other building systems such as the Heating, Ventilation and Air Conditioning (HVAC) system, electrical systems, pedestrian navigation systems. From a geographic perspective, the natural terrain around the same building can be observed as a hydrological system that catches rainfall and drains water away in rivers (Soille and Ansoult 1990). In each case, the structure and terrain are systems in their own right regardless of their subsequent representation in an information system.

Patterns of behaviour can be observed in systems, from which a simplified explanation can be proposed and validated. The development of rudimentary models, capable of representing the original features, permits the original system to be analysed in ways that would otherwise be impossible (Batty 2009). For example, the maximum loading of the structural system can be calculated without destroying the building; or, the defences required to contain a river in flood can be determined based solely on model calculations.

The concept of a system is not restricted to physical objects but also covers social behaviour. BIM (Sackey, et al. 2015) and GIS (Maguire 1991) are both socio-technical systems, and consequently, humans have an important role in each. As such, the processes of designing, constructing, and operating buildings are systems in their own right. It follows that developing representative models of these processes are an essential subject to be analysed, communicated and managed (Cerovsek 2011).

It is also possible to view a complex system as being a collection of interacting systems. Where a component system is effectively autonomous from the collective system it happens to be part of, the parent system may be referred to as a System-of-Systems (SoS) (Maier 1998). In these situations, it is important to understand how the component systems interoperate as a collective whole. In this respect, Cerovsek (2011) considered BIM to be an SoS in the context of developing a multi-standpoint BIM framework.

No matter how rudimentary, every system contains and transmits information (Meadows 2008). The semiotic representation of information and its subsequent communication is itself a distinct class of system that is referred to as an information system. As well as being general systems, BIM and GIS both belong to this class of information system (Eastman, et al. 2011; Maguire 1991). As well as being able to model real-world systems, information systems are also able to keep records, communicate knowledge, and support decision-making (Maguire 1991). However, to classify BIM and GIS as just being information systems is an over-simplification because, on closer inspection, it is possible to perceive a multitude of systems of varying types in operation (Cerovsek 2011).

3.2 Information Systems

An information system is classically defined as a group of components that interact to produce information; those components being the hardware, the software, data, procedures and people (Kroenke, et al. 2013). Although this straightforward breakdown begins to explain the basic nature of an Information System, it is somewhat over-simplified and does little to explain how the components interact (Beynon-Davies 2010).

From an analytical standpoint, frameworks are useful for breaking down the system for gaining a better understanding of how the components of an information system interact (Succar 2009). Two frameworks found in general information system literature include the Reference Model - Open Distributed Processing (RM-ODP) published by the International Standards Organisation (ISO) and the framework published by Beynon-Davies (2010), which for convenience will be referred to in this thesis as the Data, Information and Human Activity Systems (DIHAS) framework.

3.2.1 Data, Information and Human Activity Systems Framework

In the DIHAS framework the Information System is just one system within a group of three interacting systems, i.e. the *Activity System*, the *Information System*, and the *Data System*. The framework places emphasis on the *Activity System* or *Human Activity System (HAS)*, i.e. broader goals and objectives of the organisation that uses the Information System.

The HAS is considered to be a series of *performative acts* carried out in pursuit of those objectives. For these *performative acts* to be performed across the organisation, participants must communicate instructions and report situations, which are referred to as *informative acts*. The communication of the *informative act* requires the conduct of a *formative act* that will represent the information as data. The transformation of *performative acts* into data is necessary because data can exist in a communicable form, that is capable of storage and transmission. The data can later be reinterpreted as information for use within the HAS. In this way, *performative acts* encode human intention as information and that information, reinterpreted at a different time and place, triggers an expected act of behaviour.

Beynon-Davies (2010) sets out his framework in the context of a system of communication between humans. However, there are instances where a system, such as a Geographic Information System (GI System), can capture information (e.g. remote sensing) and act on information (e.g. drone navigation) without any human interaction. Either BIM and GIS are not information systems, or it is necessary to expand the framework to include systems that contain intelligent information without necessarily involving social interaction.

Together these *performative*, *informative* and *formative* acts make up a series of steps that are referred to as the *Semiotic Ladder* (Beynon-Davies 2010), as illustrated in Figure 3.1. Each step on the ladder is a separate branch of semiotics; *pragmatics* concerns the purpose of messages, *semantics* concerns their meaning, *syntactics* concerns their structure, and *empirics* governs the physical form of words. The *Activity*, *Information* and *Data Systems* are demarcated according to their class of activity and their position on the semiotic ladder. Furthermore, each system is a socio-technical system in its own right, i.e. a human interaction may be present at each level.

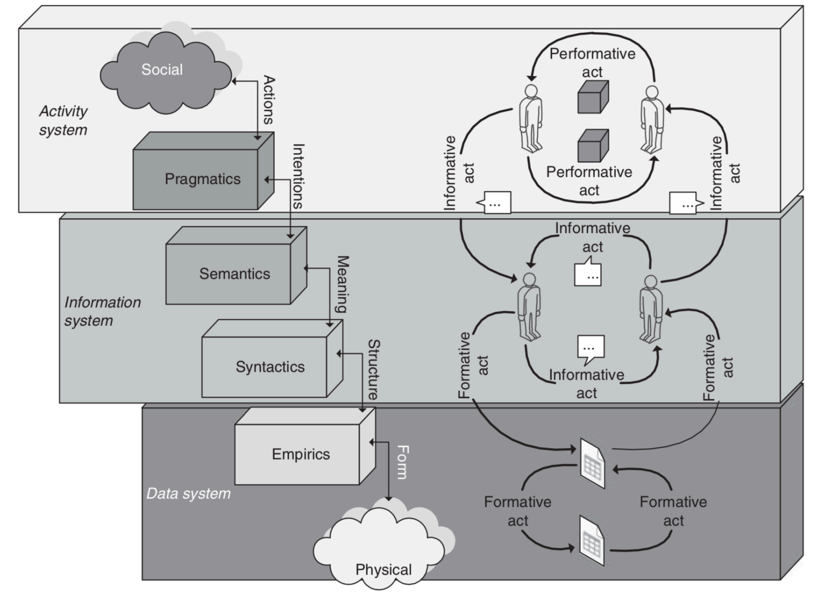


Figure 3.1: Interaction of Data, Information and Human Activity Systems (Beynon-Davies 2010)

The DIHAS framework, however, is just one way of understanding how information is used within a broader objective-achieving HAS. Instead of rigidly separating a system into Human Activity, Information and Data Systems, a less rigid approach is to observe the broader system as a series of viewpoints as can be found in the RM-ODP.

3.2.2 RM-ODP Framework

The HAS, as described in the previous subsection, can also be referred to as an *enterprise architecture framework*, thus emphasising the purpose of the organisation. The RM-ODP has been developed to describe how a particular enterprise architecture is made up, and it provides a common language of terminology to describe a system from the perspective of different five viewpoints. In ISO/IEC 10746-1:1998, each viewpoint describes the system architecture in relation to a particular set of concerns (ISO 1998), summarised as follows.

- **Enterprise viewpoint** - concerning the purpose, scope and policies governing the activities of the specified system within the organisation of which it is a part.
- **Information viewpoint** - concerns the kinds of information handled by the system and constraints on the use and interpretation of that information.
- **Computational iewpoint** - concerning the functional decomposition of the system into a set of distributed objects that interact at interfaces.
- **Engineering viewpoint** - pertains to the infrastructure required to support system distribution.
- **Technology viewpoint** - regarding the choice of technology to support system distribution.

In Section 4.4.6.2, the concept of an Spatial Data Infrastructure (SDI) will be explained as a complex system established for managing geospatial information from diverse sources across a large geographic area. Hjelmager, et al. (2008) used the Enterprise and Information viewpoints of the RM-ODP to describe the characteristics of an SDI. Béjar, et al. (2012) used the language of the RM-ODP Enterprise viewpoint to define an SDI as a federation of autonomous communities and for describing the objectives of the SDI.

The Comité Européen de Normalisation (European Committee for Standardization) (CEN) uses the RM-ODP framework to structure their technical report CEN/TR 15449-1:2012 setting out the requirements for SDI in Europe (CEN 2012). It publishes the specification in five parts, each part corresponding to the five viewpoints of the RM-ODP.

Both the DIHAS and RM-ODP frameworks provide a means for exposing the complex socio-technical arrangement of information systems. The DIHAS framework uses the semiotic ladder to achieve this aim, in contrast to the series of viewpoints used by the RM-ODP. The next section will review the process of how modellers use a hierarchical method to abstract models from physical and social real-world phenomena.

3.2.3 Modelling Theory

One advantage of a systems worldview is that it enables less critical aspects to be stripped away leaving only the most relevant characteristics. As such, a model can be considered to be a generalised abstraction of a system. However, just as the word *system* is a somewhat nebulous term, there are also many competing definitions for the term *model*.

According to the Oxford English Dictionary (OED), a *model* can mean both a "*3D representation of a projected or existing structure"* (Oxford English Dictionary 2019), i.e. a physical likeness or scale model, and also a "*set of designs (plans, sections, elevations) for a projected or existing building"* (Oxford English Dictionary 2019), i.e. a virtual representation. In other contexts, a model is understood to be "*a simplified or idealised description or concept that is put forward as a basis for theoretical or empirical understanding, or for calculations and predictions"* (Oxford English Dictionary 2019). These interpretations are often distinguished in engineering literature by the terms *product model* (Dado, et al. 2010) referring to a virtual reproduction of a physical object or product, and the term *process model* (Abdelhady and Jones 2014) as the workflow that is developed to design and construct the product.

The *Oxford Dictionary of Computer Science* describes a *data model* as an "*abstract model of some real-world situation or domain or interest about which information is to be held"* (Butterfield, et al. 2019). It goes on to state that a *data model* is constructed from a set of logical abstractions (_ibid.). There appears to be no definitive distinction in literature between the terms *information model* and *data model* except that the latter term tends to be more closely associated with structured information held in a relational database. In contrast, an *information model* can include unstructured information (BSI 2013).

All the above definitions of the word *model*[1] are relevant in the context of spatial modelling systems. In summary, models are representations of real-world systems that can be used for better understanding, as a means for storage and communication, and as a surrogate that can be analysed and experimented on.

The process of creating a model follows an iterative cycle involving analysis, design, implementation and maintenance (Worboys 1994). The principles that underpin this process are universal, although the terms used to describe the process vary. Two different terminologies that are common in geospatial modelling are the terms used in the waterfall model (Worboys and Duckham 2004) and the terms used in the documentation publihsed by the Open Geospatial Consortium (OGC) (Herring 1999).

The first set of terms described in this section are well known in the geospatial community following an article published by Worboys (1994) and have since been refined in a book published by Worboys and Duckham (2004). The logical progression of abstractions illustrated in Figure 3.2 is sometimes referred to as the waterfall model and consists of five levels: the *Application Domain*, *Application Domain Model*, *Conceptual Computational Model*, the *Logical Computational Model* and the *Physical Model*.

Application Domain – The *Application Domain* is synonymous with the real world, and as such is the subject that a modeller intends to model. The *Application Domain* includes the visible and the invisible, along with the known and the unknown, as well as the concrete and the abstract. The *Application Domain* is subject to all physical laws, all interaction with nature and all human interference. It is infinitely complex and incapable of perfect replication.

Application Domain Model – The *Application Domain Model* is the sum of human knowledge of the world that the modeller intends to model. The modeller identifies and documents those real-world phenomena that are relevant to the requirements of the model. The model is documented in natural language and is used as a standard reference for subsequent abstractions. Real-world spatial phenomena are identified either as discrete features (i.e. roads) or a continuous field (i.e. temperature) (Wise 2010).

[1] An element of linguistic confusion can arise in the use of the words *system* and *model*. The confusion is compounded in that a *system model* can be abstracted using a *modelling system*. Meanwhile, the *system model* can be stored in an *information system* under the governance of a *management system*. Furthermore, *system models* can be developed for the *modelling system*, the *information system* and the *management system*.

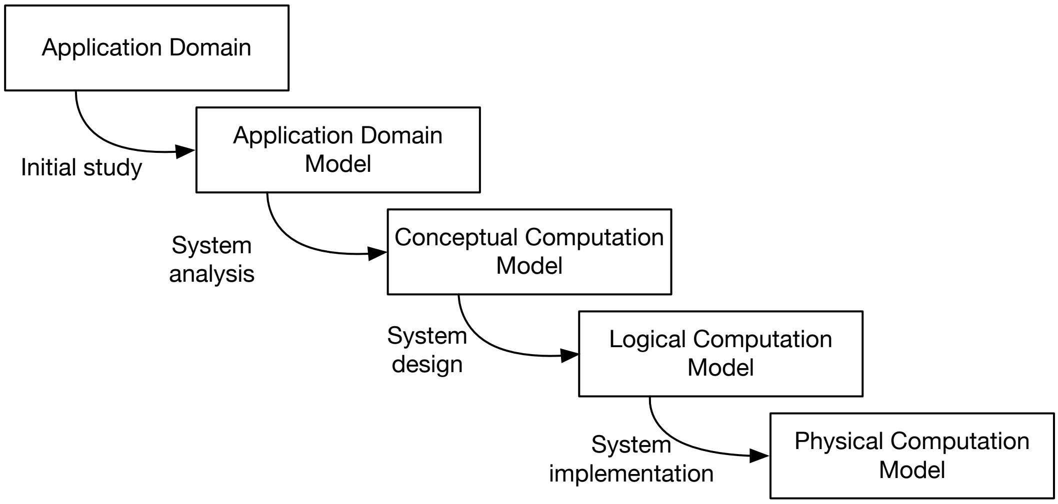


Figure 3.2: Waterfall model of system development (Worboys and Duckham 2004)

Conceptual Computational Model – Through a process of system analysis, the modeller simplifies and generalises the domain model into a conceptual framework (the *Conceptual Computation Model*) that is easier to understand and represent. In this way, discrete features can be represented using a lower-dimensional entity; for example, on a map, a 3D city can be represented using either a 2D polygon or a 0D point. Alternatively, the modeller may choose to use a regular array of representative values such as pixels or voxels (Wise 2010). Features can also be categorised into taxonomical classes, and the same features can be decomposed into component parts. Standardised schema are often available for the modeller to refer to within a particular domain.

Logical Computational Model – The next stage of the modelling process is to codify the conceptual model into a format that can be implemented in software; this is the *Logical Compuational Model*. The spatial modeller can choose between vector or raster representation formats. The modeller can also refer to an existing Implementation Specification with which to implement the model within a particular technology platform. For example, a *Conceptual Computational Model* developed as Entity-Relationship diagram is translated into SQL. If no appropriate schema or implementation specification exists, then the modeller will need to design a customised implementation or develop a workaround.

Physical Model – The final stage is to compile the logical computation model on hardware and load the model into digital memory. Once in digital form, software can be executed on the model to perform analysis, to share information or to visualise the model on a monitor.

Wise (2010) identified that in normal circumstances the waterfall abstraction process goes through two iterations. On the first iteration, standard generic models and formats are developed and published, which are then adapted on the second iteration to meet specific requirements. Modellers developing bespoke models are constrained by the abstractions adopted from the generic models.

The OGC follow a similar modelling approach as the waterfall model for documenting their standard model specifications but use a different set of terms. All OGC standards are documented using a prescribed succession of three abstracted models referred to as: the *Essential Model*; the *Abstract Model*; and the *Specification Model* (Herring 1999). This succession correlates to the *Application Domain Model*, the *Conceptual Computational Model* and the *Logical Computational Model* described by Worboys and Duckham (2004). The *Essential Model* and *Abstract Model* for a particular topic are published as the *Abstract Specification*, and the *Specification Models* are published as *Implementation Specifications* (Herring 1999).

The OGC has developed essential and abstract models that document an agreed conceptualisation of real-world phenomena as defined geographic features which it has published as an *Abstract Specification* (Kottman and Reed 2009). The OGC has adopted an approach that assumes nine layers of abstraction between the real world and a feature collection model features as illustrated in Figure 3.3. The first five layers, from *Real World* to *Conceptual World* to *Geospatial World* to *Dimensional World* to *Project World*, involve abstraction from the real world. The final four layers, from *Point World* to *Geometry World* to *Feature World* to *Feature Collection World*, concern the semiotic representation of the phenomena in the *Project World* as features. These final four layers, although included in the *Essential Model* by the OGC, have mathematical and symbolic characteristics for which they could instead be included in the *Conceptual Computational Model* described by Worboys and Duckham (2004).

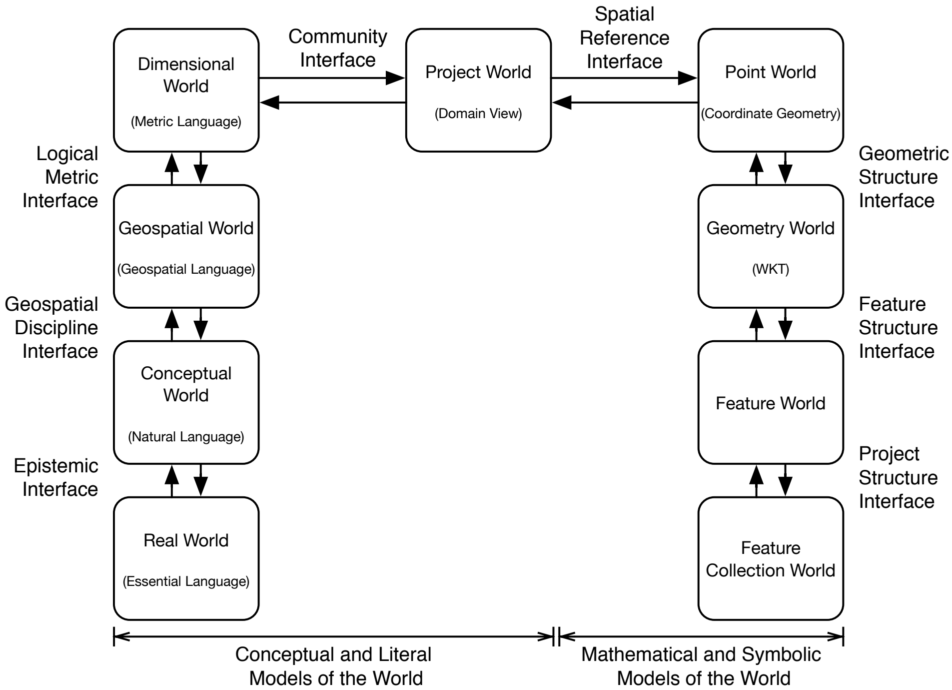


Figure 3.3: OGC nine layers of abstraction (Kottman and Reed 2009)

Bishr (1998) adapted the OGC essential model (at that time still only a draft) by dividing the model into two halves as illustrated in see Figure 3.4. The conceptual and literal models are referred to as the *Discipline Perception World*, and the mathematical and symbolic models take on the term *Project World* (which had previously been used to refer to a layer in its own right). Bishr (1998) considered the OGC *Essential Model* as being a cognitive process of breaking down reality, only to build it back up in a representative form. Each level in the *Discipline Projection World* corresponds to a level in the *Project World*.

Regardless of the framework followed, at each stage of the abstraction process, the modeller must maintain a balance between describing the real world as accurately as required so as to be fit for purpose, while at the same time taking on the cost of creating and maintaining a model that exceeds its requirement (Bishr 1998). A model is only an abstraction of reality represented with varying levels of completeness and as such each step is always an imperfect representation (Peuquet 1984). The modeller needs to accept compromise and understand the consequences of making those compromises.

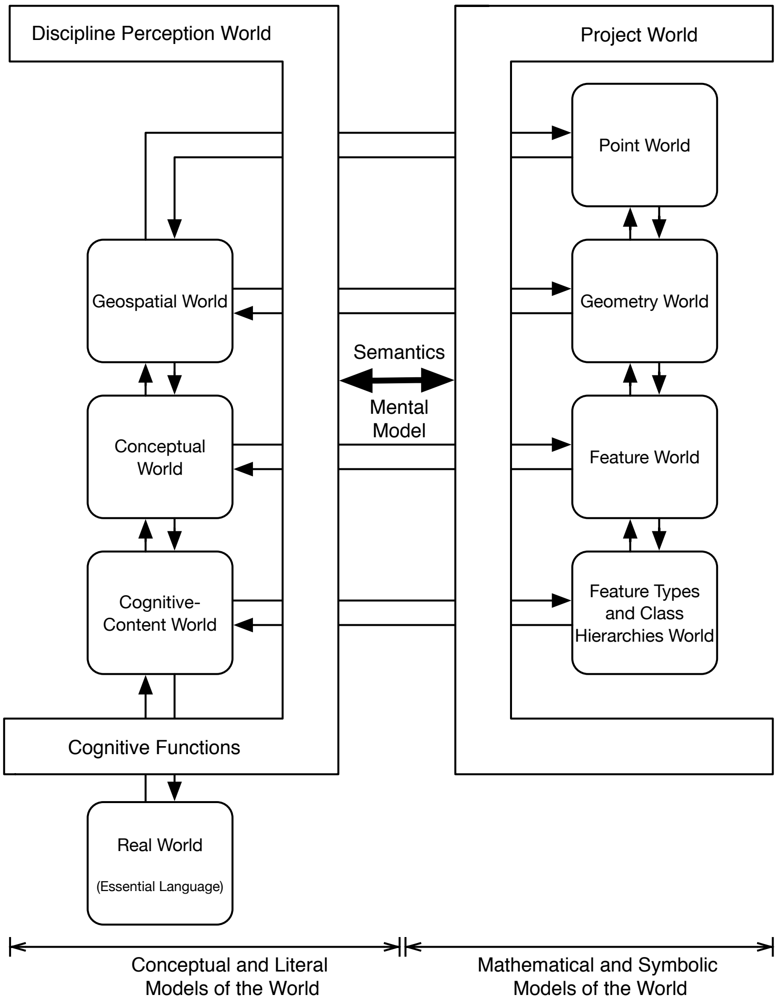


Figure 3.4: Bishr's eight levels of abstraction (Bishr 1998)

3.3 Integration and Interoperability

Fisher (2006) describes integration as the act of combining two or more subsystems to form a unified system. If the subsystems are not already interoperable, they must be made to be so before they can be integrated; thus, interoperability is a precondition of integration.

If two or more individual subsystems can be brought under the oversight of a single authority that has ultimate jurisdiction over the purpose of the subsystems and the meaning of any information within them, then the subsystems can be made to be *interoperable*. However, separate subsystems can never be fully interoperable (Fisher 2006), and therefore a fully integrated system with perfectly interoperability between subsystems is unachievable.

How then should the combination of two or more subsystems be understood? Rather than interpret them as a single system, they should instead be considered as an SoS (Fisher 2006). Thus the aim of combining subsystems should be directed towards achieving functional *interoperation* rather than perfect *integration*.

The aim of combining systems into an SoS is to harness any emergent behaviour that arises from their interoperation. Emergent behaviours are actions that cannot be localised into any single component of the system, but are instead a product of cumulative interactions (Fisher 2006). For example, information from different systems can be interoperated in a GI System where its emergent behaviour can be analysed and used for decision making.

If information between systems is not perfectly interoperable, then it must be re-interpreted as it travels from one system to the other. As a consequence, the original information is degraded and information is lost. It is presumable to say that the reliability of harnessing any emergent behaviour is dependent on the quality of any re-interpretation and minimisation of lost information.

This section will consider four system architectures with the aim of understanding how different types of systems interoperate, namely uni-directional *interchange*, bi-drectional *interchange*, and *loosely integrated* and *tightly integrated* intermediary systems.

According to Casey and Vankadara (2009), the term *interchange* is the event that occurs when data is exchanged from one system to another, whereas the term *interoperation* concerns the exchange of data and processing capability. The distinction implies that *interchange* is a uni-directional transmission between blind systems in contrast to *interoperation* being that which occurs when the systems are aware of each other's processes. Once the information passes across the interface, it is no longer recognisable by the first system. Even though the two systems may permit uni-directional exchanges in both directions, they should not be considered as being interoperable unless each system recognises the internal processes of the other (Casey and Vankadara 2009). The architecture illustrated in Figure 3.5 shows interchange of information from BIM to GIS.

Information interchange can be contrasted with the architecture in Figure 3.6 which represents a system capable of bi-directional information exchange. A fully interoperable system must experience no degradation of information on a round trip from one system to other and back again (Casey and Vankadara 2009).

Recognising that building a fully interoperable and integrated system from two independent systems is fraught with challenges, Wang, et al. (2007) considered two architectures that they refer to as *loosely integrated* and *tightly integrated* (not to be confused with *loose* and *tight coupling*). Both architectures involve the uni-directional interchange of information from two separate systems to a third system. In these cases, the third system has been designed to receive and process the information from the other two systems for a particular purpose without affecting the information in the original systems. Approaches such as these are described as achieving integration at the *Process* level (Section 3.6.2).

Wang, et al. (2007) uses the term *loose integration* to describe the architecture depicted in Figure 3.7. In this case, an intermediary system requests information from the independent BIM and GIS systems, processes the information on-the-fly, and outputs the result to the user. The intermediary system does not permanently store data for longer than is required to execute the current operation. A simple example of a loosely integrated system would be the display of feature primitives in a web browser at the visualisation level (Döllner and Hagedorn 2007). Alternatively, if the intermediary system carries out an element of spatial analysis, then integration will be said to be performed at the *Data* level (Lapierre and Cote 2007).

Wang, et al. (2007) uses the term *tight integration* to describe the architecture depicted in Figure 3.8. The intermediary system requests information from the independent BIM and GIS systems, either in a batch or following a trigger, in anticipation for a future user request. An Extract-Transform-Load (ETL) operation is performed, transforming the data in slow time into a suitable format ready to respond instantaneously to a user request. In the meantime, the information is loaded into a Spatial Data Warehouse (SDW) where it is indexed and optimised for later information retrieval and analytics (Kang and Hong 2015).

Figure 3.5: Interchange of information (via GIS)

Figure 3.6: Interoperability of information

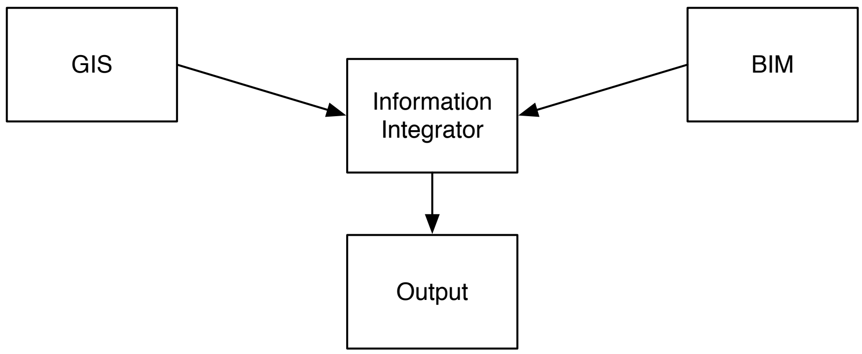


Figure 3.7: Loosely integrated Service Orientated Architecture

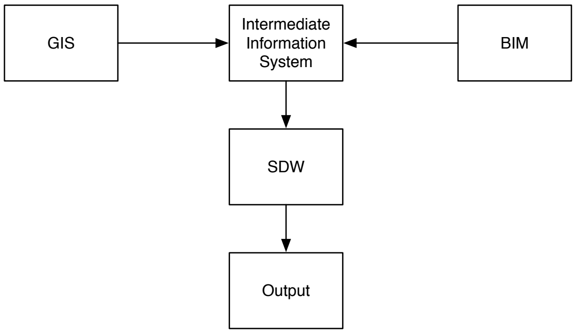


Figure 3.8: Tightly integrated Spatial Data Warehouse

3.3.1 Model Heterogeneity and Interoperability

According to Bishr (1998), the development of the model is governed by the requirements of a system. At each stage of the process, the modeller must make decisions relevant to these requirements that will inevitably compromise the precision of the model. These compromises lead to heterogeneities between models, and the nature of heterogeneity is classed as being *semantic*, *schematic* or *syntactic* depending on the stage in the modelling process at which a divergent modelling decision is made.

Semantic heterogeneity consists of two types, *cognitive heterogeneity* and *naming heterogeneity* (Bishr 1998). A modeller abstracting a particular representation from reality does so while sharing a common worldview with other modellers in the same discipline. In doing so, the modeller deems that is unnecessary to codify any shared knowledge that may exist between them.

A modeller's worldview influences how they interpret their observation of reality (*ibid.*) resulting in a bespoke representation in which only facts that are significant to the modeller are incorporated. The paradigm also influences the hierarchical classification of objects. It is for these reasons that *semantic* heterogeneity is often the principal source of interoperability (Bishr 1998).

Schematic heterogeneity relates to the hierarchical structure of information (Bishr 1998). Taking the Extensible Markup Language (XML) format as an example, the information, contained within elements, can be structured either as body text, or as an attribute, or as a sub-element. The requirements of an application will dictate the structure of information, but in some cases, decisions can be influenced by the worldview of the person carrying out the modelling. As a consequence, *schematic* heterogeneity can arise as a result of *semantic* differences (Bishr 1998).

The final form of difference is referred to as *syntactic* heterogeneity and concerns the format of data. Choosing XML instead of Javascript Object Notation (JSON) in the illustration above is an example of a decision leading to *syntactic* heterogeneity (Beck, et al. 2008). Another case in the geospatial context might relate to whether a 3D object is represented as Constructive Solid Geometry (CSG) or as Boundary Representation (B-Rep). Again, whereas the requirements of the application are likely to dictate the format, the modeller's worldview can also influence the decisions; consequently, *syntactic* heterogeneities can also find their source from underlying *semantic* understanding.

3.3.2 Levels of Conceptual Interoperability Model

Researching the nature of interoperability within the context of SoS, the Virginia Modeling Analysis & Simulation Center developed the Levels of Conceptual Interoperability Model (LCIM) as illustrated in Figure 3.9 (Tolk, et al. 2007). According to the LCIM, a hierarchy of interoperability exists on six levels ranging from *technical* interoperability at the lowest level up to *conceptual* interoperability at the top. Within the context of SoS, the broader meaning of the term *interoperability* is considered in terms of *integratability*, *interoperability* (in its narrower meaning), and *composability*.

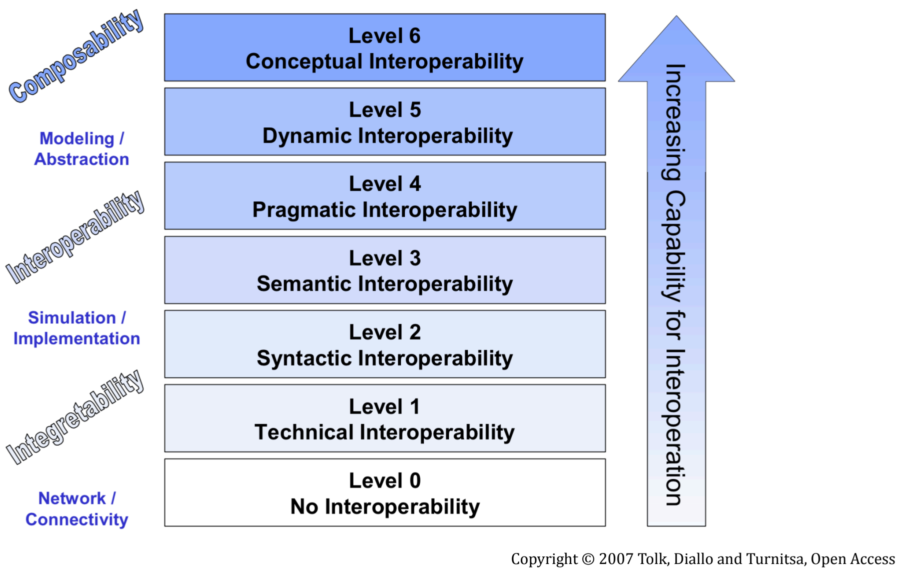


Figure 3.9: Levels of Conceptual Interoperability Model (Tolk, et al. 2007)

The first two levels are *technical* and *syntactic* and together contribute to system *integrability*. Interoperability at these levels is a technical challenge that can be resolved by adhering to openly available interoperability standards. *Technical* interoperability concerns the interchange of bits and bytes; *syntactic* interoperability concerns the structure of information (Tolk, et al. 2007).

The next two levels of the LCIM contribute to substantive or meaningful interoperability. For *semantic* interoperability, systems require access to a common reference model so as to share meaning and content (Tolk, et al. 2007); whereas, for *pragmatic* interoperability, systems are aware of the context of how information will be used in the other systems (Tolk, et al. 2007).

The top two levels of the LCIM contribute to system *composability*. *Composability* is a measure of the ability to compose an SoS using a particular system (Tolk, et al. 2007). The fifth level, *dynamic* interoperability, is used to describe systems that understand the state of other systems and whether that state changes any underlying assumptions and constraints (Tolk, et al. 2007). The topmost level, *conceptual* interoperability, is used to describe systems that share information using models that have been abstracted using the same domain of discourse (Tolk, et al. 2007). At this highest level of *conceptual* interoperability, systems can be considered as genuinely integrated as a unified system.

Furthermore, conceptual differences at this top level will effect the understanding of information at the lower levels.

It should be noted that the LCIM was developed for use in a research environment that is still striving to achieve substantive interoperability (Tolk, et al. 2007). It is fair to say that human society, as a system, itself struggles even to practise *dynamic* interoperability, let alone attain *conceptual* interoperability; indeed, fully automatic *composability* between systems may be actually impossible to achieve (Jones 2015).

To summarise, in the context of unified systems, *interoperability* is a condition for *integration*, whereas in the context of SoS, greater *integratability* leads to greater *interoperation* (Jones 2015). However, the precise definitions of the terms *integration* and *interoperability* are not always correctly followed, and the words are often misused.

Studying the LCIM is important to understand that information is exchanged at different levels of context and that the quality of the information exchange is dependent on the amount of context that is shared at each level. These various levels will be considered in alignment with the RM-ODP viewpoints and the steps of the semiotic ladder in Section 4.1.

3.4 UK BIM Mandate

Since 2016, the UK government has mandated that all government construction contracts require the contractor to deliver a building information model containing all project and asset information (HM Government 2011). This asset information will provide a foundation for Asset Management (AM) during the operational phase of the built asset (HM Government 2011).

3.4.1 UK BIM Levels of Maturity

The UK BIM Task Group (now the UK BIM Alliance) was instituted and commissioned to implement the mandate that all government procurement shall require a "*fully collaborative 3D BIM (with all project and asset information, documentation and data being electronic) as a minimum*". To describe the level of BIM capable of being incorporated into a works contract, the UK BIM Task Group adopted and advanced the *Bew-Richards* system for grading the maturity of Building Information Modelling. This system uses the illustration in Figure 3.10 ubiquitously known as the *BIM Wedge*. In this system, *Level 0* represents traditional methods of working involving no collaboration and delivery of the design product in paper form or electronic print (i.e. PDF) (McPartland 2014). *Level 1* represents the use of Common Data Environment (CDE) in compliance with BS 1192:2007 with use of 3D CAD for concept modelling but still delivering 2D output (McPartland 2014).

Level 2 of BIM maturity expects that all software must be capable of exporting to a common file format. The benefits of multi-disciplinary collaboration are realised through the use of federated models composed of 3D models natively sourced from BIM software. 2D drawings, where required, are generated from the 3D model. The UK BIM Task Group fleshed out the definition of *Level 2* BIM with a suite of documentation that includes the CIC BIM Protocol specimen contract, the UK 1192 suite of standards (being superseded by the ISO 19650 suite of standards), guidance on *Government Soft Landings*, a *Digital Plan of Works* and the *Uniclass* scheme of classification.

Level 3 indicates a grade of maturity higher than that required for *Level 2*. It envisages the use of an Object Relational Database Management System (ORDBMS) *BIM Server* in place of the file-based system expected in *Level 2*. A full scoping of what BIM *Level 3* might entail is published in the *Digital Built Britain* report (HM Government 2015). It should be noted that the *BIM Wedge* published in PAS 1192-2:2013 (BSI 2013) has been reviewed and republished in ISO 19650-1:2018 (ISO 2018) as the *Information Management Stages of Maturity* as illustrated in Figure 3.11.

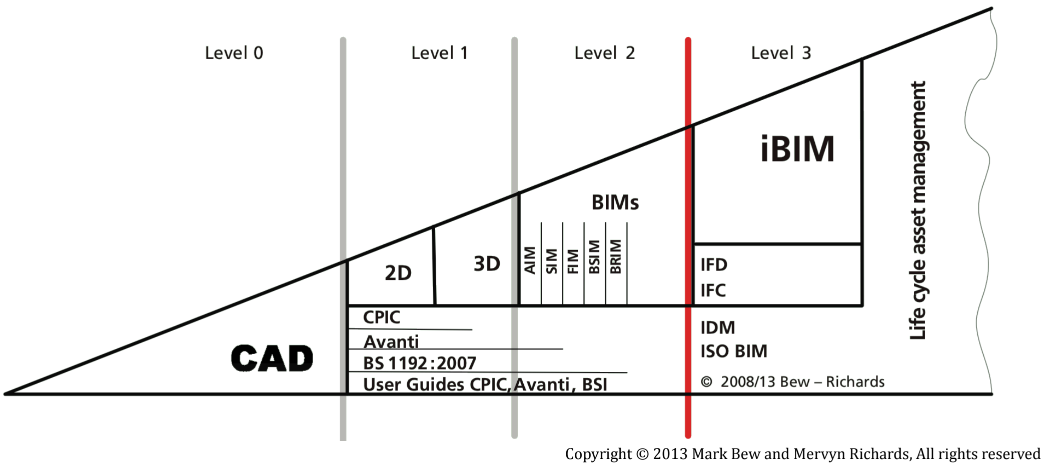


Figure 3.10: UK BIM maturity levels (BSI 2013)

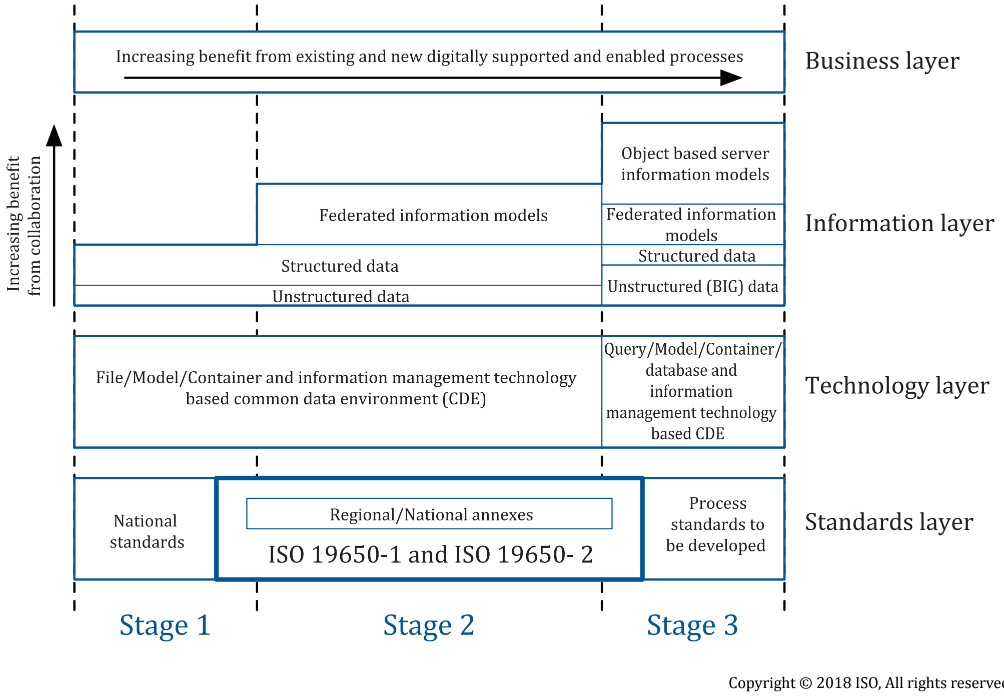


Figure 3.11: Information management stages of maturity (ISO 2018)

3.4.2 UK BIM Standards

Table 3.1: UK 1192 and ISO 19650 suite of standards

BS / ISO Number	Title of Standard	Comment	Citation
BS 1192:2007	Collaborative production of AEC information – Code of practice	Superseded by BS EN ISO 19650–1:2018, BS EN ISO 19650–2:2018	BSI (2007)
PAS 1192-2:2013	Specification for information management for the capital/delivery phase of construction projects using BIM	Superseded by BS EN ISO 19650–2:2018	BSI (2013)
PAS 1192-3:2014	Specification for information management for the operational phase of assets using BIM	Superseded by BS EN ISO 19650–3:2020	BSI (2014)
BS 1192-4:2014	Fulfilling employer's information exchange requirements using COBie – Code of practice		BSI (2018)
PAS 1192-5:2015	A specification for security-minded building information modelling, digital built environments and smart asset management	Superseded by BS EN ISO 19650–5:2020	BSI (2015)
PAS 1192-6:2018	Specification for collaborative sharing and use of structured Health and Safety information using BIM		BSI (2018)
PD 19650–0:2019	Transition guidance to BS EN ISO 19650		BSI (2019)
BS EN ISO 19650 Suite	Organization and digitization of information about buildings and civil engineering works, including BIM - Information management using BIM		
BS EN ISO 19650–1:2018	Part 1: Concepts and principles		ISO (2018)
BS EN ISO 19650–2:2018	Part 2: Delivery phase of the assets	Includes UK National Annex	ISO (2018a)
BS EN ISO 19650–3:2020	Part 3: Operational phase of the assets		ISO (2020)
BS EN ISO 19650–5:2020	Part 5: Security-minded approach to information management		ISO (2020a)

The UK 1192 suite (being replaced by the ISO 19650 suite), listed in full in Table 3.1, contains specifications on how to manage the requirements for the exchange of information requirements and including allocation of responsibilities. The original document in the suite is BS 1192:2007, the *Code of Practice for the Collaborative Production of AEC Information*, provides guidance on setting up and managing a CDE. PAS 1192-2:2013 document best practice for information management during the delivery phase of a built asset, whereas PAS 1192-3:2014 documents information management practices during the operation phase. These specifications have been (or are in the process of being) replaced by ISO 19650-2:2018 and ISO 19650-3:2020 respectively. The scope of the ISO replacements is broadly identical to that of the PAS documents, with minor variations in terminology and removal of UK specific references (BSI 2019). BS 1192-4:2014 defines expectations for the exchange of information throughout the lifecycle of a built asset, providing guidance to the AEC sector in the UK on exchanging asset information (BSI 2014a). PAS 1192-5:2015 (now superseded by ISO 19650-5:2020) provides a framework for adopting a risk-based approach to information security in a collaborative working environment. PAS 1192-6:2018 sets out the requirement for the sharing of structured health and safety information throughout the lifecycle of a built asset.

(A) Start of delivery phase
Transfer of information
from AIM to PIM

(B) Progressive development
of the design intent model
into the virtual construction
model

(C) End of delivery phase
Transfer of information
from PIM to AIM

Figure 3.12: Generic project and asset information management life cycle (ISO 2018)

The 1192/19650 suites of documents specify extensive requirements and guidance on how to manage the flow of information between the owner (a.k.a. employer, client or appointing party) and the contractors (a.k.a. suppliers, consultants, engineers or appointed parties). Among these many requirements, three areas of interest may be relevant to the questions set out in Section 2.2.

The first concerns the cyclical nature of the Project Information Model (PIM)/Asset Information Model (AIM). PAS 1192-2:2013 and PAS 1192-3:2014 and their successors ISO 19650-2:2018 and ISO 19650-3:2020, envisage the creation of a PIM during the delivery phase that contains all structured and non-structured, graphical and non-graphical information (ISO 2018). Once construction is completed, this PIM is handed over to the owners and operators of the built asset, and it is now referred to as the AIM. The AIM is maintained as a virtual representation of the material condition of the built asset. In the future, if the built asset ever undergoes a major refurbishment, the AIM is handed back to design consultants to form the basis for a new PIM. The information cycle, as illustrated in Figure 3.12, is repeated until the disposal of the built asset (ISO 2018).

ISO 19650-1:2018 states that the AIM "*supports the strategic and day-to-day asset management processes of the built asset"* (ISO 2018). PAS 1192-3:2013 defines the AIM as "*the data and information that relates to assets to a level required to support an organisation's asset management system"*. Both these statements show clear intent that it is the AIM that should be the primary repository of information for AM activities.

ISO 19650-3:2020 states that the AIM shall be a federated information model. It does not go so far as to specify the storage technology (e.g. file-based or ORDBMS) to be used, provided the technology is compatible with the workflows of the CDE. According to PAS 1192-3:2014, the AIM shall be a 3D object-based federated model managed within the CDE. From these statements, it would appear that asset owners are guided to use federated CAD-based information containers as the primary repository for AM activities.

To overcome this, ISO 19650-3:2020 provides that the AIM can incorporate existing enterprise systems provided that these are appropriately linked. Likewise, PAS 1192-3:2014 tempers its requirements by allowing information to be accessed via links to existing information systems, stipulating that any links should be implemented through two-way connectivity as illustrated in Figure 3.13. ISO 19650-3:2020 and PAS 1192-3:2014 both provide examples of enterprise systems that may be appropriate for linking with the AIM. These include computer-aided facilities management and asset management, condition monitoring, GIS and spatial analysis toolkits.

The second area of interest is two-way linking (Section 2.1.2). Linking provides synchronisation and avoids the creation of duplicated, and potentially conflicting, information between two systems. It establishes a link between a non-spatial Enterprise Asset Management System (EAMS) and the graphical CAD model, thus bestowing geometric representation upon information within EAMS. Geometric representation is beneficial for distinguishing assets from their neighbours and provides visualisation and spatial analysis of assets in the context of their surrounding environment. However, ISO 19650-3:2020 and PAS 1192-3:2014 are less clear on how to establish these links.

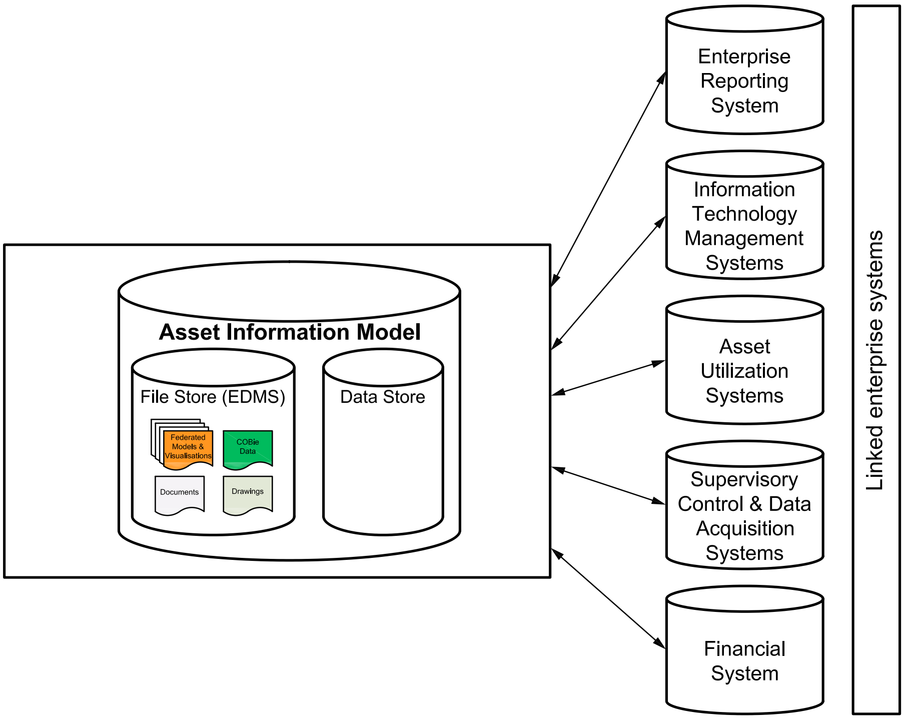


Figure 3.13: Interface between the AIM and the existing enterprise systems (BSI 2014)

The final area of interest concerning the UK 1192 suite relates to the exchange of information in compliance with BS 1192-4:2014 (BSI 2014a). This standard provides guidance on the use of Construction Operations Building Information Exchange (COBie), for the exchange of information from PIM/AIM to other enterprise information systems. COBie is a Model View Definition (MVD) capable of extracting a subset of information from an Industry Foundation Classes (IFC) model as a simple spreadsheet (East 2016). Under BS 1192-4:2014, every asset listed in a COBie exchange file must be attributed to a named space as a means of locating the asset. BS 1192-4:2014 also states that the generation of COBie should, where possible, be performed automatically. Therefore, every element in the PIM/AIM must belong to a named spatial location, if the model is to be able to support information exchange in compliance with BS 1192-4:2014. However, the assignment of a named location to every element has already been identified as an issue that impedes interoperability in Section 1.2.1.

3.5 Representation of Solid Geometry

A significant dilemma faced by modellers is how best to represent the geometry of natural phenomena, whether as *conceptual computational models* or *logical computational models* (Wise 2010). The modeller's choice is a balance between how accurately the representation must reflect the phenomenon and the computational constraints of measurement, memory and processing power (Requicha 1980; Lattuada 2006). This section will describe the principal methods for representing solid objects (and, by analogy, watertight spaces) before reviewing different approaches to advancing BIM and GIS interoperability.

3.5.1 Boundary Representation

Boundary Representation (B-Rep) is the principal method used in 3D GIS applications for representing surfaces and solids (Kothuri, et al. 2007). B-Rep describes the surface of the geometric object using nodes, edges and faces as illustrated in Figure 3.14. Because the surface is explicitly defined, the geometry is in a form instantly capable of visualisation (Requicha 1980). Because B-Rep can be used to represent any manifold surface or solid, it follows that any spatial analytical tools developed for use on B-Rep can be universally applied to all surfaces and solids. This universal application permits the overlay of spatial features, thus meeting one of the fundamental requirements of a GI System (Cowen 1988).

A set of conditions must be satisfied before a collection of polygon faces can be considered as a valid B-Rep solid (Ledoux 2013; Kothuri, et al. 2007). Failure to meet these conditions invalidates the assumptions that are pre-conditional for spatial analysis algorithms. However, B-Rep geometry is prone to corruption (Zlatanova, et al. 2020), and using an algorithm with invalid geometry will result in rejection, runtime errors, or incorrect output. In this state, B-Rep geometry is inherently unstable and requires careful management and regular repair.

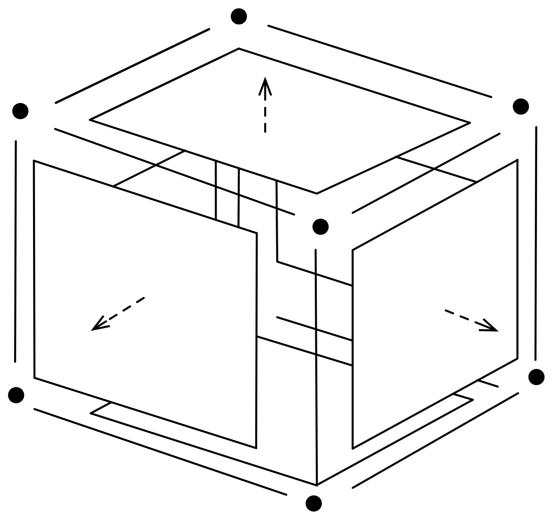


Figure 3.14: Boundary Representation (Lou 2011)

3.5.2 Constructive Solid Geometry

Constructive Solid Geometry (CSG) is a scheme for representing solid geometry through the sequential manipulation and combination of primitive instances (Requicha 1980). Construction consists of an ordered binary tree as illustrated in Figure 3.15 (Zottie 2005). Primitive instances, i.e. cuboids, spheres and cylinders may either be transformed through rigid-body motions, i.e. translation or rotation, or combined together using boolean operations, i.e. union, difference or intersection. Sweep Representations, described next, may be used in the place of primitive instances in IFC (Section 4.4.3.1).

The benefit of CSG is that it constructs an unambiguous solid product provided the primitive instances are unambiguous solids (Requicha 1980). The surface boundary of a CSG representation is implicit and must be derived prior to visualisation (Requicha 1980).

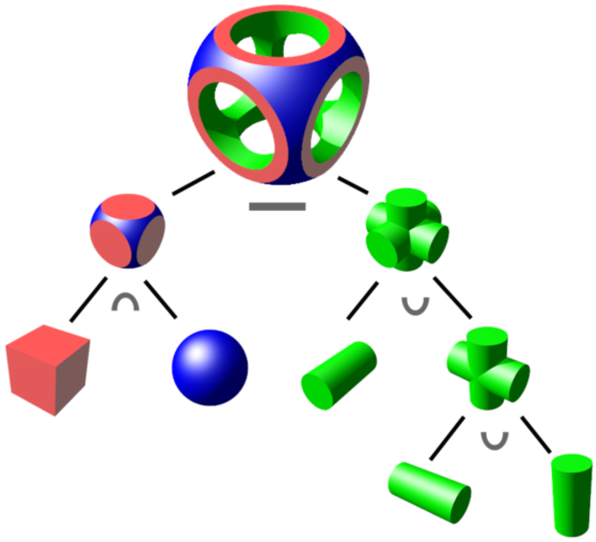
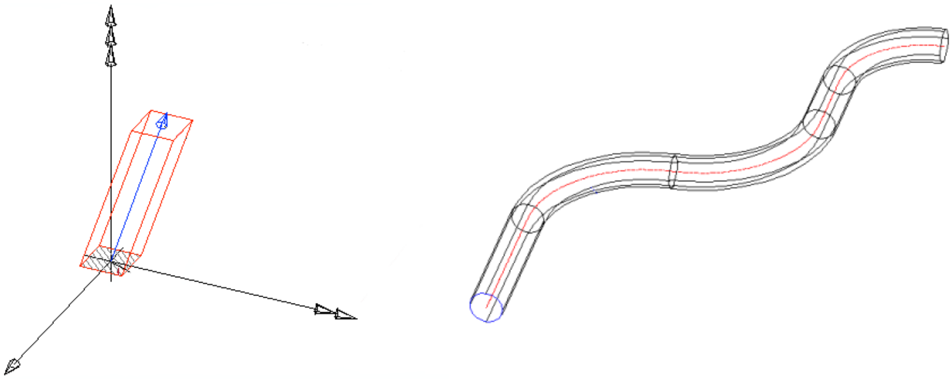


Figure 3.15: Constructive Solid Geometry (Zottie 2005)

3.5.3 Sweep Representation

A Sweep Representation, (a.k.a. Swept Volume) is the volumetric product created when a surface or a solid travels along a path. Its purest form is a perpendicular extrusion of a bounded planar surface as illustrated in Figure 3.16 by (a) the *IfcExtrudedAreaSolid* belonging to the IFC schema (buildingSMART 2007). More complex forms take the form of a circular disc swept perpendicular to a directrix as illustrated by (b) the *IfcSweptDiskSolid*.

Extrusion of a polygon derived from a two-dimensional (2D) floor plan is the most practical method for representing building elements and spaces in a CAD-based model. Like CSG, the surface boundary of a Sweep Representation is implicit and must be derived prior to visualisation (Requicha 1980).



(a) (b)

Figure 3.16: Swept Volume Representation (a) IfcExtrudedAreaSolid (b) IfcSweptDiskSolid (buildingSMART 2007)

3.5.4 Spatial Occupancy Enumeration

Spatial Occupancy Enumeration is a particular case of CSG involving a union of adjacent non-intersecting primitive cells, which may either be regular or irregular (Lattuada 2006). Two forms of spatial occupancy enumeration are regularly used: voxel arrays and octrees (Lattuada 2006). Voxel-based enumeration is realised as a 3D array of an equally spaced solid and empty cubes known as voxels as illustrated in Figure 3.17; each voxel is uniquely identifiable using a sequential index that is a function of its position in the array. The voxel array is digitally represented as a one-dimensional array of binary numbers; solid cubes being represented by a non-zero value in the binary array and cuboid voids being represented by a zero. Voxels can be classified or assigned values using arrays of integer or floating-point numbers.

The octree-based enumeration also uses cuboid voxels to represent solid matter with an alternative data structure for digital representation. Solid objects are represented as a single cube divided into eight octants (Lattuada 2006). If the voxel contained within an octant does not have the same value then the octant is recursively subdivided into eight more octants; but, if the contained voxels are identical, then no further subdivision is carried out, and the octant is designated as a leaf with a particular value. Octrees are a more efficient way of storing objects with large regions of the same value (Lattuada 2006).

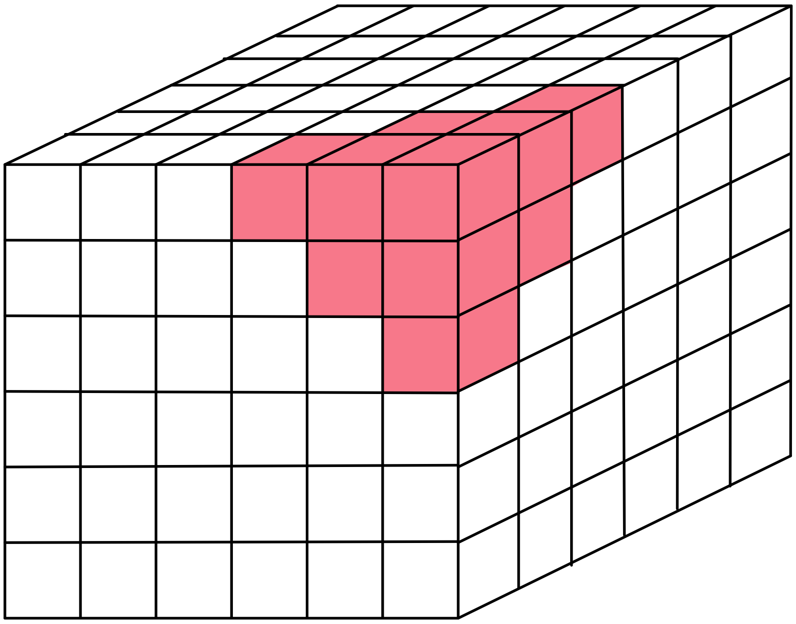


Figure 3.17: Spatial Occupancy Enumeration

3.6 BIM/GIS Interoperability and Integration

There is an abundance of literature on the topic of BIM/GIS interoperability and integration, with at least 90 articles published on the subject when counting the publications cited in the review articles published by Amirebrahimi, et al. (2015), Kang and Hong (2015), Fosu, et al. (2015), Liu, et al. (2017), Kang (2018), Zhu, et al. (2018) and Beck, et al. (2020). To make sense of the range of contributions, the authors of these review articles have grouped publications together according to the various approaches that have been followed towards integration.

Reviewing the taxonomical efforts of these authors, together with analysis of the Integrated Digital Built Environment (IDBE) working group (Gilbert, et al. 2020), it is apparent that there is a high degree of synergy between the groupings regardless of the terminology. The taxonomical groupings are laid out alongside each other in Figure 3.18 to illustrate similarities between them and an example piece of literature has been selected to represent each level in the diagram. The consolidated groupings have themselves been ordered using the headings, *Application* level, *Process* level and *Data* level - these are the headings initially used by Amirebrahimi, et al. (2015). The number of publications cited and classified by Amirebrahimi, et al. (2015), Kang and Hong (2015), Fosu, et al. (2015), Liu, et al. (2017), Kang (2018) and Zhu, et al. (2018) have been counted against each of the three levels.

An attempt has been made to incorporate some of the categorizations advanced by Beck, et al. (2020) into Figure 3.18. In their literature review into BIM/GIS integration, Beck, et al. (2020) formulated that integration approaches can be classified under three categories; although, it might be more helpful to consider these categorizations as overlapping perspectives rather than a single taxonomy. These categories are *Purpose*, subdivided into *Data Quality* and *Data Context*; *Information Characteristics* sub-divided into *Information Level*, *Real-world Objects* and *Conceptual Differences*; and *Solution Characteristics* sub-divided into *Integration Methods* and *Communication Methods*. Using these categorisations, the different types of *Integration Methods* correspond with the *Data* and *Process* levels in Figure 3.18 and the *Purpose* categorisation has potential to align with the *Application* level. However, there is no clear alignment of the *Information Characteristic* categorizations as this categorization of literature is concerned with understanding the principal differences that exist between BIM and GIS information.

3.6.1 Application Level

According to Amirebrahimi, et al. (2015), *Application* level integration involves reconfiguring or rebuilding GIS or BIM tools to include the functions of the other. The approach involves dedicated research and development to produce bespoke systems tailored to meet a particular user’s requirements. In their review Liu, et al. (2017) adopted the groupings used by Amirebrahimi, et al. (2015). The *Application* level advanced by Amirebrahimi, et al. (2015) closely correlates with the *Systems-based* integration approach identified by Kang and Hong (2015), which is described as a consolidation of other approaches into a systematic architecture.

Kang and Hong (2015) developed an SDW for facility managers that performed ETL operations on BIM information to visualise 3D assets using a GIS-based user interface.

Mignard and Nicolle (2014) developed a standalone application that enabled facility managers to query and edit asset information sourced from BIM and GIS sources.

In the groupings here, *Application* level integration is expanded to include the *Process-based* integration approach identified by Kang and Hong (2015)(not to be confused with the term *Process level* used by Amirebrahimi, et al. (2015)). This approach involves the manual use of BIM and GIS together in pursuit of a particular objective. Research following this approach will review case studies, identify best practices and publish guidelines and workflows on how to integrate information in an enterprise setting. The article published by Schaller, et al. (2017) showcases how BIM and GIS were used together in an interdisciplinary project for the preparation of environmental assessments.

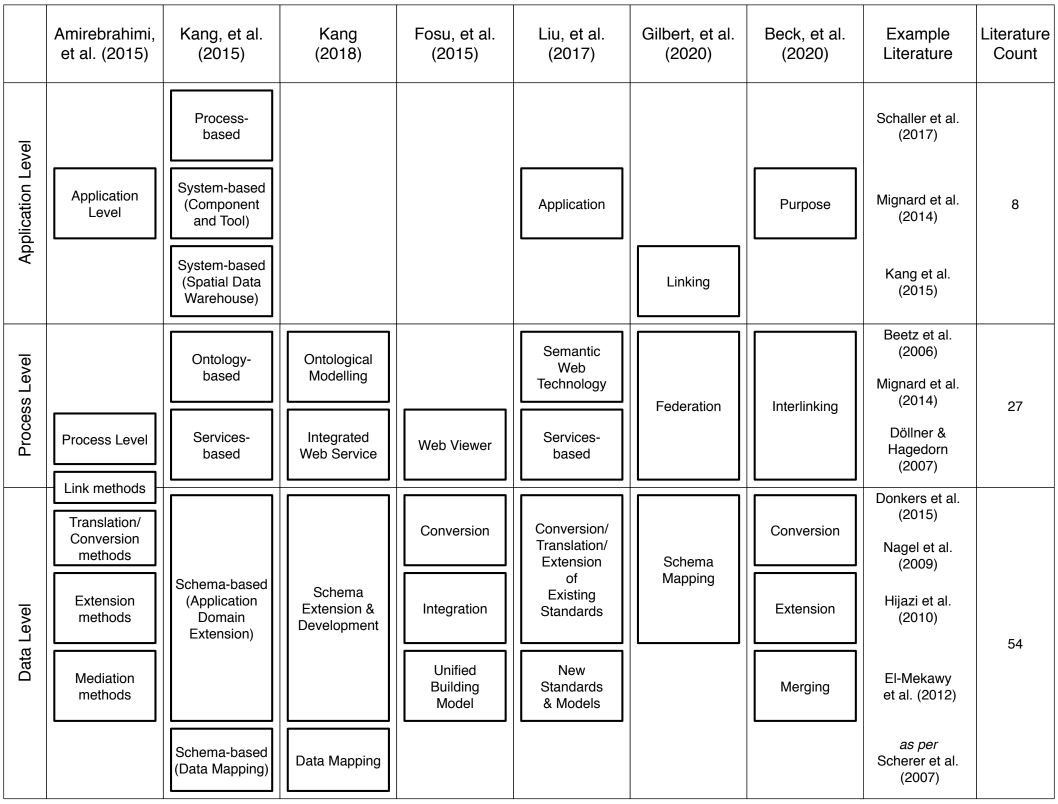


Figure 3.18: Groupings of BIM/GIS integration approaches consolidated from Amirebrahimi, et al. (2015), Kang and Hong (2015), Fosu, et al. (2015), Liu, et al. (2017), Kang (2018), Gilbert, et al. (2020) and Beck, et al. (2020)

3.6.2 Process Level

According to the definitions used by Amirebrahimi, et al. (2015), *Process* level integration permits the participation of BIM-based and GIS-based systems in tasks that require both while allowing them to remain distinct. It concerns integration approaches that do not attempt to convert information into a standardised model. Following Amirebrahimi, et al. (2015), Liu, et al. (2017) also adopted the term *Process* level dividing it into *Semantic Web-based* and *Services-based* integration, which strongly correlate with the *Ontology-based* and *Services-based* approaches proposed by Kang and Hong (2015). Although the review of Fosu, et al. (2015) mostly focusses on integration methods undertaken at the *Data* level, their *Web Viewer* method would here be classified as *Services-based* integration.

The *Services-based* group identified by Kang and Hong (2015) and Liu, et al. (2017) covers methods carried out in requesting information from data sources, with particular reference to Service Orientated Architecture (SOA). Döllner and Hagedorn (2007) reported their findings responding to the OGC Web Services Testbed Phase 4, in which they demonstrated the access of information using a Web Feature Service (WFS) and the integration of that information visually in a web browser. However, the information, in this case, is not converted into formats that can be overlaid, and thus spatial analysis cannot be performed between data sets (Cowen 1988).

The *Process* level also includes an ontological modelling approach using semantic web services. This level concerns the use of methods that request information in Resource Description Framework (RDF) form and generate evolutive ontological models that are capable of consistent adaptation (Mignard and Nicolle 2014). In a similar approach, Beetz, et al. (2006) proposed the use of a topological reasoning service to extract data in the form of RDF triples suitable for semantic web applications.

3.6.3 Data Level

The majority of research in BIM/GIS interoperability and integration has been conducted at the *Data* level. Amirebrahimi, et al. (2015) did not formulate a general definition for this level but only stated that it covered a range of methods described as *Linking*, *Translation/Conversion*, *Extension* and *Mediation* (Amirebrahimi, et al. 2016). At this level, the research focus has been on converting information into a format that has different syntax, schema or semantics. The *Data* level may concern the conversion of geometric information Zhu, et al. (2019) or it may involve the transformation of semantic information, i.e. non-geometric information which includes object hierarchies and object attributes, properties and relationships (Donkers, et al. 2015). Research has focussed on converting information into existing standards (Nagel, et al. 2009), into extended standards (Hijazi, et al. 2010) and into new standards (El-Mekawy, et al. 2012) specially created to avoid loss of information in conversion.

Kang and Hong (2015) identified an area of research not included in other reviews that they refer to as *Data Mapping*. This area relates to work done by Scherer (2007) who investigated methods - in the context of BIM alone - for extracting a subset from an IFC model as a partial model, mapping that partial model to a different format (albeit not GIS-related), and making changes to the partial model within that format. The altered partial model is then remapped back to the original format, and the changes reintegrated back into the full model. As no related literature was found in the context of BIM/GIS interoperability, there appears to be a gap in the body of research concerning full-cycle interoperability and the re-adoption of *peripatetic* information (literally information that is sent out on a journey).

3.6.4 Geometry Conversion

The direct conversion of element geometry from a BIM-based format, e.g. IFC to a GIS-based, e.g. City Geographical Mark-up Language (CityGML) format is straightforward. If an element is already represented as a B-Rep (see Section 3.5.1) then the conversion required is merely syntactic.

Established algorithms exist within BIM platform software and in ETL software such as *FME Workbench* to convert the *CSG* and *Sweep Representations* to B-Rep (Zhu, et al. 2019). *Sweep Representations* that involve a linear extrusion of a 2D polygon into B-Rep can be performed by stitching a collection of rectangular faces between two polygon faces and the result is as accurate as the input geometry (Zhu, et al. 2019). Elements with complex surfaces such as a Non-Uniform Rational B-Spline (NURBS) representation may require an algorithm such as marching cubes to aid conversion (Lewiner, et al. 2012).

However, conversion between IFC and CityGML rarely requires direct conversion of element geometry because IFC and CityGML do not share the same semantics and schematics due to conceptual differences in abstraction influenced by the modeller's worldview (Bishr 1998). As it is, the model schema for IFC was developed independently and for different purposes (Nagel, et al. 2009).

Nagel, et al. (2009) identifies that BIM-based models are created as detailed designs for the purpose of intended construction. The model is comprised of a collection of building elements such as slabs, walls, pipes and equipment. As such, the components of the model are a set of instructions for the constructor to follow.

In contrast, a GIS-based representation is more likely to be a product of surveyed observations. Without access to other information sources, the composition of the model is limited to what a surveyor can see, thus limiting the model to a collection of connecting surfaces (Nagel, et al. 2009). The exterior of the building is represented as a shell, as are the interior rooms.

A transformation from IFC to CityGML requires an aggregation of the individual building elements into a unified solid object and then classifying the surfaces of that solid as ground surfaces, wall surfaces and roof surfaces (El-Mekawy, et al. 2012a). The transformation may involve generalisation to reduce the complexity of the model (Isikdag and Zlatanova 2009).

El-Mekawy, et al. (2012a) catalogues the challenges of converting from IFC to CityGML. Isikdag and Zlatanova (2009) and Donkers, et al. (2015) proposed methods for converting IFC to CityGML at the first three levels of development. Isikdag and Zlatanova (2009) also proposed methods for converting IFC to CityGML LOD4 (Section 4.4.3.2); however, their proposals are dependent on the IFC model being well defined, i.e. requiring that it is fully complemented with *IfcSpace* elements.

The conversion of a building represented in CityGML to an IFC model cannot be performed automatically without making assumptions about the construction of the building (Nagel, et al. 2009). To overcome the challenge of making these assumptions, El-Mekawy, et al. (2012) proposed the adoption of a Unified Building Model that fully encapsulates the modelling requirements of both CityGML and IFC and supports automatic conversion to either format.

Specific tasks, such as the identification of parallel wall surfaces, in the conversion process from a surveyed model to an element-based model, may be automated. However, there is a strong potential for erroneous conversion (Kang 2018) and manual assistance in the conversion, even if only for final validation and quality control, cannot be eliminated.

3.6.5 BIM to GIS Conceptual Mapping Standard

Looking to promote better integration at the *Data* level, Kang (2018) identifies that the particular requirement to convert BIM information into GIS-based format will be different depending on the circumstances of the case in hand. Kang (2018) criticises applications or solutions that look to provide a one-size-fits-all conversion into one specific format. Conversion should not take place hidden away in a *black box*; instead, the conversion process should be exposed in a *white box* environment that can be opened up for inspection and configured to meets the requirements of the user.

Kang (2018) reports on the work in progress developing a B2GM standard to be a candidate for publication as ISO 19166 (not yet published). The standard does not seek to entrench an inflexible conversion process, but instead to provide the terminology with which to prescribe a particular BIM to GIS standard.

The mapping process is split up into three components, the *Element Mapping (EM)*, the *Level of Development Mapping (LM)* and the *Perspective Definition (PD)*. EM and LM describe how BIM elements are represented as GIS features at different Levels of Detail

(LoDs). The PD describes what information should be extracted from a BIM model and how it should be represented in the GIS, i.e. as SI units in double-precision floating-point.

In recognising that every conversion from BIM to GIS needs to be tailored according to the user's requirements, Kang (2018) identified five conceptual levels with which to view BIM to GIS integration (BG-IL) as illustrated in Figure 3.19.

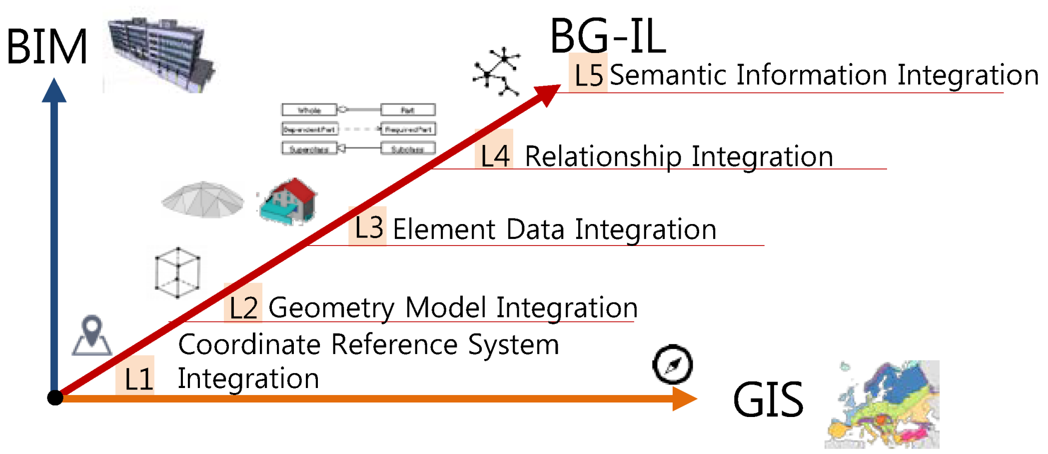


Figure 3.19: BIM to GIS Integration Levels (Kang 2018)

- IL1 - Coordinate Reference System Integration (CRSI)
- IL2 - Geometry Model Integration (GMI)
- IL3 - Element Data (Property) Integration (EDI)
- IL4 - Relationship (Topology) Integration (RI)
- IL5 - Semantic Information Integration (SIM)

Kang (2018) is careful not to define his five levels as being a measurement of model integration. For example, a conversion required for an Facilities Management (FM) application will be skewed towards attribute conversion with little emphasis on geometry conversion. Instead, the levels are intuitively matched to the practical needs of the AEC AM/FM community. The levels are supported by the results from a survey of user requirements in which users responded that they identified a higher prevalence for Coordinate Reference System (CRS) integration than for semantic integration.

3.6.6 Integrated Digital Built Environment Joint Working Group

The OGC (Section 3.2.3) and *buildingSMART* (Section 4.4.3.1) have come together to form an *IDBE* joint working group with the mandate to ensure that conceptual models by the two organisations are consistent and that data created using OGC standards (e.g. CityGML) and buildingSMART standards (e.g. IFC) can seamlessly operate (OGC and buildingSMART 2017). This working group has published an analysis of CityGML, IFC and *LandInfra*/*InfraGML* setting out the commonalities and differences and identifying the challenges to integration (Gilbert, et al. 2020).

Gilbert, et al. (2020) categorised the integration of CityGML and IFC information into three paradigms. *Schema mapping* concerns the integration at *Data* level by conversion of information into the existing schema or by extending the existing schema to receive information. *Federation* involves the momentary integration information for a particular purpose which fits in with integration at the *Process* level in Figure 3.18. The final category is *Linking* where a user can access information in another domain by linking information via Uniform Resource Identifiers (URIs); this integration at the *Application* level corresponds to the FM portal developed by Kang and Hong (2015).

3.6.7 GeoBIM

The term *GeoBIM* is a convenient moniker broadly used to describe the result of integrating geo-data and BIM-data (Noardo, et al. 2019). It was first used as the title of a scoping study set up by Delft University of Technology and Eindhoven University of Technology with the primary aim of developing an interface for reusing GIS data in the BIM domain and vice versa (Arroyo Ohori, et al. 2018). The GeoBIM project has been adopted by the European Spatial Data Research group to gain a better understanding of the needs and challenges of integrating GIS-based and BIM-based data. The first phase of the project has been to gather questionnaire responses from industry professionals (Ellul, et al. 2018) to scope out the benefits and challenges. The second phase involves two investigations: the first into the use of GeoBIM in the approval of building permits (Noardo, et al. 2019a); and the second as a tool in the domain of AM.

From reviewing the breadth of literature on BIM/GIS integration, it is apparent that the main thrust of publications focuses on integration at the Data level. Of 89 publications cited by Amirebrahimi, et al. (2015), Kang and Hong (2015), Fosu, et al. (2015), Liu, et al. (2017), Kang (2018) and Zhu, et al. (2018), 54 (60 percent) have been classified as belonging to the *Data* level (Figure 3.18). But BIM/GIS integration involves more than having interoperable data models and this gap in the literature will be explored in Chapter 4.

3.7 BIM and GIS for Infrastructure

The fundamental principle of BIM, i.e. the collaborative use of a common 3D information model as introduced in Chapter 1, can be applied to the design, construction and operation of a wide range of built assets. The range of applications can be classified into two groups, the first labelled as *Vertical BIM*, the second as *Horizontal BIM* (Costin, et al. 2018). *Vertical BIM* includes the use of BIM for buildings such as hospitals (El-Mekawy, et al. 2012), municipal buildings (Kiviniemi and Codinhoto 2014), universities (Lavy and Jawadekar 2014), residential buildings (Ciribini, et al. 2016) and opera houses (Schevers, et al. 2007). Whereas *Horizontal BIM* is used to describe the use of BIM for infrastructure assets including bridges (Davila Delgado, et al. 2017), highways (Floros, et al. 2019), railways (Kurwi, et al. 2017) and ports (Beetz, et al. 2014). Applications may span both groups being both *Horizontal* and *Vertical*; for example a water utility network is *Horizontal* in the street and *Vertical* within a building (Gilbert, et al. 2021). Although the principles are the same, the specific nature of infrastructure projects elicits its own set of requirements (Costin, et al. 2018).

Infrastructure assets require an additional library of object classes with which to describe the structure of the asset (Costin, et al. 2018). Although standard object classes are currently in the process of being agreed for each infrastructure sub-domain (e.g. road, rail, bridge, tunnel), organisations may have a requirement to develop local object classes to meet the requirements of their own Asset Data Management Manual (ADMM) (Floros, et al. 2019). In this context, Floros, et al. (2019) carried out research in conjunction with a national highway authority to develop object classes for a gantry, a retaining wall and a bridge.

The manner of how and why infrastructure projects are commissioned is very different (Costin, et al. 2018); for example, transportation projects have a propensity to be government-led and driven by a national strategic policy (Agdas and Ellis 2010). These distinctions lead to subtle cultural, managerial and contractual differences between professionals in the building sector and their colleagues in infrastructure; for example, government-led projects can hinder competitive innovation among contractors (Agdas and Ellis 2010). However, initiatives such as those directed by the UK Government show that change can happen fast when a clear government mandate is given (HM Government 2011).

The final special requirement of *Horizontal BIM* concerns the appropriate choice of CRS to describe the location of elements (Costin, et al. 2018). *Horizontal BIM* must use multiple survey stations to maintain an acceptable scale factor, i.e. the ratio of North-South distance to East-West distance projected at a particular location within a CRS. The constraint of using a Cartesian CRS, as experienced in Computer Aided Design (CAD) software, is overcome in GIS through the use of an ellipsoidal system of specifying geographic positions.

The benefits of BIM and its applications are set out in general literature on the discipline (Eastman, et al. 2011). In their review of BIM in infrastructure, Costin, et al. (2018) confirmed that these benefits extend to sub-specialisation that they refer to as *Horizontal BIM*. As well as providing more efficient collaboration between stakeholders, BIM can be used to manage risk more effectively and improve the safety record. Furthermore, it provides an interface for the latest technological advances, such as remote access to documentation via cloud computing, and visualisation of assets in virtual and augmented reality (Costin, et al. 2018).

3.7.1 BIM and GIS for Rail Infrastructure

In the review conducted by Costin, et al. (2018), the literature published on the use of BIM in transport infrastructure is dominated by bridge and highway applications; however, this interpretation is probably influenced by the principal author's doctoral research into bridges. However, their review was useful for bringing to light a selection of relevant articles on railway and tunnel infrastructure.

In the context of the use of BIM and GIS in the rail industry, Kenley, et al. (2016) carried out a case study on two rail projects in Australia and Malaysia. They concluded that the biggest challenge of interoperability lay with the lack of standards for infrastructure. In one of the case studies, an alignment-based modelling standard was found to be fundamentally incompatible with the object-based IFC format with the effect of limiting the exchange of information between formats. It is worthy to note that the work of buildingSMART on IfcAlignment (buildingSMART 2020) is soon expected to overcome these concerns (Kenley, et al. 2016).

Also in the context of BIM and GIS for rail, Borrmann, et al. (2015) selected the construction of an underground metro system line to use as a case study. In their research, the tunnelled space was represented at multiple levels of development using scale-appropriate geometry. From this, it was found that parametric design methods were essential for maintaining the consistency of multi-scale representations during dynamic planning and design cycle (Borrmann, et al. 2015).

The track and shield tunnels are just one part of the infrastructure of underground metro systems. Underground stations and other subterranean facilities are also a constituent part of this infrastructure system. The underground station ticket hall and the access shaft used as case study models in Chapter 7 belong to this particular set of underground metro system facilities. As this set of built assets have characteristics in common with buildings and with infrastructure, it is appropriate to review the literature that has been published specifically on the use of BIM and GIS in underground metro system stations.

3.7.2 BIM and GIS for Metro System Stations

BIM-based models were used for storing and visualising data from indoor environmental quality monitors in research on the Cairo metro system (Marzouk and Abdelaty 2014). Customised elements were added to models to store humidity and temperature data as element properties as illustrated in Figure 3.20. In this research, Marzouk and Abdelaty

(2014) proposed to use environmental data in the development of a multi-criteria decision-making tool to prioritise asset management maintenance schedules. Although their article demonstrated the ability to store data within the BIM-based model, Marzouk and Abdelaty (2014) did not consider linking the environmental data with the volumetric space being monitored by instrumentation.

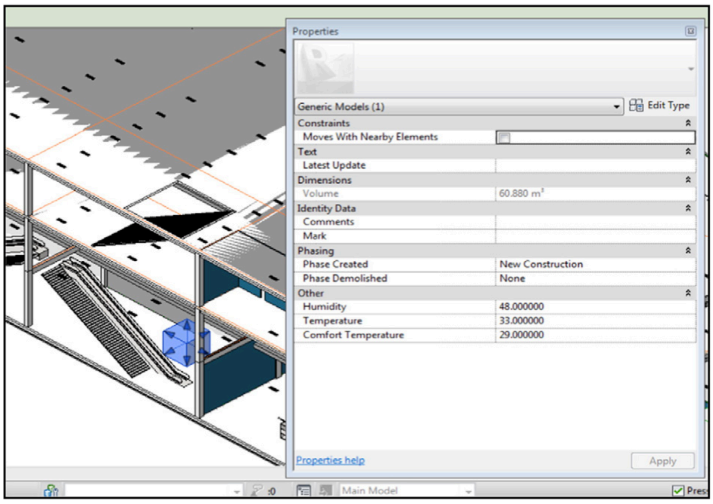


Figure 3.20: Storage of data in BIM-based model (Marzouk and Abdelaty 2014)

The use of BIM-based models was also researched for the construction of cadastral maps for enforcing property rights and responsibilities at an underground metro system railway station. Kim, et al. (2015) proposed a workflow for producing a 3D space model from terrestrial laser scans. Random Sample Consensus (RANSAC) segmentation and boundary tracing methods were employed to process the scans into polygon models which are then imported into *AutoDesk Revit* for conversion into a BIM-based model. Kim, et al. (2015) surveyed Gangnam station on the Seoul metro system and constructed a 3D model of the standard illustrated in Figure 3.21. It should be noted that the production of a high-quality 3D model first requires an extensive laser scan survey which imposes significant disruption on operations, and then incurs expense in terms of time, effort and money on post-processing and manual finishing.

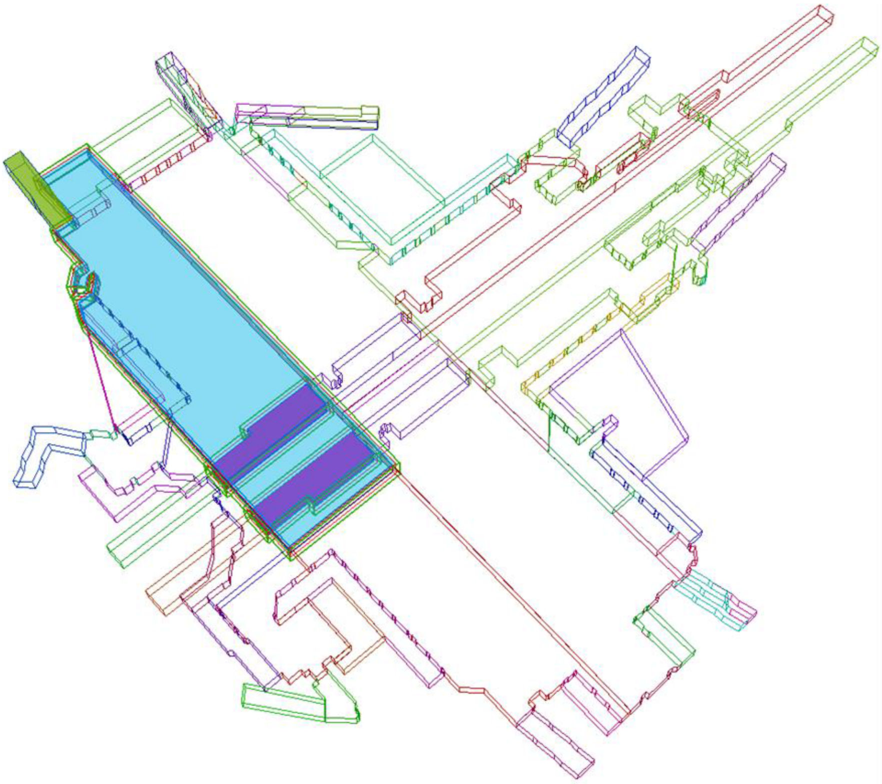


Figure 3.21: 3D parcel boundaries of Gangnam station with one parcel highlighted (Kim, et al. 2015)

Also in the context of 3D cadastre, Kitsakis and Dimopoulou (2017) used a 3D BIM-based model to highlight the challenges of complying with public law restrictions during the construction of a metro system station. In the case study, the resolution of encroachments between 2D legal spaces caused lengthy delays in receiving planning approval; however, the use of 3D legal spaces had potential to reduce the number of infringements. The use of BIM-based models in conjunction with 3D GIS was, therefore, proposed to identify the 3D relationships that exist between legal spaces to reduce the number of encroachments and thus speed up the planning approval process.

It can be seen from the literature in this section that BIM can be used to model metro system stations; however, there are still opportunities for further research. For example, although a BIM-based model was used to store temperature and humidity data (Marzouk and Abdelaty 2014), the analysis did not join that data with the geometry of the space in which it is located. The visualisation could have been enhanced by showing the volume of the space shaded with a colour scale if the space geometry had been readily available. Finally, the amount of time and effort required to carry out an indoor cadastral survey (Kim, et al. 2015) suggests that better tools need to be available to construct space geometry.

3.8 BIM and GIS for Asset Management

Asset Management (AM) provides organisations with a systematic approach for realising the total value of their assets (Section 1.1.3). Efficient bottom-up *managing of assets* is essential for effective top-down *asset management* (Zach 2017). The ground-level activities related to *managing of assets*, i.e. keeping a register of assets and planning maintenance, are the same when carried out under the aegises of AM or FM, despite having different top-level objectives (Section 1.1.4). For this reason, many observations from FM literature can be just as applicable as those learned from AM-orientated research (Section 1.1.5).

While the case for the adoption of BIM is generally accepted, it is also recognised that the adoption of BIM during the operational life of a built asset following handover is fraught with challenges and barriers (Becerik-Gerber, et al. 2012). Furthermore, there is an acknowledgement that there are almost no in-depth studies into the business value of BIM for AM/FM within an organisation (Kiviniemi and Codinhoto 2014; Kassem, et al. 2015). The literature on BIM for AM/FM can roughly be divided into three categories: the benefits and challenges of BIM in AM/FM (Becerik-Gerber, et al. 2012; Kiviniemi and Codinhoto 2014; Carbonari, et al. 2015) , the development of Employer's Information Requirements (EIRs) (Ibrahim, et al. 2016; Cavka, et al. 2017; Farghaly, et al. 2018), and the challenges of information handover (Da Luz Patacas, et al. 2014; Thabet, et al. 2016; Bayar, et al. 2016).

The overall theme that is pervasive in these articles is that organisations want to use the wealth of information contained with the design project model, however, that there are still many challenges to be overcome for BIM to be used meaningfully and consistently. Literature also strongly recommends that AM/FM practitioners be involved in the implementation of the model right from the start of a construction project (Becerik-Gerber, et al. 2012). However, these AM/FM practitioners need to know what their EIRs will be and be prepared to provide syntactic and semantic specifications on how they should be implemented.

3.8.1 Benefits and challenges of BIM for AM

BIM provides a wealth of information, but access to that information needs to improve. With this in mind, Kang and Hong (2015) proposed a system for giving facility managers access to BIM information using a GIS portal. BIM object information is extracted, transformed and loaded to a SDW for visualisation in a GI System using a star schema data structure to link the BIM model with the SDW model. The geometry of the BIM objects stored in the SDW exists in different levels of detail to support fast visualisation in the GI System. Selecting objects from the GIS portal leads to BIM information and geometry to be viewed in a BIM viewer.

There has been extensive research into the applicability of BIM for AM. Ibrahim, et al. (2016) identified six challenges that prevent the widescale adoption of BIM in the FM sector. One of these challenges is data interoperability, and although there has been a concerted effort within the industry to achieve greater technical interoperability, construction projects need to have "*well-developed practical strategies for the purposeful exchange, compatibility and integration of meaningful information*" (Ibrahim, et al. 2016). Still related to interoperability but higher up the conceptual hierarchy, another challenge arises from the difference in management requirements between the project phase and the remaining lifecycle of the building.

Ibrahim, et al. (2016) and Kiviniemi and Codinhoto (2014) both commented that for BIM to work for AM/FM, it must fit in with existing AM/FM management practices. Owners/operators will have invested capital into their information management systems, and there is corporate experience in running them. If an owner wishes to add a BIM-ready building into their portfolio, they must decide whether to manage that asset using different systems and protocols, downgrade the new building to the existing system or upgrade the rest of the portfolio to be BIM-compliant.

Farghaly, et al. (2018) identified the need for effective processes to extract, manage and integrate information to ensure interoperability. They conducted a two-part investigation; the first part researched the owner's AM requirements in the operational phase of a building's lifecycle; the second part sought to develop a practical *Revit* plug-in to extract that information.

Farghaly, et al. (2019) then followed up their earlier work by prototyping an architecture for transferring information from a BIM source to AM software. This was achieved by collecting and analysing schema such as IFC, *Revit* classes and *Uniclass* to determine their semantic structure. Interviews were then conducted with subject matter experts to establish links between classes and produce a refined ontology. Once the ontology had been verified, the data sources were then transformed into RDF and integrated with each other. The architecture could then be used to discover novel links and perform SPARQL queries. Farghaly, et al. (2019) concluded that being able to read data sets in RDF and then analyse them in conjunction with ontologies enables information in different information systems to be syntactically and semantically linked.

Lu, et al. (2019) considered the relationship of BIM with the evolving concept of the *Digital Twin*. They reach the conclusion that BIM is biased towards the design and construction phase and is not fit for the remaining lifecycle of a built asset. Lu, et al. (2019) see the *Digital Twin* as a larger framework that includes the as-built model as just one part. They propose a three-layer framework for understanding *Digital Twins*. The first *Smart Asset Layer* essentially incorporates the as-built model, the second *Smart Asset Integration Layer* provides integration and interoperability with the other information systems such as IoT and human actors. Finally, the *Smart Digital Twin-enabled Asset Management Layer* provides the strategies for maximising the potential of the *Digital Twin* when interoperating with other systems.

A megaproject is a large-scale infrastructure construction project that "*requires a large investment commitment, take many years to develop and build, involves multiple public and private stakeholders, and has a long-lasting impact on the economy, the environment, and society"* (Sergeeva, Natalya and Zanello, Chiara 2018). Megaprojects are natural adopters of innovation (Sergeeva, Natalya and Zanello, Chiara 2018) and, as such, provide excellent opportunities to carry out case studies on the implementation of innovative technology. Sergeeva, Natalya and Zanello, Chiara (2018) identified the following projects as just some of the megaprojects currently underway in the UK: Crossrail, Thames Tideway Tunnel, Bank Station Capacity Upgrade, High Speed Two, and Hinkley Point C.

A search of literature relating to these megaprojects was performed to gain an appraisal of how BIM is used for AM by considering other case studies. This search only brought to light two publications, the first being a study by Floros, et al. (2020) into the suitability of information generated in the design and construction phase of High Speed Two for use in the operational phase. Initial analysis into the export of data from *Bentley AECOsim* identified concerns relating to the mis-mapping of information interchanged via IFC (Floros, et al. 2020). The second publication is a report by Whyte, et al. (2019) commissioned by the *Centre of Digital Built Britain* analysing system interdependence using the Thames Tideway Tunnel digital twin as a case study. The report investigated the use of three analytical methods, namely *BIM Query*, *Network Analysis* and *Multi-modelling*, to investigate system interdependencies.

3.8.2 Handover

At the end of the construction phase, a built asset is delivered back to its owner. If the standard guidance (PAS 1192-3:2014) is followed, the PIM is also handed over to the owner in a new reconfiguration, now referred to as the AIM. However, if the owner is not capable of utilising the AIM as expected by PAS 1192-3:2014, a more practical solution is for the owner's Asset Information Requirements (AIRs) to be extracted, and a separate enterprise system to be populated with asset data, as envisaged by BS 1192-4:2014.

A consistent theme in the literature is that the handover of BIM information to asset and facility managers is a challenging and expensive process. Thabet, et al. (2016) published a case study documenting the transfer of a new educational facility on a university campus. They identified the poor quality of information relating to asset location and the locations served by assets as being one particular challenge. They propose a workflow supporting the handover of asset information that emphasises the importance of stipulating information requirements at the earliest opportunity.

Da Luz Patacas, et al. (2014) researched the suitability of using the COBie export function in *Autodesk Revit* for exporting the 22 AIRs specified in BS 8210:2012 *Guide to Facilities Management* (BSI 2012). They found that 10 out of 22 attributes were unsupported by IFC and 7 out of 22 attributes were unsupported by COBie. Da Luz Patacas, et al. (2014) proposed methods to overcome these gaps by using IFC property sets and writing bespoke export tools.

Lavy and Jawadekar (2014) also researched the handover of information using COBie as a data source for FM. Collecting observations from the personnel engaged in three case studies, they were able to make recommendations for future projects. Like Thabet, et al. (2016) and Da Luz Patacas, et al. (2014), Lavy and Jawadekar (2014) identified that handing over information did not meet the requirements of the FM team. In each case, the services of a consultant were necessary to recreate the required information. Their primary recommendation is that projects need to involve FM professionals to plan the information handover from the start of the design phase.

3.8.3 Business Case of Implementing BIM for AM

Munir, et al. (2018) conducted an in-depth study into how a large UK retailer derives business value through BIM-based processes. This article appears to be the only publication in the full-scale adoption of BIM for AM that considers its financial justification. Overall the case study concluded that the implementation of BIM had been a success the authors identified as being due to three key factors: development of a clear strategy prior to the adoption of BIM, connecting this strategy to business goals, and identifying information requirements.

Munir, et al. (2018) considered that too much effort has previously been placed on implementing BIM within a single platform. Single-platform implementations are inflexible and don't meet the organisation's requirements due to their one-size-fits-all approach. Instead, they identified that the multiple platform approach could be achieved so long as information between platforms is linked. By establishing a hierarchy of platforms and defining links between platforms, the flow of information within the hierarchy can be managed to support business-critical reporting and decision making.

Munir, et al. (2018) also reported that successful implementation of BIM for AM could be more easily achieved by dropping the requirement for 3D geometric models. The amount of time and resources required to implement and maintain 3D models is prohibitive, especially for existing buildings that do not have the benefit of receiving a 3D model from the designers. Munir, et al. (2018) found that 2D floor plans are sufficient for the needs of asset management for a retail organisation.

3.8.4 GIS for Asset Management

In addition to BIM, GIS is also a useful tool for supporting AM. Zhang, et al. (2009) consider the use of BIM and GIS applications for data collection and management, data analysis and visualisation in AM. Although BIM applications provide a valuable source of information, Zhang, et al. (2009) reasoned that it is the GIS applications that provide the tools that are needed to perform spatial analysis. The ability to use information in disparate systems is conditional upon their interoperability and level of integration.

Lin, et al. (2007) identified that AM systems are greatly enhanced by linking the assets with features in GIS (Section 3.4.2). However, they highlighted the importance of ensuring that information is synchronised between systems and recommend the use of automatic database triggers and well documented manual workflows to quality control the synchronisation of information.

These comments highlight the unease of asset managers concerning the dangers of maintaining important information on assets in more than one repository. It is sensible to suggest that reliable and fast synchronisation of data sources is reliant on seamless interoperability.

Halfawy (2004) reported on the use of GIS to support infrastructure AM activities within the context of municipal management. Halfawy (2004) identified that infrastructure assets are predominantly attributed with a geographic spatial location, and consequently, the functionality of AM systems is greatly enhanced by providing a data link with GIS.

Halfawy (2004) recognised that some municipal authorities used CAD to track the location of assets. Traditionally CAD has been very poor at managing semantic objects, however, there had been a move within mainstream CAD software, such as *AutoCAD* and *MicroStation*, to provide more significant use of objects stored in levels. However, GIS provides greater functionality by offering spatial analysis tools to the user in addition to capture, storage, management and visualisation. The power of GIS lies in the ability to perform a spatial query, for example, asking what assets are located within 100 metres from a particular location. Halfawy (2004) also reported that GIS provides greater functionality for accessing data stored in Relational Database Management System (RDBMS) whereas CAD information tends to be stored in standalone files.

From the literature sources in this section and the previous Section 3.4.2, it can be observed that the role of the space has significant importance. Spaces provide a means of identifying the location of assets without the need for giving precise coordinates. BS 1192-4:2014 sets a requirement for every asset exported by COBie to be provided with a named location, i.e. the space in which the asset is found. There is, however, little mention in the literature on the best practices for drawing up space information and the best way for representing spaces for AM/FM. Given that quality of location is an identified concern (Thabet, et al. 2016), this literature review will continue with a study into how spaces are identified.

3.9 Spaces

The concept of space inside buildings and infrastructure assets has been the subject of much academic discussion as *space* and *a space* are nuanced terms that have subtly different meanings. The first term *space* is generally considered to be infinite and boundless, although the use of the word with a determiner, i.e. *some space*, goes to show that it can be apportioned. However, using the word in conjunction with a definite or indefinite article, i.e. *a space*, changes its meaning to something that has determined boundaries (although there are various ways of identifying those boundaries). Indeed ISO 6707-1:2014, the standardised vocabulary for buildings and civil engineering works (ISO 2014a), defines a space as “*an area or volume bounded actually or theoretically*”.

The conceptual definition of a space creates a peculiarity if it is supposed that all objects must either be classified as physical or abstract. For although a space is intangible and has no physical substance, it still has physical dimensions, and it is *somewhere* that can be experienced.

Although the lay concept of *space* and *spaces* is considered to be self-explanatory, a formal definition is somewhat elusive (Ekholm and Fridqvist 2000) because the term has come to be understood in different ways within different professional and academic disciplines (Zlatanova, et al. 2020). The concept of *a space* in the domain of indoor navigation is slightly different from the concept used in interior design. As previously explained in Section 3.2.3, abstraction is affected by the worldview of the modeller, which means that conceptual differences can lead to semantic, schematic and syntactic heterogeneities between model formats (Bishr 1998). Zlatanova, et al. (2020) argues that a harmonised understanding of the term *space* should be used in urban applications, and that special use of the term in a particular field should be managed appropriately.

Spaces are generally the primary functional product of built assets and are usually the reason for their construction (Lee, et al. 2012). All activities are performed within a spatial envelope (Maher, et al. 1997) and it is the function of buildings to provide spaces that are environmentally controlled, secure and accessible. The management of these spaces on behalf of a business is one of the functions of facility management (Then and Akhlaghi 1992; Svensson 1998).

In essence, *a space* is a void that provides an opportunity for a certain period of time to be filled by something or someone else. The option to exercise this opportunity has commercial value, and therefore space is a valuable asset. It is fair to consider spaces as intangible assets with certain tangible characteristics.

Zlatanova, et al. (2020) carried out a comprehensive review of 147 articles that address the topic of space in urban applications; however, it should be noted that the principal author co-authored 25 of the citations in the review. Although the study has been written with a bias towards urban applications, its findings may also be applied to the infrastructure domain. Zlatanova, et al. (2020) investigated how the term *space* is used for indoor navigation, Global Navigation Satellite System (GNSS) locating services, micro-climate and thermal comfort modelling, landscape and urban planning and design, urban heat island, interior design, transportation and intelligent spaces; an intelligent space is a space that is observable by a system that can also control an aspect of that space.

From their review, Zlatanova, et al. (2020) identified that similarities and differences should be compared under four headings. The first heading *classification* mainly concerns whether a space is classified as indoor, outdoor or somewhere in between, such as a covered courtyard.

The second heading *boundary and modelling components* concerns how a space is bounded. In some disciplines such as indoor navigation, spaces are bounded by building elements such as walls and floors. In other disciplines, the boundaries of space may be more fuzzy; for example, in urban planning, a space can be bounded by a vegetation canopy. These boundaries may also be classified; for instance, in indoor navigation, space boundaries are considered as being top, side or bottom.

The heading *relevant standards* compares which standards are used by each of the disciplines. CityGML can be used for most applications, although IFC may be more prevalent in disciplines such as indoor navigation (See Section 4.4.3 for a detailed description of IFC and CityGML). Finally, the *granularity* heading concerns how space is subdivided up. A large indoor space may be split up into functional spaces, navigation spaces and object spaces where furniture is arranged.

In their review, Zlatanova, et al. (2020) make only a passing mention to the use of spaces in facility management citing the work of Kara, et al. (2018) who carried out a review of international standards for measuring the floor area of buildings.

A conceptual paradigm exists between BIM-based and existing GIS-based models (specifically CityGML 2.0) concerning how interior spaces are represented differently between the domains (El-Mekawy, et al. 2012a). This conceptual difference has the effect of preventing applications from being able to determine the name of the room in which an asset is located thus challenging interoperability (see Chapter 2).

In the BIM-based model, the spaces are not explicitly defined, but they are instead implicitly represented by the voids that exist between the building elements. The IFC schema is capable of representing these spaces as *IfcSpace* elements, but the geometry of the spaces must be constructed from the building elements (i.e. the walls and floors) either manually or using an automatic process.

This paradigm was recognised during the development of the IFC modelling standard. In proposing a conceptual model to be used as the basis for IFC, Björk (1992) identified that there is a need to reconcile the construction-viewpoint and the space-centred viewpoint in the building model schema. In the IFC schema (see Figure 4.10), construction elements and spaces are separately modelled and linked together with by the *IfcRelSpaceBoundary* relationship. The construction element and the space element are topologically related at the surface where the two elements intersect, and the option exists to store this geometry separately within the schema.

It is important to mention that CityGML is in the final stages of a major revision that will incorporate significant changes from the existing 2.0 model (Kutzner, et al. 2020). The new 3.0 conceptual model (OGC 2021) has been specifically re-written to have greater interoperability of IFC. As such, the new schema for CityGML will bring the modelling of interior spaces into alignment with the IFC approach, which will resolve the conceptual paradigm that currently exists between the two systems.

As part of this re-alignment, the OGC has made fundamental changes to the core model. Going forward, it will be possible to represent all features in CityGML using one of two base classes of object, as illustrated in Figure 3.22, namely an *Abstract Space* object or an *Abstract Space Boundary*; the former being volumetric in nature (i.e. a watertight solid), the latter being areal (i.e. a surface). The use of the term *space* should not be confused with how the word is used in the rest of this thesis, where it is used to described a void within a building (i.e. a room). In the CityGML conceptual model, the term has been aligned with how it is used in the field of robotics to include both occupied and unoccupied objects. As such, the term *space* includes the space taken up by a wall, as well as the space that is a room in a building.

Abstract Space objects are sub-classified into *Abstract Physical Spaces* and *Abstract Logical Spaces*. *Abstract Physical Spaces* are used to describe spaces that are bound or partially bound by other, objects whereas *Abstract Logical Spaces* are used to describe arbitrary spaces based on thematic considerations. This type of space might be bound by a virtual boundary to make up a part within another space, or it might be an aggregation of spaces that share a common property, i.e. ownership or administrative oversight.

Abstract Physical Spaces are further sub-classified into *AbstractOccupiedSpaces*, which are spaces that are partially or entirely filled with matter, and *AbstractUnoccupiedSpaces*, which are spaces that are entirely or mostly free of matter. The distinction is, however, somewhat subjective according to the type of object being modelled, such that a building is classed as occupied although it contains rooms that are unoccupied, whereas a stockroom is deemed to be unoccupied despite it being filled to the ceiling with boxes. A distinction is therefore made in CityGML 3.0 between unoccupied space and navigational space, the latter being derived from an unoccupied space by subtracting the enclosed occupied spaces.

Every *AbstractSpace* object can have any number of *AbstractSpaceBoundary* objects, which may either be physical or virtual boundaries. This is similar in concept to the *IfcRelSpaceBoundary* element in IFC, although this element is singular in contrast to the dual nature of the *AbstractSpaceBoundary* (i.e. both a wall and a room will each have their own *AbstractSpaceBoundary*).

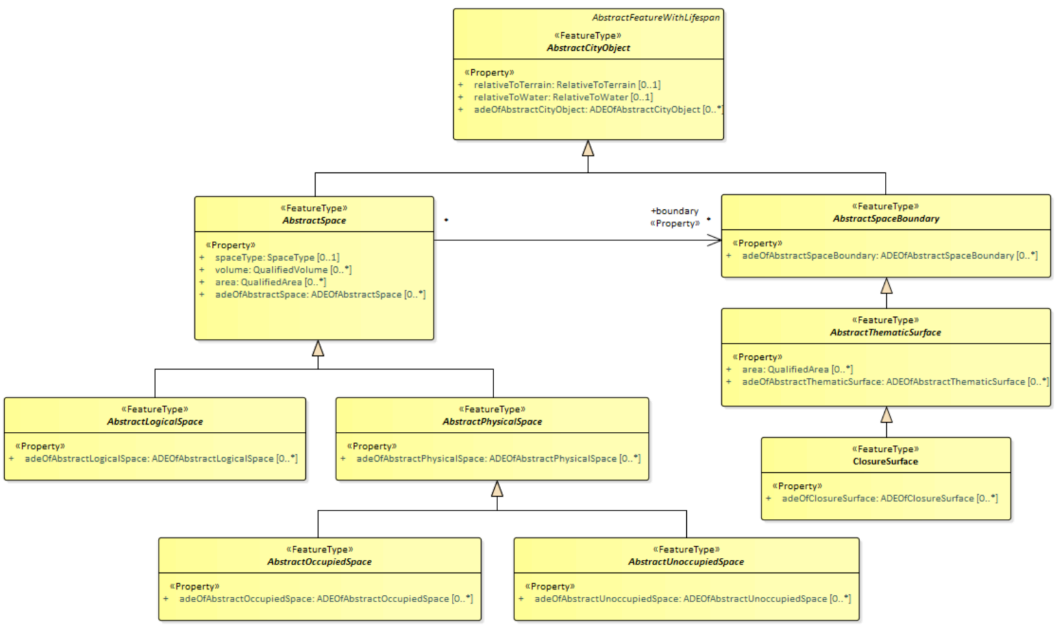


Figure 3.22: CityGML 3.0 space concepts UML diagram (OGC 2021)

The CityGML schema can be extended through the use of Application Domain Extensions (ADEs). Agugiaro, et al. (2018) considered the requirement for spaces to meet the requirements of the building energy modelling community as part of their development of the *Energy* ADE. In *Energy* ADE, a building is sub-divided into thermally homogeneous spaces that belong to a class *ThermalZone* objects, which are a subclass of CityGML *BuildingRoom*. *ThermalZone* objects are bounded by *ThermalBoundary* objects, which are a subclass of CityGML BoundarySurface_ objects. A building is also sub-divided into *UsageZones* with each *ThermalZone* contains a number of *UseageZones*.

As the *Energy* ADE is an extension of CityGML 2.0, any issues relating to IFC/CityGML interoperability are also likely to apply to the *Energy* ADE. If the conceptual model of CityGML 3.0 is more closely aligned with IFC and if the *Energy* ADE is updated to be compliant with CityGML 3.0, then the *ThermalZone* and *ThermalBoundary* objects will become more closely aligned to the IFC schema.

The distinction between *Abstract Physical Spaces* and *Abstract Logical Spaces* is not new. Maher, et al. (1997) proposed an *activity space* model formalising the representation of activities and their associated spatial envelopes. From this perspective, spaces are first defined by activity and not necessarily by their location. As such, a single room may contain multiple non-interfering activity spaces across a period of time. The model provides architects and planners with tools to encapsulate requirements and design physical spaces that are tailored to those requirements.

In an attempt to resolve the dilemma identified by Björk (1992), Ekholm and Fridqvist (2000) defined spaces according to two perspectives. From the material perspective, a space is defined as "*an aggregate of things with a materially or experientially enclosed void that may accommodate users or equipment"* as illustrated in Figure 3.23. In contrast, from a functional perspective, a space is "*the spatial extension of the process of performing an activity"* as shown in Figure 3.24.

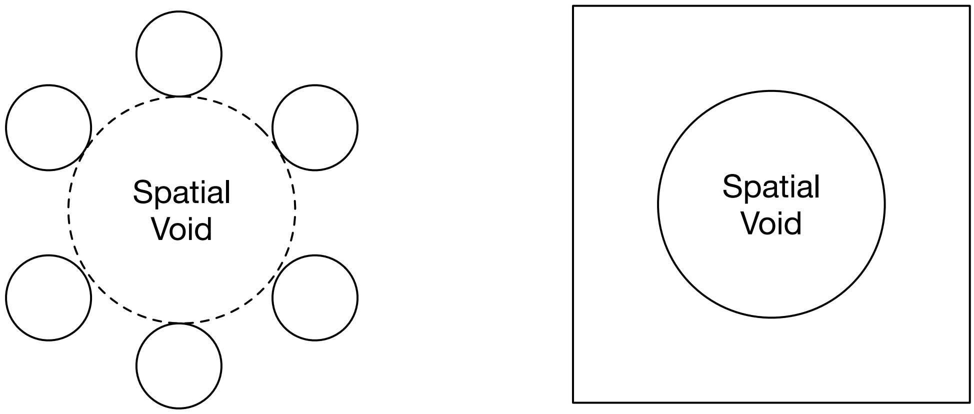


Figure 3.23: Material space (Ekholm and Fridqvist 2000)

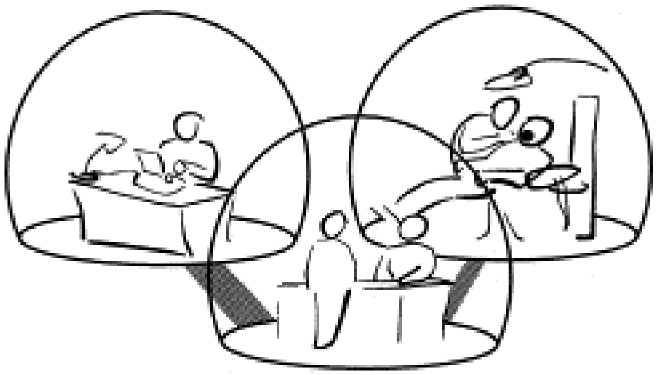


Figure 3.24: Activity space (Ekholm and Fridqvist 2000)

Ekholm and Fridqvist (2000) then attempted to address the complex relationships that exist between the material elements enclosing spaces and the functional activities that occasion a spatial envelope. They sought to do this by decomposing the components of a model into *aspectual units* which can then be aggregated depending on the desired aspect.

The definition of space as a void is problematic when the concept of a space extends to the centreline of a wall, such as the 3D equivalent of the gross floor area as defined in ISO 6707-1:2014 (ISO 2014a). With this in mind, *Autodesk Revit* provides users with the option to calculate room spaces using either the surface of the wall or the centreline. Working in the context of 3D cadastral surveying, Atazadeh, et al. (2016) highlight the importance of centreline boundaries for defining 3D cadastres and suggests changes to the IFC schema to distinguish different space geometries depending on whether the space is bounded by the inner or outer surface or the median line of the building element. This choice of position would be necessary when using the centreline of a party wall as the cadastral boundary between two terraced houses.

The ISO 6707-1:2014 definition allows for spaces that are bounded by virtual boundaries (ISO 2014a). Such boundaries might exist at an opening between two spaces that are not filled by a door or a window, or above the balustrade of a mezzanine balcony. A more technical example of virtual boundaries is described by Borrmann, et al. (2015) who use procedural modelling methods to define the clearance envelope for a train travelling in a railway tunnel. For this, the main building schemas permit the use of virtual boundaries using an *IfcRelSpaceBoundary* attribute in the case of IFC and by using a closure surface in the case of CityGML 2.0.

3.9.1 Applications Requiring Defined Spaces

The use of named spaces as a means of locating assets in accordance with BS 1192-4:2014 (BSI 2014a) has already been explained in Section 3.4.2. Further examples of the use of spaces in BIM/GIS applications are listed in Table 3.2.

Although GIS has an inherent ability to perform spatial operations (see Section 3.8), Daum and Borrmann (2014) identified that there is a lack of spatial query tools in BIM-based software. They proposed a query language for BIM and developed prototype tools and demonstrated their application on queries such as "*Does Room 107 contain any heating equipment?*". Indeed, this lack of spatial query in BIM necessitates the extraction of BIM-based elements and their subsequent transformation and loading up to a GIS platform to provide suitable tools to perform spatial queries (see Chapter 6).

Table 3.2: Applications requiring defined spaces

BIM/GIS application requiring spaces	Example
Referencing spatial location for asset management	BSI (2014a)
Performing spatial queries	Daum and Borrmann (2014)
Visualisation	Hagedorn and Döllner (2007)
Floor plan generation	Konde, et al. (2018)
Climatic monitoring	Marzouk and Abdelaty (2014a)
Building energy modelling	Bazjanac (2010)
Emergency planning	Boguslawski, et al. (2015)
Property ownership	Kim, et al. (2015)
Interior design	Zlatanova, et al. (2020)
Internal navigation	Diakité and Zlatanova (2016)
Drone operations	Li, et al. (2018)
Room planning	Schevers, et al. (2007)
Wi-Fi coverage	Lee, et al. (2018)
Indoor positioning	Kohoutek, et al. (2013)

The research in this thesis is focussed on the use of spatial queries to support the allocation of a named spaced to assets within an infrastructure model. There are many more examples of the use of spaces in the context of BIM and GIS, such as visualisation (Hagedorn and Döllner 2007), floor plan generation (Konde, et al. 2018), climatic monitorring (Marzouk and Abdelaty 2014a), building information modelling (Bazjanac 2010), emergency evacuation (Boguslawski, et al. 2015), internal navigation Diakité and Zlatanova (2016), drone operations (Li, et al. 2018), property ownership (Kim, et al. 2015), interior design (Zlatanova, et al. 2020), room planning (Schevers, et al. 2007), Wi-Fi coverage (Lee, et al. 2018) and indoor positioning (Kohoutek, et al. 2013).

3.10 Space Generation

Spaces have been identified in the previous section as being fundamental in the context of AM for several reasons. Firstly spaces provide a means by which to locate assets using a unique place name, and secondly, they provide a geometrically closed volume to be used for topological queries.

Although spaces are fundamental assets with economic value, their prescription and the finalisation of their representation tends to have a lower priority than that of construction elements such as slabs and walls. As explained earlier in Section 3.9, the explicit representation of spaces is derived from the geometry of the enclosing construction elements. However, the generation of well-defined spaces can only be completed once the design of the construction elements has been finalised.

Following a review of the use of indoor spaces in literature in the previous section, it was observed that geometrically defined spaces are not completed to the same standard as construction elements. In these examples, spaces are nearly always limited to extruded 2D floor plans. Although well suited for buildings designed to a standard shape, these tools do not work well with more involved space topology such as lecture theatres (Pang, et al. 2018), atria and stairwells (Xiong, et al. 2016). With this in mind, this section will review the methods that can be used to generate spaces from BIM-based models, paying particular attention to complex space layouts.

3.10.1 Floor Plan Extrusion

The standard BIM software applications provide tools for designers to create functional spaces semi-automatically (Eastman, et al. 2011). These spaces are generated from the polygon that is formed from the intersection of the walls with the floor slab, which is then vertically extruded to a user-defined height. Further to this, *Autodesk Revit* provides additional functionality for clipping the extruded volume with the ceiling geometry as illustrated by the space coloured magenta in Figure 3.25, instead of relying on the user to input a ceiling height (Autodesk 2014).

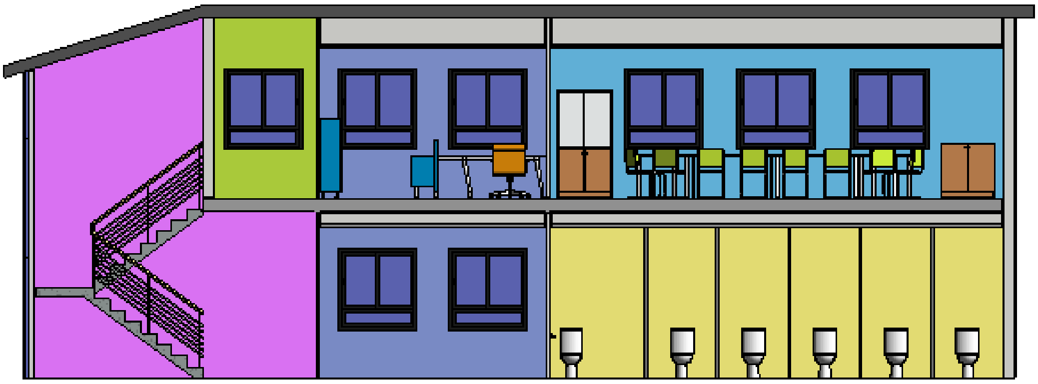


Figure 3.25: Rooms spanning floors levels in Revit (Autodesk 2014)

Boyes (2014) investigated the tools available in *Revit 2014* for generating rooms from building elements. The user selects the room tool and positions the cursor inside a room on the floor plan of the building. The tools automatically detect the boundary of the space from the enclosing elements, and the user confirms the formation of the space. If the room is not closed, the user can go back and draw a virtual boundary. Once the room is formed, the user can specify a ceiling height. The user then has the option to form rooms in 2D or 3D using the *Area & Volume Computation* option and can specify the method for defining ceiling geometry. The user also has the option to form the room using the surface of the wall or on the median centreline (Boyes 2014).

The algorithms behind the tools available in proprietary software are not open for review. The principles are, however, similar to the workflow and algorithms developed by Lewis and Séquin (1998) who demonstrated the use of a tool for generating 3D spaces from 2D architectural floor plans provide in a CAD format. The tool was used as part of a semi-automatic workflow that relied on manual assistance to provide any missing information.

The spaces generated from extruded floor polygon are limited in that they consist of vertical sides with horizontal floors. Although these tools are straightforward for practitioners to use, they are not capable of representing complicated space geometries, such as the sloping floors, staircase rises and concave tunnel sides that might be found in an underground metro system railway station as illustrated in Figure 2.1.

3.10.2 Boolean Difference

Solid modelling software, such as *OpenSCAD* (OpenSCAD 2020) that uses tools from the Computational Geometry Algorithms Library (CGAL) library (CGAL 2020), perform boolean operations on B-Rep solid objects. Using a *boolean difference* operation (Mäntylä 1988), the internal spaces can be generated by removing building element geometry from a minimum bounding volume of the building. The operation will leave behind an outer volume that surrounds the building and the individual spaces; however, a space must be totally enclosed by geometric elements for the method to be of practical benefit. Furthermore, the operation is dependent on geometry quality and the robustness of the operation (Shen, et al. 2001).

Although a literature search has not revealed the specific use of this method to form spaces from elements, boolean methods have been used to calculate navigational free space by removing furniture from a space that has already been formed spaced as illustrated in Figure 3.26 (Diakité and Zlatanova 2016).

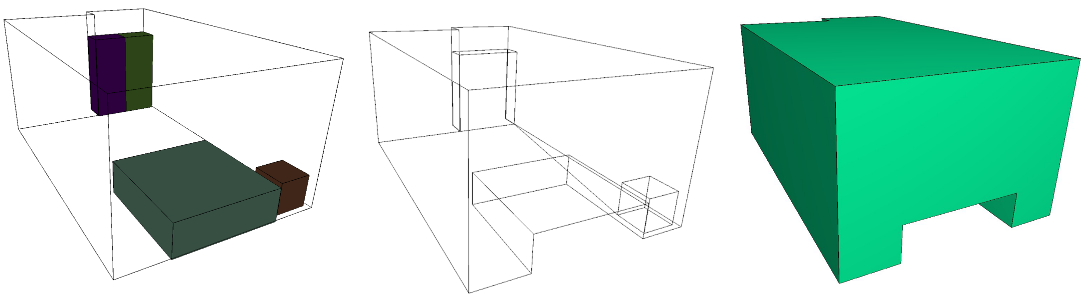


Figure 3.26: Extraction of 3D free space from building model (Diakité and Zlatanova 2016)

3.10.3 Topological Reconstruction

A method very similar to the boolean difference method uses *topological reconstruction* to generate volumetric spaces. In this method, solid construction elements are de-constructed into their primitive parts, intersecting faces are then carved up illustrated in Figure 3.27, and then a graph is constructed to represent the topological relationships between faces and spaces, similar to the one shown in Figure 3.28. Volumetric spaces can then be reconstructed the topological graph (van Treeck and Rank 2006).

Van Treeck and Rank (2006) used this method to classify space boundaries as being *Type 2a* and *Type 2b*[2] as required to perform building energy modelling. Diakité, et al. (2014) used this method to extract volumetric spaces from a geometric building model without referring to semantic information.

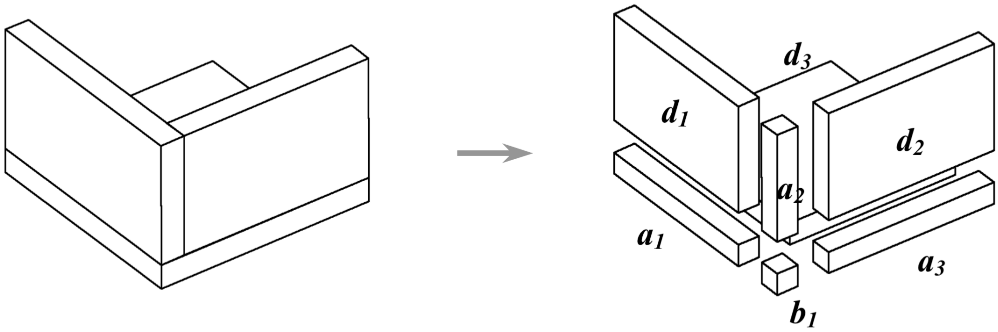


Figure 3.27: Decomposition of building elements into connection model (van Treeck and Rank 2006)

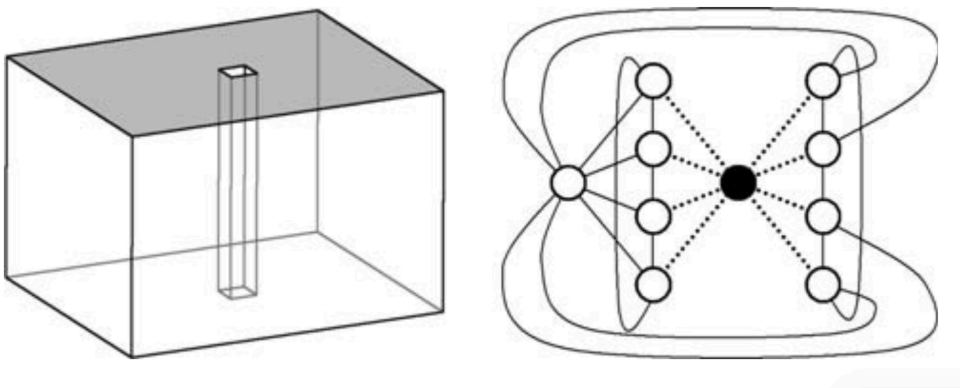


Figure 3.28: Graph of room faces of indoor air volume with enclosed pillar (van Treeck and Rank 2006)

[2] The *IfcRelSpaceBoundary* relationship between *IfcSpace* and *IfcElement* classifies every surface boundary of a space as being either *Type 2a* or *Type 2b*. A *Type 2a* boundary connects two spaces and is a thermal conduit between them, while a *Type 2b* boundary does not and is adiabatic (Bazjanac 2010).

3.10.4 Surface Pairing

An alternative method for determining the topology of volumetric spaces for the purposes of Building Energy Modelling is the *surface pairing* method (Jones, et al. 2013). Its application is, however, limited to building models in which all walls have a constant thickness and parallel surfaces. In this method, the inward-facing back surfaces of every element projected onto the back surface on the other side of the wall. The method classifies surfaces into *Type 2a* and *Type 2b* thermal space boundaries, the topology of which can then be used to reconstruct volumetric geometry representing the spaces (Jones, et al. 2013) .

Jones, et al. (2013) recognised their method could not be applied to spaces with open doorways that leave the shell boundary as incomplete and identified this as an area for further research. Indeed, none of the above methods are capable of generating a space that has unfilled openings. Openings into spaces are a common architectural feature, but they may also exist because of missing elements in an incomplete federated model or due to microscopic gaps arising from corrupt geometry.

3.10.5 Cell and Portal Graph Analysis

Spaces are used by game developers in the field of computer graphics to increase the frame rate of indoor visualisations. The time taken to visualise an indoor scene can be reduced if unnecessary scenery can be pre-culled. If a player is contained within a particular room, the Graphics Processing Unit (GPU) only needs to render the faces that belong to that room and those adjacent rooms visible through doorways.

During the development of a new game, every triangular face in the game environment is assigned to a space (a.k.a. a *cell*); the connectivity of these cells through doorways (a.k.a. *portals*) is encoded into the game as a Cell and Portal Graph (CPG). The graph table is a fast method for returning a list of faces in the vicinity of the player, which helps towards a high refresh rate, thus keeping the visualisation as realistic as possible. However, the manual construction of the CPG is a slow, laborious process that contributes to the cost of product development.

3.10.6 Watershed Segmentation

An automatic process for the construction of CPGs was published by Haumont, et al. (2003) who adapted the classic *watershed transform* algorithm to exploit the graphical capabilities of the GPU to create planar portals between cells. They used an adaptive octree-based distance field transform to simulate the *flooding up* of the distance field following the immersion analogy used in the watershed algorithm. When the rising flood-waters from two spaces meet, Haumont, et al. (2003) used the z-distance values in the GPU to calculate the position at which to create a planar portal between the spaces.

Haumont, et al. (2003) demonstrated that the process could be applied to real-life architectural scenes and medical tomographic images. Although Beucher and Meyer (1993) had recognised that the watershed algorithm could be used to segment 3D images, the work of Haumont, et al. (2003) appears to be the first use of the algorithm in the architectural domain. The work of Haumont, et al. (2003) is mostly cited in the field of computer gaming, although Koopman (2016) cites the principle as part of his research into indoor navigation.

3.10.7 Summary of Space Generation Methods

The benefits of each approach for creating spaces are summarised in Table 3.3, alongside their dependencies and the challenges that they present. *Floor plan extrusion* is the simplest method to implement and is widely used to create spaces in proprietary software such as *Autodesk Revit*, however, it is dependent of the provision of 2D floor plans as closed 2D polygons. Furthermore, the representation struggles to represent complex spaces such as those that might be found in an underground metro system station, as illustrated in Figure 2.1.

The next three methods, *boolean difference*, *topological reconstruction* and *surface pairing*, are all reliant on the provision of building elements that form closed watertight spaces. However, there are many reasons, intentional and otherwise, why watertight spaces may not be available. It may be that spaces have been designed with open portals between them, or there may be tiny openings between spaces that have not yet been plugged by the designers, such as conduits. There may also be technical reasons, for example if the closing elements are contained in a file that is not be accessible. Lastly, the geometry may become corrupted resulting in minuscule slivers between building elements. As such, these three methods may not always be suitable for use in the context of a complex infrastructure project such as Crossrail.

The *watershed segmentation* approach, using either the classical watershed transform or the GPU-assisted method, appears to be a suitable candidate for segmenting open space and creating closed watertight spaces that can be used for locating assets in the context of infrastructure asset management. The next section will review the literature published in connection with the *watershed transform*.

Table 3.3: Summary of space generation methods

Method	Dependencies	Benefits	Challenges
Extruded Floor Plan	Closed 2D Spaces Elevation of Floor Height of Space	Simple algorithm easy to implement Concise representation	Floor plan dependent on project timeline Inadequate representation
Boolean Difference	Bounding elements must enclose space Bounding elements must be valid solids	High fidelity representation (i.e. as good as bounding elements)	Boolean operations unreliable
Topological Reconstruction	Bounding elements must enclose space	High fidelity representation (i.e. as good as bounding elements) Less susceptible to boolean error Topology fully representated	
Surface Pairing	Bounding elements must enclose space	Same as topological reconstruciton Meets requirements for energy modelling	
Watershed Transformation		Bounding elements can be invalid Detects portals between spaces Voxels guarantees solid geometry	Difficult to detect portals with tunnels Over-segmentation Voxels require conversion to B-Rep
GPU assisted Watershed		Bounding elements can be invalid Detects portals between spaces GPU speeds up watershed calculations	Difficult to detect portals with tunnels Over-segmentation Topological reconstruction method still required for B-Rep

3.11 The Watershed Transform

The *watershed transform* is a tool used in the field of mathematical morphology. The transform was fundamentally developed to calculate the catchment basins in a topographic terrain. By applying the watershed transform to a rasterised Digital Elevation Model (DEM) image, every pixel can be assigned to a catchment basin (Soille and Ansoult 1990).

Although the tool has obvious uses in the field of hydrology, it can be used for other applications such as object detection in grey-scale 2D images. Using watershed analysis, Meyer and Beucher (1990) used a watershed transform for processing images from a vehicle-mounted camera to determine the trajectory of the road ahead.

By filtering a photographic image, Beucher and Meyer (1993) demonstrated that the *watershed transform* algorithm could be used to count the number of oval-shaped grains in an image. Figure 3.29a contains beans of coffee that are not distinctly bounded but instead overlap each other. The grey-scale picture is converted to a boolean image, and a Euclidean distance field transform is applied (Figure 3.29b). Minima in the inverted distance field are identified (Figure 3.29c) and the watershed field transform applied demarcating individual beans in (Figure 3.29d.

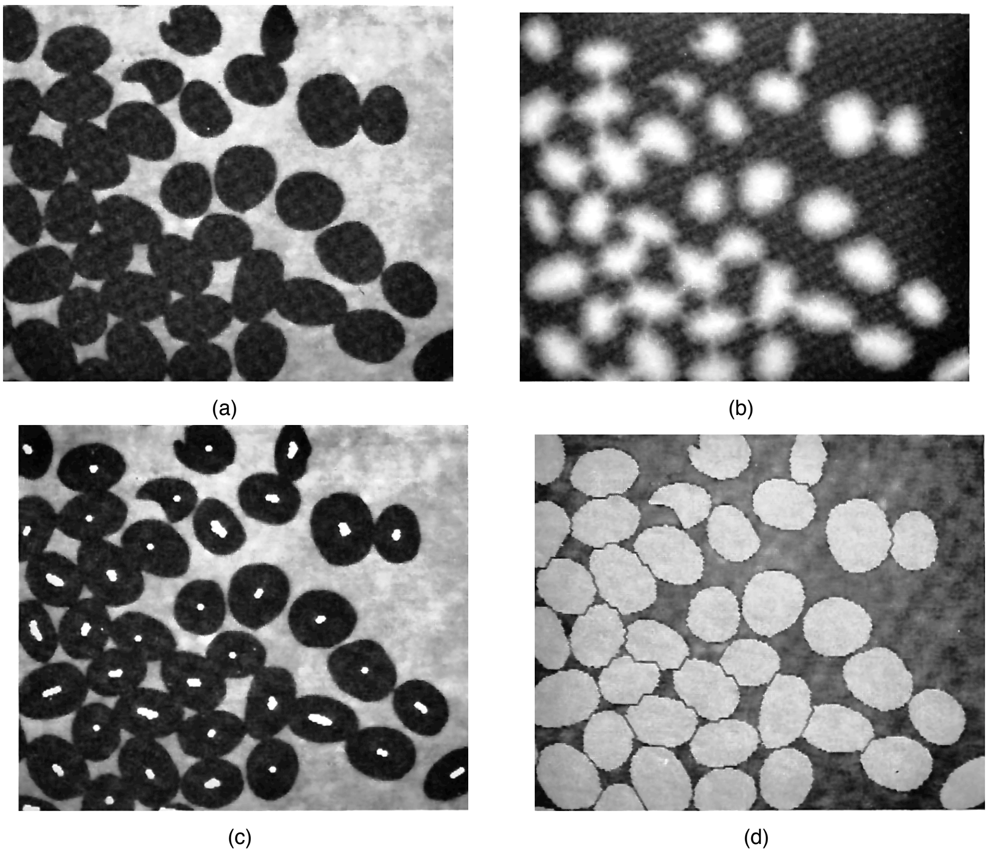


Figure 3.29: Watershed segmentation of coffee beans (Beucher and Meyer 1993)

The transform is easily extended in principle to 3D, although handling data at higher dimensions is contingent on the available processing power and memory. In this way, the *watershed transform* is widely used by radiologists in the field of medical imaging to segment 3D images captured with MRI and CT scanning equipment (Sijbers, et al. 1997).

Practical implementation of the *watershed transform* algorithm can be broken down into a four-part strategy; *marker selection*, *image pre-transformation*, *watershed transformation*, and *post-processing*. (Beucher and Meyer 1993). Choosing the most appropriate methods for *marker selection*, *image pre-transformation* and *post-processing* is highly subjective and dependent on the image type and the desired features for segmentation, indeed these steps may require an element of manual input and calibration. In contrast, the execution of the *watershed transform* itself is mechanical in nature and requires limited oversight (Beucher and Meyer 1993).

Numerous publications provide a comprehensive review of the various conceptual foundations and the implemented algorithms (Roerdink and Meijster 2000; Romero-Zaliz and Reinoso-Gordo 2017; Kornilov and Safonov 2018). There are, in essence, two different approaches to computing a solution; the *rainfall* approach considers the flow of water downhill within the topography, while the second *immersion* approach considers how the topography will flood up (Kornilov and Safonov 2018). Within each of these two concepts, there are different ways of calculating solutions. Soille and Ansoult (1990) used homotopic thinning and pruning in sequential iteration until reaching an idempotent result, whereas Meyer (1994) used a shortest-path algorithm on a graph weighted using pixel value gradient.

3.11.1 Flooding by Immersion Analogy

The *immersion* approach, as used by Soille and Vincent (1990), is implemented for use in *Python* as the *SciKit-Image watershed* function[3] (scikit-image 2020). It is this function that will be used to segment open spaces in Chapter 8. In addition, the concept of flooding by immersion is used by the GPU-assisted adaptation published by Haumont, et al. (2003).

[3] The *SciKit-Image* code (scikit-image 2020) specifically cites Soille and Ansoult (1990) but this would appear to be an mistaken reference. On closer reading, it would appear that the code should instead refer to Soille and Vincent (1990).

In the *immersion* analogy, the terrain is porous at the lowest point, or minima, of each catchment area (Soille and Vincent 1990). Below the surface of the terrain there is a water table that is slowly rising. As the rising water table reaches the minima points, the waters enter the catchment area and start to flood the terrain as illustrated in Figure 3.30. The waters in a catchment area are sourced from a single minima, and these sources define the identity of the catchment area.

As the water table continues to rise, the basins are kept separate by ridges in the terrain until the waters meet at the head of the valley, also known as the saddle. As they meet, the waters from separate sources do not mix at the saddle; as per the illustration in Figure 3.30, it is as if an artificial dam has been built (Wegner, et al. 1998). There may be situations where the saddle is not a sharp ridge, but is instead a flat plateau. In these cases, the artificial dam is deemed to exist along a line that is equidistant between the catchment areas on either side of the plateau. Literature refers to this artificial dam as the Geodesic Skeleton of Influence Zone (SKIZ) (Soille and Vincent 1990; Wegner, et al. 1998; Haumont, et al. 2003).

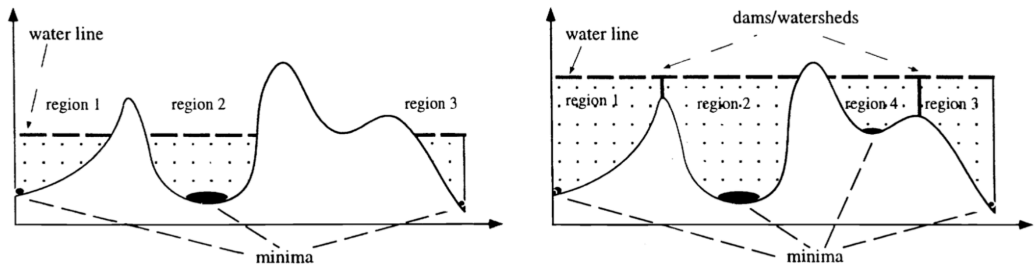


Figure 3.30: Flooding of catchment basins (Wegner, et al. 1998)

Eventually the entire terrain will be immersed, and the only objects that can be seen on the water surface are the dams that lie on the catchment area watersheds. Although the immersion concept is based on a 2D analogy, the watershed algorithm can easily be extended to a 3D array; instead of producing watershed lines and catchment areas, the algorithm creates surface boundaries and space volumes (Haumont, et al. 2003).

3.11.2 Space Segmentation using Distance Field

When a 2D boolean image of overlapping objects, such as the coffee beans in Figure 3.29, is pre-filtered using a distance field transformation, the *watershed transform* can be used to estimate the missing boundaries and segment the image accordingly (Beucher and Meyer 1993). Haumont, et al. (2003) extended this application of the algorithm to "*plug up"* the portals between spaces and segment the internal space into manageable cells.

By convention, the *watershed transform* algorithm is written to start with the lowest values in an image field and work up, in keeping with the analogy of flooding by immersion. In order to comply with this convention, a distance field must first be made negative, so that the centres of the catchment areas, i.e. the furthest points from the boundaries, are the minima (Soille and Vincent 1990; Beucher and Meyer 1993).

3.11.3 Marker Selection

In its simplest implementation, the *watershed transform* algorithm floods up the catchment areas from sources at every minima point in the image field (Soille and Vincent 1990). The locations of these sources, are used as markers to seed the algorithm (Beucher and Meyer 1993) and to generate identifiable zones of influence. The location of minima can be calculated by applying a gradient function to an image, however, over-reliance on this method will over-segment a noisy image with local anomalies (Beucher and Meyer 1993). By carefully selecting appropriate markers, the quality of the output can be improved.

3.11.4 FIFO Queue Implementation

Beucher and Meyer (1993) published a practical algorithm for segmenting an array of discrete values. A First-In, First-Out (FIFO) queue is established for each unique value in the distance field array. The algorithm requires a set of marker cells with which to seed the procedure. These marker cells are then added to the FIFO silos according to their value.

The first cell is pulled from the queue containing the lowest values. If the corresponding cell in the output array is already populated, the algorithm moves onto to the next cell. Otherwise, the cell in the output array is assigned the unique identity of the marker with which it is associated. The neighbouring cells in the input array adjacent to this cell are then dropped into the relevant queue depending on their value. As each neighbouring cell is added to a queue, it is tagged with the marker of the cell to which it is adjacent. The next cell is pulled from the lowest value queue for consideration, and this process is repeated until every voxel in every queue has been considered at which point the output array will have been populated.

3.11.5 Heap Queue Implementation

Instead of establishing a FIFO queue for each value in the voxelated distance field, the watershed algorithm as implemented by the *SciKit-Image* package (scikit-image 2020) uses of a heap queue. A heap queue works by sorting items so that the item with the lowest value is always the next item to be pulled from the queue. The rest of the items in the queue are not sorted in exact order but are instead stored in such a way that the next lowest value will *gravitate* to the top of the heap. Using a heap queue does not waste computational time maintaining the precise order of the queue along its entire length.

In the *SciKit-Image* package (scikit-image 2020), the value of each voxel is ranked using a tuple-pair consisting of distance field value and an heap queue counter. When sorting two voxels with the same value, the voxel that has been in the queue longer will gravitate to the front of the queue. Using this counter enables the algorithm to allocate cells of identical value that lie on opposite sides of the SKIZ.

3.12 Summary

In this chapter, a review of literature concerning various different fields has been conducted, including information systems theory, the use of BIM and GIS in infrastructure and the use of spaces in asset management. From this review, it is possible to summarise a few observations.

From the literature concerning information systems and their interoperability and integration, it is evident that information systems are a socio-technological phenomena that span a range of levels from technical to conceptual. These levels will be explored in Chapter 4 and compared with the various research approaches that have been published on interoperability and integration of BIM and GIS systems.

The review of BIM and GIS in infrastructure has shown that the two technologies can be used together to combine their strengths throughout the life-cycle of a built asset from the delivery phase to the operational phase. Among their many applications, The literature shows that BIM and GIS have a role to play in the management of assets to support both AM and FM. Within this role, the name of a particular space is commonly used to identify the location of an indoor asset, however the concept what a space differs depending on the application it is needed for. Furthermore, these spaces are rarely described as explicit 3D geometrical representations.

A variety of methods for creating explicitly represented 3D spaces have been described in this chapter with various strengths and weakness. The suitability of using these methods to create spaces in the context of the case study described in Section 1.2.1 will be investigated in Chapter 8 and Chapter 9.

4 Developing a Spatial Information System Framework

This chapter will set out to investigate the hierarchical socio-technical structures that can be found in Building Information Modelling (BIM) and Geographic Information Systems (GIS) and compare them alongside existing frameworks with the aim of proposing a novel Spatial Information System Framework. This framework will then be used in Chapter 5 to analyse the Technical Information systems at Crossrail and identify the level of interoperability and the challenges that hinder full integration.

From the literature published on BIM/GIS interoperability, as described in Section 3.6, it is apparent that there are many different approaches for achieving better interoperability between the two modelling systems. Indeed, the authors of seven publications have grouped the literature on BIM/GIS interoperability into various groupings as summarised in Figure 3.18. It is apparent from these groupings that the level of interoperability differs within each group; for example, they range from the simple visualisation of features together in a web viewer (Döllner and Hagedorn 2007) to using information together to make decisions in a project (Schaller, et al. 2017). From this grouping, it can be observed that the various forms of integration achieved in the literature correspond to the levels of conceptual interoperability in the Levels of Conceptual Interoperability Model (LCIM) (El Mekawy, et al. 2008).

In Chapter 1, it was remarked that the acronyms BIM and GIS are often used colloquially and in a different sense to their formal definitions. To confuse matters further, the standardised definitions of BIM and GIS vary, ever so slightly, between authoritative sources. A comprehensive investigation into BIM/GIS interoperability is muddled by this everyday use of the terms BIM and GIS in conjunction with an imprecise understanding of what interoperability and integration seek to achieve. Indeed, Hijazi and Donaubauer

(2017) considered it necessary to explain that while the term *CAD/GIS Integration* is more appropriate at a systems level and that *BIM/UIM Integration* (Urban Information Modelling) is more appropriate at a procedural level, they would nevertheless use the more colloquial *BIM/GIS Integration* in the text of their article. This confusion was also commented on by Beck, et al. (2020) in their paper.

To address this issue, it is proposed to take a closer look at the formal definitions of BIM and GIS in the context of their socio-technical structure to understand how the various concepts relate to each other. This chapter will use the viewpoints used by the Reference Model - Open Distributed Processing (RM-ODP) framework (Section 3.2.2) and the steps on the semiotic ladder used in the Data, Information and Human Activity Systems (DIHAS) framework (Section 3.2.1) to develop and propose a hierarchical framework to describe the make-up of a generic spatial information system. The proposed system will provide a common framework with which to explore the socio-technical nature of both GIS and BIM and then analyse the levels of interoperability between the two systems. The common framework will then be used to analyse interoperability between BIM and GIS and to gain a better understanding of what integration entails.

4.1 Existing Information System Frameworks

In order to better understand the semantic relationship of the terms used in BIM and GIS, it may be prudent to refrain from referring to existing definitions and instead look at the structure of the generic information system frameworks reviewed in Section 3.2 and Section 3.3.

The first to be reviewed is the Levels of Conceptual Interoperability Model (LCIM) described in Section 3.3.2. This model is used to describe levels of interoperability between System-of-Systems (SoS) and identifies six levels, starting at the *technical* level and leading up to the *conceptual* level.

The second of these is the Reference Model - Open Distributed Processing (RM-ODP) framework (ISO 1998) described in Section 3.2.2. This framework is influential in developing the standards for Spatial Data Infrastructure (SDI) as published by Comité Européen de Normalisation (European Committee for Standardization) (CEN) (CEN 2012,a,b, 2013). The RM-ODP has five viewpoints - *Engineering*, *Technology*, *Computational*, *Information* and *Enterprise*. According to ISO 19101-1:1998, these viewpoints do not form a fixed sequence of layers and as such are not dependent on each other. The viewpoints are deliberately non-hierarchical so that the architecture of a system can be described from one viewpoint without having to refer to a different viewpoint. That said, the viewpoints are nearly always published in the same order, which tends to suggest an underlying quasi-hierarchical structure.

The third framework in Section 3.2 is the semiotic ladder used by Beynon-Davies (2010) and referred to in this thesis as the DIHAS framework (Section 3.2.1). In contrast to the RM-ODP, this framework does have a hierarchical structure. The *Data System* concerns the empirical representation of information in a physical form, while the *Information System* concerns the use of semantic and syntactic modelling of information. These *Data* and *Information Systems* are subservient to the *Activity System*, which involves the pragmatic use of information to support higher functions.

As part of this work, the viewpoints in the RM-ODP have been mapped onto the different steps of the semiotic ladder. The *Engineering*, *Technological* and *Computational* viewpoints can be mapped with empirical modelling, while the *Information* viewpoint corresponds with syntactics and semantics, and the *Enterprise* viewpoint is concerned with pragmatics. If these correlations are tabulated together in Figure 4.1, then an underlying hierarchical pattern can be observed.

There are two other frameworks developed in the BIM domain that are worthy of consideration in developing a hierarchical framework of perspectives. The international firm of engineering consultants, WSP Parsons Brinckerhoff, published an article on their website explaining the broader context of BIM in commercial practice (Kennerley 2013). The article included a graphical framework (Figure 4.2) that is regularly referred to in trade publications. As such, it is worthy to include it in this comparative analysis.

The graphical framework consists of a series of concentric circles, with *3D CAD* in the centre expanding out to *Institutional and Cultural Framework* in the outer ring. However, a

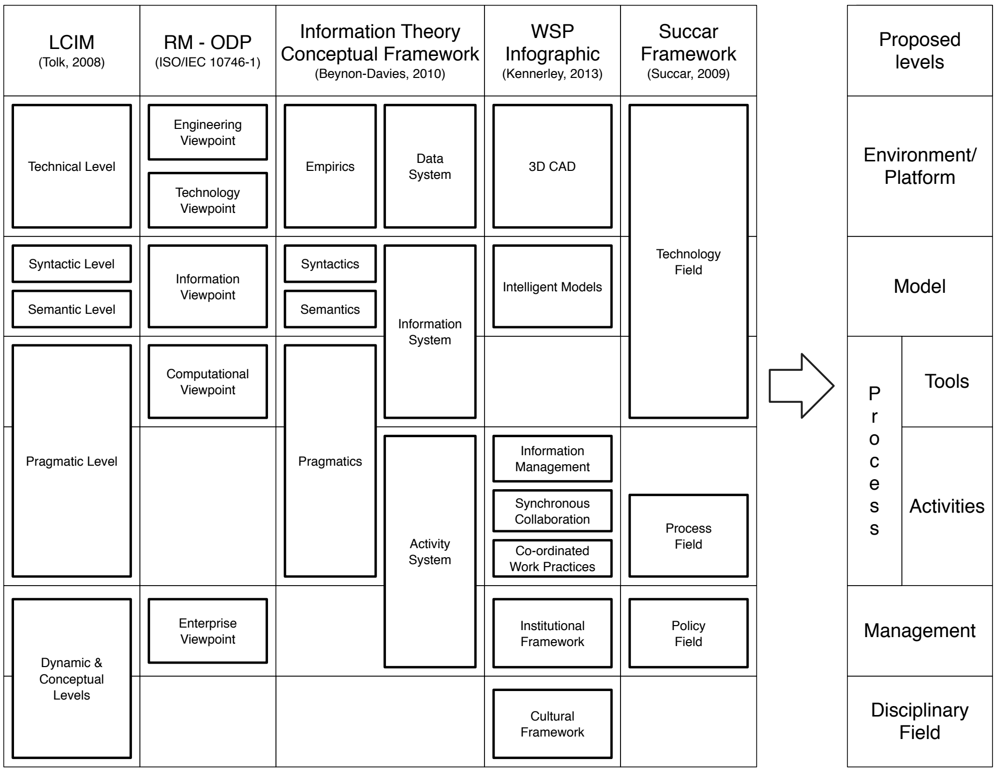


Figure 4.1: Alignment of frameworks with proposed levels

closer analysis of each layer is limited because the accompanying text carries no explanations. The infographic is helpful because it conveys that BIM is a socio-technical framework and that the discipline is more than just a 3D model within a Computer Aided Design (CAD) application. The infographic also highlights the relevance of the *institutional and cultural framework* as the outer layer, which will be explored later on in Section 4.4.7.

In the same context, Succar (2009) considered that BIM consisted of three overlapping and intersecting fields (Figure 4.3). A *technology* field is made up of the platform, model and tools levels. A *process* field corresponds to the routine business processes that facilitate design, construction and operation, and the *policy* field corresponds to the managerial functions that establish the protocols for information exchange.

The various levels of the Kennerley (2013) and Succar (2009) frameworks are have been mapped alongside the levels, viewpoints and steps of the LCIM, RM-ODP, DIHAS frameworks and arranged together in Figure 4.1. The resulting alignment indicates an underlying hierarchical pattern. It is proposed to use this hierarchical pattern as the

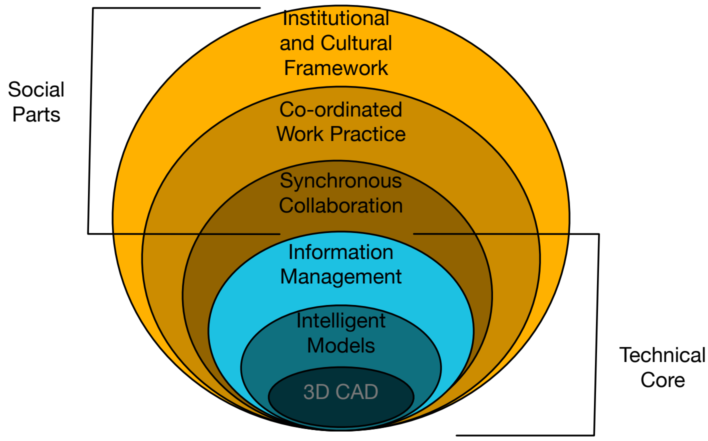


Figure 4.2: WSP infographic illustrating BIM as socio-technical system (Kennerley 2013)

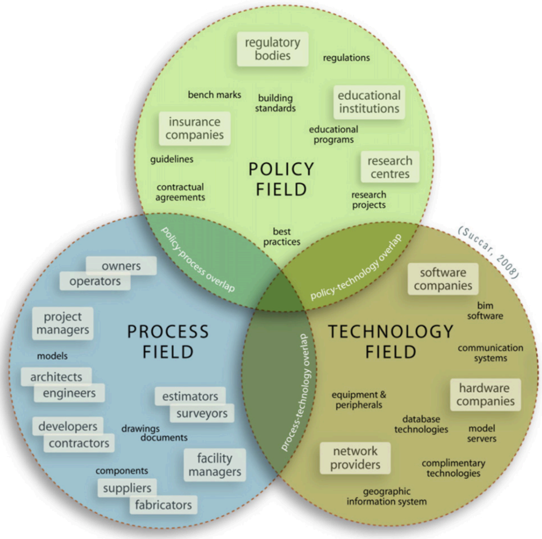


Figure 4.3: Intersecting technology, process and protocol BIM fields (Succar 2009)

foundation for a novel Spatial Information Systems socio-technical framework that will be further developed in this chapter.

4.2 Existing Definitions of BIM and GIS Terms

Before developing a new socio-technical framework, a review of accepted definitions in current circulation may also reveal similarities and differences between BIM and GIS. Starting with the ISO definition, as set out in the *Concepts and Principles used for Organising and Digitising of Information about Buildings and Civil Engineering Works* ISO 19650-1:2018 (ISO 2018), Building Information Modelling is "*the use of a shared digital representation of a built asset to facilitate design construction and operation processes to form a reliable basis for decisions"*.

The earlier definition used in the UK-based *Specifications for Information Management using Building Information Modelling* (PAS 1192-2:2013 and PAS 1192-3:2014) (BSI 2013, 2014) differs slightly to the one used in ISO 19650-1. The PAS definition instead refers to BIM as the "*process of designing, constructing or operating a building or infrastructure asset using electronic object-oriented information"*. The concept, that could be considered as a principal function in PAS 1192-2:2013, has now become a facilitating function in ISO 19650-1:2018. It would appear that the international definition is more in line with the one given by the US National Institute of Building Sciences (2015) where BIM is defined as a "*business process for generating and leveraging data to design, construct and operate a building"*. The PAS 1192-2:2013 definition instead implies that Building Information Modelling is a Human Activity System (Beynon-Davies 2010) supported by an Information System.

Over in the geospatial domain, there is no standardised definition of a Geographic Information System (GI System). Taking a description from an authoritative textbook, a GI System is a "*computer-based information system that enables capture, modelling, storage, retrieval, sharing, manipulation, analysis and presentation of geographically referenced data"* (Worboys and Duckham 2004). It is noteworthy that the acronym GIS can commonly refer to *Geographic Information Systems* in the plural. It is in this sense that Tsou (2018) describes GIS as a "*generalized concept for describing geospatial technologies, applications, and knowledge"* being an umbrella term for GI Systems, GI Services and GI Science.

These are, therefore, nuanced semantic differences between the definitions for BIM and GIS. The definition in ISO 19650-1:2018 and the PAS 1192 series focus on BIM as an activity or process, whereas GIS refers to a system for implementing a model or the general practice of using that system. This observation is supported by the comments of Hijazi and Donaubauer (2017) and Beck, et al. (2020). Furthermore, GIS is in essence agnostic with regards to the process, whereas BIM is dedicated to the process of designing, constructing, and operating buildings and infrastructure assets.

These semantic definitions can be expressed in graphical form using the symbology of a Unified Modeling Language (UML) diagram. The diagram in Figure 4.4 has been constructed using neutral terms equivalent to Building Information Modelling and Geographical Information System. The central column of the diagram shows the processes of design and operation being facilitated by *Spatial Information Modelling* using a *Spatial Information Model* similar to that defined by ISO 19650-1:2018. On the left side of the diagram, the term *Spatial Information Modelling* in line with the PAS 1192-2:2013 definition is shown in dashed form. A link is shown classifying the PAS 1192-2:2013 definition as being the same as the process of designing and operating, which is also classified as a Human Activity System (HAS).

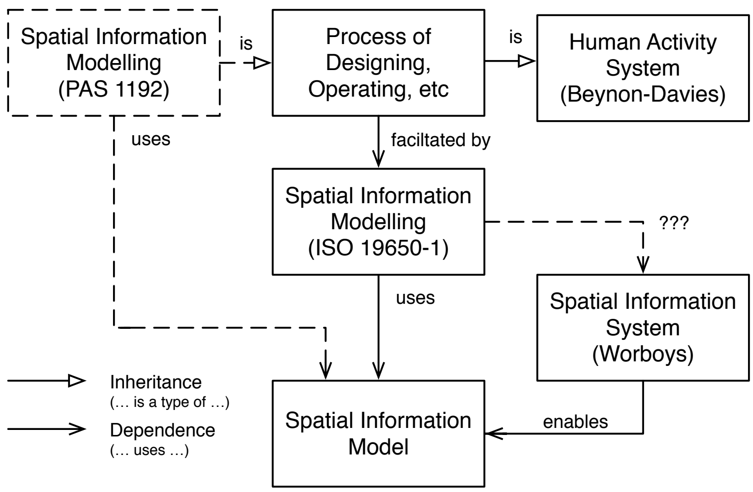


Figure 4.4: Spatial Information Modelling definition

The right-hand side of the UML diagram in Figure 4.4 shows a *Spatial Information System* as defined by Worboys and Duckham (2004) that implements a *Spatial Information Model*.

From this diagram, it is evident that the undefined relationship between *Spatial Information Modelling* and the *Spatial Information System* requires further investigation and clarification. It is necessary to ask whether the term *Spatial Information Modelling* is a gerundive such that it refers to *use of a Spatial Information System*, or whether the term is a noun, and thus it is a *Spatial Information System*.

If a *Spatial Information System* is narrowly understood to be just the hardware and software on which the information is hosted, then *Spatial Information Modelling* can be defined as the use of a *Spatial Information System*. But if a *Spatial Information System* is seen as a socio-technical system operating at a pragmatic level (Beynon-Davies 2010) then the use of *Spatial Information Modelling* must be viewed as a *Spatial Information System*.

It can be observed that a *Spatial Information System* (aligned with the Worboys and Duckham (2004) definition) is agnostic with regard to the HAS that it supports, whereas *Spatial Information Modelling* (aligned with the ISO 19650-1:2018 definition) has an intended purpose. Taking into account the hierarchical layout in Figure 4.4, it follows that a *Spatial Information System* would be lower in a hierarchical framework compared to *Spatial Information Modelling*.

Taking a higher perspective, Building Information Modelling and the processes of designing, constructing and operating a built asset can be seen as business processes within the context of an enterprise. The built asset has a value within an enterprise, and the BIM processes related to that asset must be aligned with the requirements of the enterprise. The higher functions of Building Information Modelling that direct and guide the enterprise processes are grouped together and referred to as *Building Information Management*. It should be noted that the term *Geospatial Information Management* is not generally defined, although that is not to say that the practice does not exist.

A convenient definition of *Building Information Management* does not exist in ISO 19650-1:2018. There are references to *Information Management* as "*the management and production of information during the life cycle of built assets"*, and the *Specification for Information Management using Building Information Modelling*, ISO 19650-2:2018, (ISO 2018a) sets out an extensive list of activities that constitute the practice of information management. Meanwhile, PAS 1192-2:2013 strictly defines *Information Management* as the "*tasks and procedures ... to ensure accuracy and integrity of information"*. This definition appears to be at odds with the title and context of the document which, like its ISO 19650-2:2018 counterpart, sets out the activities that constitute information management. The use of the term throughout the ISO 19650 series and PAS 1192-2:2013 generally follows the definition used by the National Institute of Building Sciences (2015) which defines *Building Information Management* as the "*organisation and control of Building Information Modelling by utilising the information [model] to effect the sharing of information."*

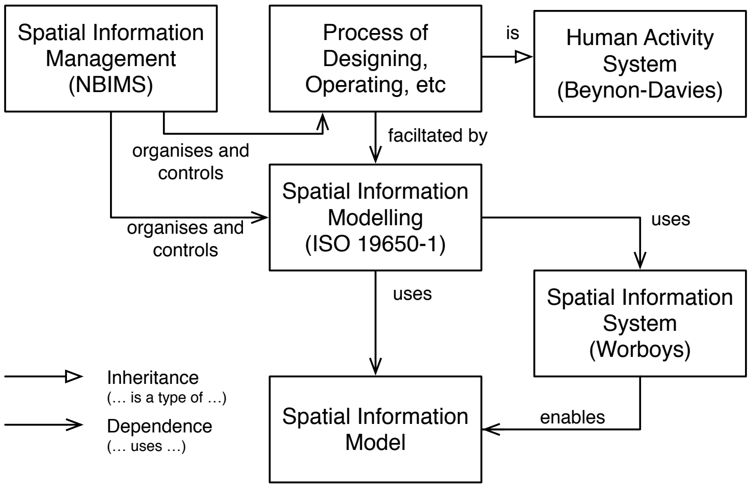


Figure 4.5: Spatial Information Management definition

How then should the relationship between *Building Information Management* and the Human Activity System be considered? In one sense, *Building Information Management* is just a higher-level function of *Building Information Modelling* and therefore it is merely facilitating the processes of the Human Activity System. It could also be argued that the management of communication between project stakeholders as set out in ISO 19650-1:2018 is a fundamental part of the design, construction and operation of built assets; so much so, that *Building Information Management* should be considered to be the organisation and control of the design and construction process.

Figure 4.5 shows how a generic definition of *Spatial Information Management* might fit into the previous UML diagram. The diagram shows *Spatial Information Management* as organising and controlling both *Spatial Information Modelling* and the Human Activity System.

This section has shown the complexity of describing the various elements of BIM and GIS which makes it especially difficult to compare the various definitions side-by-side. The confusing state of the definitions supports the creation of Spatial Information Framework with which to describe the hierarchical levels of both domains.

4.3 Proposed Spatial Information Systems Framework

The hierarchical pattern revealed in Figure 4.1 covering existing frameworks helps to understand the nature of socio-technical systems. It is proposed to use the levels identified in Figure 4.1 as a basis for comparing BIM and GIS and identifying their similarities and differences. This exercise aims to gain a deeper understanding of the terms as they currently stand and apply this knowledge for the purposes of resolving heterogeneity and increasing interoperability. Furthermore, this exercise will provide a framework for analysing the Crossrail Technical Information Systems and contribute towards identifying challenges that hinder interoperability.

The proposed framework consists of the following seven levels, as shown in the right-hand column of Figure 4.1 and the left-hand column of Figure 4.6.

- **Environment** level corresponding to the LCIM Technical level.
- **Platform** level also corresponding to the LCIM Technical level.
- **Model** level corresponding to the LCIM Syntactic and Semantic levels.
- **Processes** level corresponding to the LCIM Pragmatic Level, to be split into **Tools** level and **Activities** level.
- **Management** level loosely aligned with LCIM Dynamic & Conceptual levels.
- **Disciplinary Field** level loosely aligned with LCIM Dynamic & Conceptual levels.

Referring to the infographic used by Kennerley (2013), and also to the framework used by Succar (2009), a demarcation is evident between the technical core and the social levels. At the *Processes* level, it is apparent that some processes are computational and algorithmic, while others require human involvement and decision-making. For this reason, it is proposed to split the *Processes* level into a distinct *Tools* level encompassing computational tools and an *Activities* level covering manual workflows.

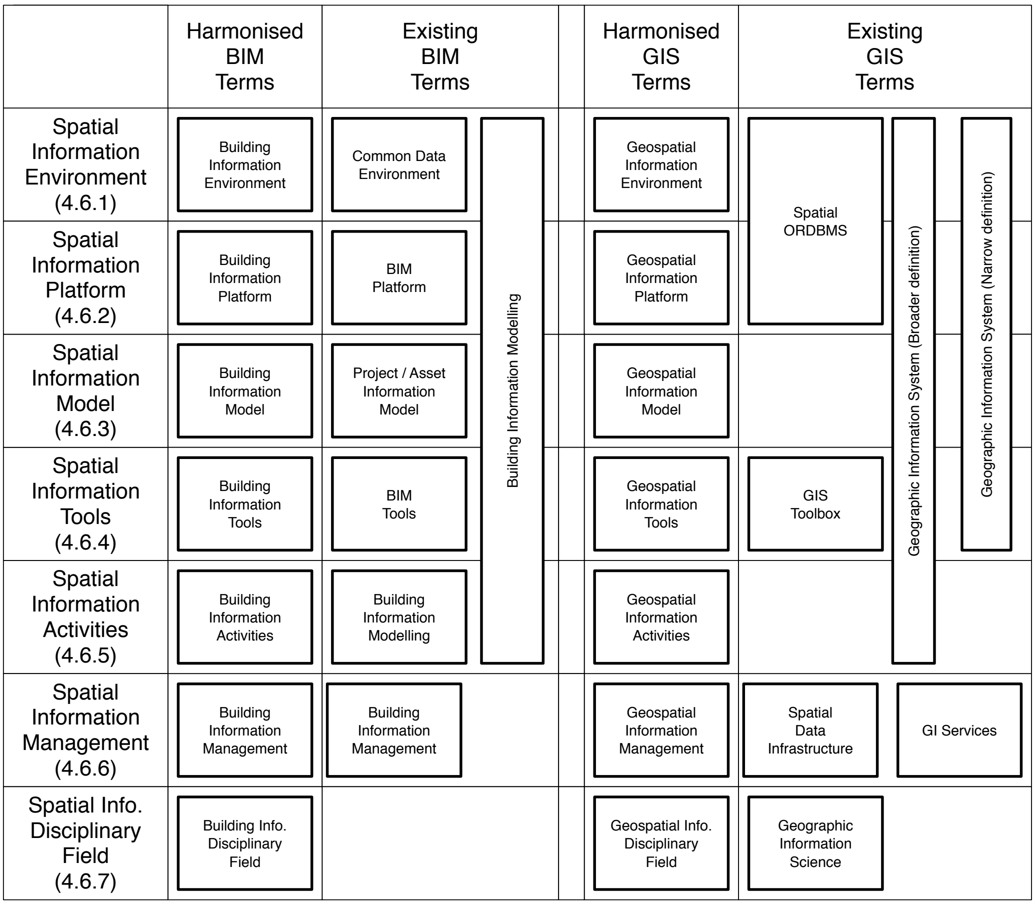


Figure 4.6: Proposed Spatial Information System Framework

Each of these levels will be explained in detail in the following section in the context of both BIM and GIS, and the similarities and differences between the two domains, at each level, will be explained. These levels will then be evaluated by applying them to describe the range of interoperability approaches found in literature that are summarised in Figure 3.18.

4.4 Applying the Framework to BIM and GIS

4.4.1 Spatial Information Environment

The base level of the socio-technical system proposed in Section 4.3 is the *Spatial Information Environment*. This level describes the technical systems which are used to store the information models that will be described in Section 4.4.3 in physical form. As well as the hardware, it also includes the file operating systems and database systems and the other software required to access the model.

Within the DIHAS Framework (Beynon-Davies 2010), the *Spatial Information Environment* level would act as the Data System where information is empirically represented in electronic form suitable for storage and transmission. The level can be described using the language of the *Technology* and *Computation* Viewpoints in the RM-ODP.

Eastman, et al. (2011) uses the terms *Environment*, *Platform* and *Tool* to describe the various technical systems that support BIM, and it is proposed to extend the general meaning of the first two terms to describe the technical systems levels, and use *Spatial Information Tool* in Section 4.4.4. Distinguishing the terms, the *Spatial Information Environment* is the network architecture and hardware and data management software that enables information storage and retrieval. The information in it can be structured as an Relational Database Management System (RDBMS), a file management system, or alternative systems such Resource Description Framework (RDF) and NoSQL storage.

Although the widespread implementation of technical standards (Tolk 2006) covered by the Transmission Control Protocol/Internet Protocol (TCP/IP) model ensure reliable exchange of data anywhere, there are still challenges that affect data access within the *Spatial Information Environment*. The effect of slow network access and bandwidth limitations need to be considered on the practical user experience. Furthermore, the use of firewalls, encryption and security privileges will prevent interoperability if the network privileges are not configured correctly.

4.4.1.1 Building Information Environment

Within the context of the BIM domain, the *Building Information Environment* is generally referred to as the Common Data Environment (CDE) which provides all stakeholders with shared access to project or asset information.

Building Information Models are structured as federated models in that they consist of a federation of standalone models that can be linked together. The most prevalent storage method is to save each component model as an individual file. Each file is then managed by a file management system which controls access depending on users' privileges, checking the file in and out to prevent inconsistent changes during concurrent access. More advanced *Building Information Environments* exist in database form; however, the overall structure of the model is still maintained as a federation.

The proprietary software publishers generally provide *Building Information Environment* products to their customers on commercial terms. *Autodesk* customers can choose between hosting *Revit* models on an in-house server using *Revit Server* or accessing models in the cloud using *Autodesk BIM 360*. Similarly, *Bentley* customers may decide to use *ProjectWise* on their own servers or *ProjectWise 365* in the cloud. Open-source *Building Information Environment* software is also available through the *BIMserver* group for running on managed servers, however, cloud services are not available for free (van Berlo and Krijnen 2014).

4.4.1.2 Geospatial Information Environment

Just as in the BIM domain, geospatial information can be stored as standalone files on a desktop computer, and this environment is likely to be used for an individual project run by a small team. However, it is more predominant in commercial practice to set up a *Geospatial Information Environment* that provides concurrent multi-user access to a remote Object Relational Database Management System (ORDBMS) across a large organisation. The popular database installations provide functionality for storing geometric objects, such as the proprietary *Oracle Spatial* and the open-source *PostGIS* extension to *PostgreSQL*. These popular database installations can be configured to store data in a multi-user *geodatabase* for applications in the *ArcGIS* platform through the installation of *ArcSDE* software (ESRI 2020).

4.4.1.3 Similarities and Differences

The *Spatial Information Environments* for both BIM and GIS domains are similar in that they provide systems in each domain with a common environment within which to exchange information and only differ in that file-based environments are predominant in BIM, and ORDBMS environments are predominant in GIS.

4.4.2 Spatial Information Platform

The *Spatial Information Platform* is the software that enables the *Spatial Information Tools* to access the *Spatial Information Model* that is stored in the *Spatial Information Environment*. The use of the *Spatial Information Platform* is most apparent when proprietary software applications do not allow unfettered access to the information model. However, a standalone platform application with a Graphical User Interface (GUI) may be unnecessary if *Spatial Information Tools* can directly access openly documented information models, or access them through a licensed Software Developer Kit (SDK).

4.4.2.1 Building Information Platform

The *Building Information Platform* is the interface between the *Building Information Tools* and the *Building Information Model*. In the context of proprietary software, *Autodesk Revit* and *Bentley MicroStation* are examples of well-known platform applications that provide the user with a GUI environment for creating and viewing BIM models. These platforms also provide functionality for users to develop their own generative designs and automated scripting tools such as *Dynamo* and *Bentley MicroStation Visual Basic Application (MVBA)*.

Bentley MicroStation is a proprietary CAD application that provides a platform within which additional software referred to as extensions can be run. The functionality to support BIM is delivered through the use of the BIM extensions such as *Bentley Architecture*, *Bentley Structural Modeller*, *Bentley Building Electrical Systems* and *Bentley Building Mechanical Systems*. The names of these extensions have a tendency to change in line with *Bentley* product branding, and for a while, these BIM extensions were bundled together and marketed under the *AECOsim* brand.

Open-source BIM platforms are also available, for example *FreeCAD* (FreeCAD 2020) which is a GUI editor that is built using the *OpenCascade* CAD libraries (Open Cascade 2020) and uses the *IfcOpenShell* library (Krijnen 2020) to provide functionality to read and write Industry Foundation Classes (IFC) files. *Python* wrappers are also available for these libraries enabling programs to be written in a scripting language to access IFC files.

4.4.2.2 Geospatial Information Platform

A *Geospatial Information Platform* provides programmatic access to information stored in the *Geospatial Information Model*. This access may be provided through the use of desktop applications such as the *ArcGIS* suite of software from *Esri* or an open-source product such as *QGIS*. An alternative set-up is for ORDBMS software to provide *Geospatial Information Tools* with the ability to run spatial queries on geospatial information from within the database.

Cloud-based *Geospatial Information Platforms*, such as *ArcGIS Online*, provide an alternative to visualising information in desktop applications. These services run on remote servers and provide the user with a graphical interface through a web portal running on a browser or smartphone application.

In many instances, it is possible to avoid the use of a dedicated Geospatial Information Platform and software can be written to access data through the use of Application Programming Interfaces (APIs) and SDKs. For example, a developer can write a *Python* script that calls the *ArcPy* module to access information on the *ArcGIS* platform or calling the *geopandas* package that enables access to standard geospatial information formats.

4.4.2.3 Similarities and Differences

Spatial Information Platforms are fundamentally similar in that they provide a means for *Spatial Information Tools* to access *Spatial Information Models*; they only differ because the *Tools* and the *Models* are different. Indeed, a *Spatial Information Platform* can provide access to both BIM and GIS information within the same application. As an example, the ability to access multiple *Spatial Information Model* formats is the primary feature of specialised interoperability software as provided by the *FME* suite of applications. The underlying technology used in *FME* is included in *ArcGIS* software as the *Interoperability Toolbox*.

4.4.3 Spatial Information Model

The *Spatial Information Model* plays an integral role in the *Spatial Information System* as it provides a structure for extracting relevant, pragmatic details about real-world phenomena. These abstractions aim to be able to implement information in an empiric form able to be stored, communicated and analysed. The empiric representation of the *Spatial Information Model* is physically performed within the *Spatial Information Environment Spatial Information Platform*, as described in the previous subsections.

This reinterpretation of reality into a more suitable form is foundational to the art and science of modelling. The process of observing the real world (or designing a world for future realisation) and abstracting the relevant information follows a series of steps. These steps are referred to as the semiotic ladder by (Beynon-Davies 2010) in his DIHAS framework (Section 3.2.1) or as a waterfall by Worboys and Duckham (2004) (Section 3.2.3). Within the context of this proposed framework, the employment of the *Spatial Information Model* corresponds to the semantic and syntactic rungs on the semiotic ladder and the Conceptual Computational model and Logical Computational model in the waterfall model.

In addition to the theoretical approach, research carried out by Bishr (1998) led him to conclude that divergent abstractions in the modelling process cause heterogeneities between models that ultimately result in system interoperability. From the nine layers of abstraction used by the Open Geospatial Consortium (OGC), Bishr identified three types of heterogeneity described as *semantic*, *schematic* and *syntactic* (Section 3.3.1).

It should be noted here that the development of model schemata is an iterative process (Wise 2010). In the first instance, *Spatial Information Systems* use generic model formats that are designed for use on a particular Spatial Information Platform. However, a specialised area will require a bespoke schema to be developed that supports the needs of that situation.

These frameworks, in particular the provision of semantic, schematic and syntactic levels, provide a basis for understanding *Building Information Models* and *Geospatial Information Models* and the standard formats and schema that used to create them.

4.4.3.1 Building Information Models

In the context of BIM, ISO 19650-1:2018 avoids the term *Building Information Model* but instead refers to an *Information Model* in the delivery and operational phases differently. The standard instead refers to a Project Information Model (PIM) in the delivery phase and an Asset Information Model (AIM) in the operational phase. An *Information Model* is defined as "*a set of structured and unstructured information containers"*, where the term *Information Container* is used to describe any persistent electronic information source. In this sense, structured information refers to geometrical models and database data, while unstructured information refers to documents created for human interpretation.

Despite the distinction of terms in ISO 19650-1:2018, the term *Building Information Model* is still widely used throughout academic and industry literature. As such, the *US National BIM Standard* (National Institute of Building Sciences 2014) defines a *Building Information Model* as a "*digital representation of the physical and functional characteristics of a facility and its related project/lifecycle"*. It seems that the term used in this context is closer to the definition of structured information in ISO 19650-1:2018 rather than the expanded definition of *Information Model* that extends to include unstructured information. The rest of this section will, therefore, use the term *Building Information Model* to refer to this structured 3D digital representation of a built asset.

It is normal within the Architectural, Engineering and Construction (AEC) industry to use proprietary software applications (incorporating *Building Information Platform* (building-information-platform) and *Building Information Tools* (building-information-tools)) for the design and construction of Project Information Models (PIMs). These proprietary software applications use their own proprietary building information model schema to create models. With this, these proprietary information models can hold information to support generative parametric design. Significant research and development have been invested by the commercial software providers to developing parametric design functions that automate many of the procedural aspects of design (Eastman, et al. 2011). Consequently, these proprietary information models support features that give software publishers a competitive edge over their rivals.

The AEC and Asset/Facilities Management (AM/FM) industries, together with the software publishers, have instituted the formation of *buildingSMART* and organisation tasked with the development of the IFC schema. Models created using the proprietary data formats can be exported using the IFC format and exchanged with other applications. The descriptions that follow will use the IFC schema to describe the semantics, schematics and syntactics of *Building Information Models*.

Semantics and Application Domain – As described in Section 3.2.3, the practical art of modelling is underpinned by the consensual abstraction of an *Application Domain* by which the domain community hold an understanding of the phenomena and processes that are mutually agreed within their domain. The implementation of the conceptual and pragmatic levels of this *Application Domain* will be described in the later sections of this chapter, while this section shall address to the semantic, schematic and syntactic concerns. Although this framework is structured in terms of distinct levels, it should always be remembered that the process of modelling is less clear cut and there is an element of reciprocity among the levels.

The abstraction of the *Application Domain* is foundational to the semantic development of a model, and the subsequent schematic and syntactic development (Bishr 1998). The way that terms are understood influences the schematic structure and subsequently, the syntax chosen to represent that structure.

Within an academic or professional discipline, there is no single document that is capable of fully describing the *Application Domain*. For the geospatial community, the OGC encourage the use of *Abstract Specifications* to document a mutually accepted understanding of the domain. The rudimentary concepts that underpin the development of models in the AEC and AM/FM models were published in as the *General Architectural Research Model* (Gielingh 1988) and the *Building and Construction Core Model* (Wix and Bloomfield 1995). However, these documents are 30 years old, and there is no system for reviewing and augmenting their scope and content.

Without a standard *Abstract Specification* an ensemble of semantic systems has developed within the BIM domain. These semantic systems can be classified as either schemes of classification or as computational object classes. The schemes of classification are governed by the ISO 12006-2:2015 *Framework for Classification of Information about Construction Works* (ISO 2015). Different schemes exist that are suited to local practices and national dialects; these include the UK *Uniclass* (NBS 2015) and the US *Omniclass* schemes (Construction Specifications Institute 2020).

The second type of semantic system concerns the naming of object classes in model schemas, such as the IFC schema and the proprietary schemas developed by *Autodesk* and *Bentley*. It should be noted that the semantic classification of phenomena influences the structure of the schema. In light of this, this subsection shall consider the semantic concerns while the next subsection shall look more closely at the schematic issues.

The Industry Foundation Classes (IFC) have been progressively developed over the past 25 years by a consortium of stakeholders convened from across the AEC and AM/FM industry and the software industry. The classes constitute an object-oriented model schema that is used to represent built assets, both in the design and construction phase and in the asset management operational phase. Each physical entity that makes up a built asset belongs to a class of objects such that the attributes and behaviour of each instantiated object are inherited from a hierarchy of parent object classes.

At the same time, each physical entity can be grouped with other physical objects to make up aggregate objects. For example, *IfcStairFlight* objects are aggregated to form an *IfcStair* object. Furthermore, each physical entity can belong to a spatial composition. For example, the *IfcStairFlight* object can belong to the *IfcSpace* object, i.e. the stairwell. It can also belong to an *IfcBuildingStorey*, an *IfcBuilding*, an *IfcSite* and *IfcProject*.

The IFC schema is not just used to model physical entities, i.e. the product of construction, there is capacity within the schema to model processes, control, resources, actors, projects and groups. The IFC schema does not just model the 3D physical representation as entities, it also has the functionality to display views, i.e. 2D representations of entities such as sections, plans and elevations.

As well as IFC, other proprietary schemas are widely used in the AEC industry. The proprietary schemas use different semantics to IFC, not just in the names used but also how entities relate to one another within the schema. For example, *Autodesk Revit* uses *family* instances to aggregate entities, and *Bentley AECOsim* uses *families* and *parts* to implement similar functionality.

Schematics and Information Schema – The model schema contains the hierarchical structure of information within the model. In principle, the schema is agnostic to the choice of syntax, but in practice, they are self-influential. The schema used by IFC is publicly available and well documented. The IFC schema itself is available in both *EXPRESS* and XML Schema Definition (XSD) formats (buildingSMART 2020a).

Figure 4.7 illustrates a line of STEP file format (STEP) data taken from file *LPL-C-1-41052* (one of the files listed in Table 7.1 that will be used for evaluating the Extract-Transform-Load (ETL) operations in Chapter 6). This line corresponds to an element belonging to the *IfcBuildingElementProxy* class representing a reinforced concrete retaining wall.

```
#2050= IFCBUILDINGELEMENTPROXY('0Aej7yTjX5kRK94JjGwj6i',
       #16, 'Insitu Reinforced Concrete--Retaining Walls',
       '0, LPL-C-1-41052.dgn, Default:287503',
       'Insitu Reinforced Concrete:Retaining Walls',
        #2215, #1477, $, $);
```

Figure 4.7: IFC data representing IfcBuildingElementProxy

This line of data has been broken up, and the nine position-based attributes for the element are summarised in Table 4.1. The first three attributes include a GUID, an *owner*

Table 4.1: Summary of IFC data representing IfcBuildingElementProxy

Inherited Class	Attribute Name	Attribute Data	Description
IfcRoot	GlobalId	'0Aej7yTjX5kRK94JjGwj6i'	Unique reference
IfcRoot	OwnerHistory	#16	Reference to #16
IfcRoot	Name	'Reinforced Concrete–Retaining Walls'	
IfcRoot	Description	'0, LPL-C-1-41052.dgn, Default:287503'	Source File and Element ID
IfcObject	ObjectType	'Reinforced Concrete:Retaining Walls'	
IfcProduct	ObjectPlacement	#2215	Reference to #2215
IfcProduct	Representation	#1477	Reference to #1477
IfcElement	Tag	$	Null character
IfcBuilding ElementProxy	CompositionType	$	Null character

history reference used to keep track of edits and a string used to hold a name for the element.

The fourth attribute contains a description string. In this case, the attribute has been populated by *Bentley AECOsim* with information on the source of the element, i.e. the source file and the *MicroStation* element ID. This attribute is followed by an *ObjectType* attribute containing information on the *AECOsim* object family/part from which the element was sourced.

The *ObjectPlacement* and *Representation* attributes are references to another line of data in the same IFC file. These lines are reproduced in Figure 4.8. The *ObjectPlacement* attribute refers to an *IfcLocalPlacement* entity that contains information on the geometry insertion point. The *Representation* attribute refers to an *IfcProductDefinitionShape* entity that lists all the geometric representations associated with that element, namely a simple bounding box and the full Boundary Representation (B-Rep) geometry (Section 3.5.1).

```
#16   = IFCOWNERHISTORY(#15,#11,$,.MODIFIED.,$,$,$,1574676192);
#2215 = IFCLOCALPLACEMENT(#2211, #3442);
#1477 = IFCPRODUCTDEFINITIONSHAPE($, $, (#1729, #1730));
#1729 = IFCSHAPEREPRESENTATION(#1306, 'Body', 'Brep', (#1211));
#1730 = IFCSHAPEREPRESENTATION(#1307,'Box','BoundingBox',(#1312));
```

Figure 4.8: IFC data referenced by IfcBuildingElementProxy

In addition to attribute information contained within a line of IFC data, the IFC schema adopts the use of inverse references to link information to entities. This use of inverse references provides the ability to cater for many-to-many relationships within the schema.

Figure 4.9 contains those lines of IFC data that relate to the *IfcBuildingElementProxy* described above in Figure 4.7 and these inverse references are summarised in Table 4.2 together with some additional lines (which in this case do not link to other entities). These inverse references record that the element is associated with the material type of *Concrete* and that it is defined by a *Retaining Wall* element type. The element has also been allocated to a specific storey in the building model named *Roof*. Finally, the element is also linked to a set of properties that typically belong to roof elements. These property sets are used to bridge the gap between the information in proprietary models, i.e. *MicroStation* and *IFC*. They are capable of extending IFC to meet the needs of specific projects.

```
#221 = IFCRELDEFINESBYTYPE('3uD5bCd3H4tQ7J86$WK0IV',
       #16, $, $, (#2050, #2055, #2056), #224);
#224 = IFCBUILDINGELEMENTPROXYTYPE('1I6IZNJaX6gfcR6dCgRP_q',
       #16, 'Insitu Reinforced Concrete:Retaining Walls',
       $, $, $, $, $, $, .NOTDEFINED.);

#228 = IFCRELASSOCIATESMATERIAL('08KcuzhHvC_uIbIA$bgkSF',
       #16, $, $, (#2050), #387);
#387 = IFCMATERIAL('Concrete');

#1471= IFCRELCONTAINEDINSPATIALSTRUCTURE('2BPUBD6aX4jONs4F8fNOxq',
       #16, $, $, (..., #2060, ...), , #1472)
#1472= IFCBUILDINGSTOREY('0W4cqEUHr7L9hCEt66chqb',
       #16,'RF','Roof', $, #2211, $, '', .ELEMENT., 0.);

#51  = IFCRELDEFINESBYPROPERTIES('3IxFuNHRvBDfMT6_FiWPEz',
       #2, $, $, (#2050), #52);
#52  = IFCPROPERTYSET('18RtPv6efDwuUOMduCZ7rH', #2,
       'Pset_Roof', $, (#53, #54));
#53  = IFCPROPERTYSINGLEVALUE('FireRating', 'FireRating',
       IFCTEXT(''), $);
#54  = IFCPROPERTYSINGLEVALUE('IsExternal', 'IsExternal',
       IFCBOOLEAN(.T.), $);
```

Figure 4.9: IFC data referencing IfcBuildingElementProxy

This example illustrates that information can be structured in four ways in the IFC schema. Information is captured by the class of object, by attributes, by references to other entities, and through inverse relationships.

Table 4.2: Summary of IFC data inversely referencing IfcBuildingElementProxy

Inherited Class	Inverse Name	Inverse Reference	Description
IfcObjectDefinition	HasAssignments		
IfcObjectDefinition	IsDecomposedBy		
IfcObjectDefinition	Decomposes		
IfcObjectDefinition	HasAssociations	#228	Reference to material
IfcObject	IsDefinedBy	#51	Reference to property set that includes fire rating and whether external
IfcObject	IsDefinedBy	#221	Reference to Reinforced Concrete object type
IfcProduct	ReferencedBy		
IfcElement	FillsVoids		
IfcElement	HasOpenings		
IfcElement	ProvidesBoundaries		
IfcElement	ContainedInStructure	#1471	Reference to 'Roof' storey

The ability of the IFC schema to provide multiple representations within the same model using the *IfcProdcutDefintionShape* entity is also observable from this example. In this example, the element is represented by both a bounding box and B-Rep, but the set of representations could be extended to include other forms of representation.

Internal Spaces

As well as building elements, the IFC schema is capable of storing information on internal spaces. Figure 4.10 shows a UML diagram of the relationship between an *IfcWall* building element and an *IfcSpace* element. Both entities belong to classes that inherit from *IfcProduct*, and as such, both entities are capable of being represented by *IfcSolidModel* geometry which is referenced through attributes.

The *IfcWall* has an inverse relationship with an *IfcRelContainedInSpatialStructure* relationship entity to show that it is contained within an *IfcSpace*. At the same time, the *IfcSpace* has an inverse relationship with an *IfcRelSpaceBoundary* relationship entity to show that the *IfcWall* is a boundary of the *IfcSpace*. This *IfcRelSpaceBoundary* entity has an optional attribute that can hold a geometrical representation of that boundary surface.

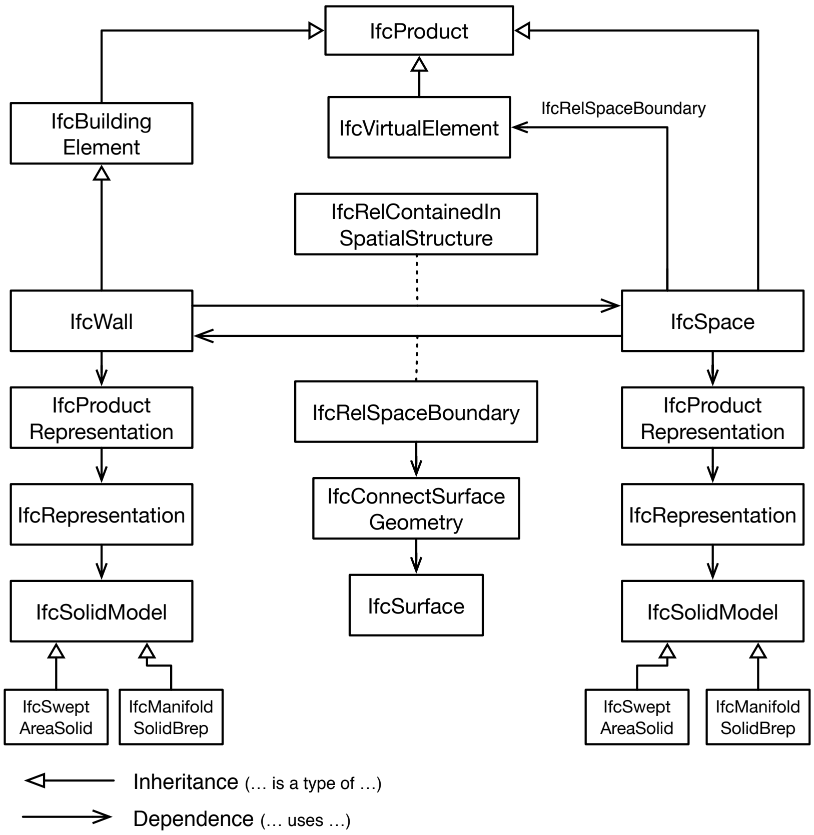


Figure 4.10: IFC spatial relationships

Syntactics and File Format – There is a range of formats used in the BIM domain as the primary software vendors provide their own proprietary software formats. One of the principal proprietary formats is the *RVT* format that supports *AutoDesk Revit*. In the *AutoDesk* ecosystem, this is a distinct BIM format in addition to the *DWG* format that can be used to store plain 3D CAD models. Both formats are established within the AEC industry, and although they remain proprietary ownership of *AutoCAD*, the formats are available for third-party developers to use; however, the community does not drive the standard nor have control over it.

Bentley delivers its BIM software products as extensions to the *MicroStation* platform. Individual extensions save information stored in families and parts that belong to that extension under the auspices of the *DGN* file extension. Parts of a model created using different BIM extensions are saved in separate files that are coordinated together to form a federated model. Access to the *Bentley* BIM model format is more tightly controlled than the equivalent formats in the *Autodesk* environment, and it is not common to see the *DGN* format used outside the *Bentley* ecosystem.

The IFC schema is syntax neutral as advocated by *buildingSMART* (buildingSMART 2020a) and, as such, models created using IFC can be stored and exchanged using the most appropriate format. *buildingSMART* supports the use of both the STEP format and Extensible Markup Language (XML).

4.4.3.2 Geospatial Information Models

Geospatial systems are different from BIM systems in that their use is not constrained to one particular industry or academic field. As well as surveying, design, construction and asset management, activities that are more familiar within the context of the AEC and AM/FM industries, geospatial systems are used extensively in other fields including demographics, epidemiology and meteorology.

It is not practical or even possible to develop a universal information model that can be used in all situations (Peuquet 1984). Each discipline that uses geospatial information must develop models that are pragmatic for their own purposes. However, there are situations where bespoke information must always interoperate with other information, if only to provide users with base-map topography as a point of reference.

To enable disparate GIS models to be used together, the ISO 19100 suite of standards is published by International Standards Organisation Technical Committee (ISO/TC) 211 to support interoperability between information sources. The OGC approve the formation of working groups in particular subject areas to develop interoperable standards within their respective communities,(e.g. CityGML Standards Working Group and the Integrated Digital Built Environment (IDBE) Joint Working Group).

The OGC provides standards that set out conceptual modelling methods for developing a shared understanding of geographic information semantics (Roswell 2012). These standards work together to provide mutually recognisable concepts that can be used together.

The OGC document their standards by following consistent modelling methods as described by Herring (1999). The OGC broadly follows the concepts of the *Application Domain*, *Conceptual* and *Logical Computational Models*, as described by Worboys and Duckham (2004) but instead uses the terms *Essential* and *Abstract* model (the two together collectively known as the *Abstract Specification*), and *Implementation Standard* (also referred to as *Implementation Specification*).

Semantics and Application Domain – The OGC *Abstract Specification* is published as a suite of documents each referred to as a *Topic*, some of which have been adopted as ISO standards. Many of these abstract specification documents describe an essential model describing the relevant topic in natural language. For example, *Topic 5* relating to *Features* (Kottman and Reed 2009) contains the essential model that describes the concepts used to develop the abstract specifications for geographic features.

The OGC also coordinate the development of standards for specific domains, for example, the *Land and Infrastructure Conceptual Modelling Standard* (LandInfra), a standard supporting the modelling of land and infrastructure features (OGC 2016). These specialised standards contain a dictionary of terms and relationships defining the semantics of the domain.

Schematics and Information Schema – Within the abstract specifications, the OGC develop abstract models from the essential models. These abstract models are conceptual computational models for core concepts that are common across all implementation specifications. These core concepts include specifications for geometry (ISO 19101:2005) and features (Kottman and Reed 2009).

The implementation specifications are published as formats which are syntactically complete. However, their authorship is still reliant on the abstraction of conceptual computational models from their own application domains.

The standard geometric model used by OGC standards is described in ISO 19101:2005 (ISO 2005). The UML diagram in Figure 4.11 illustrates the fundamentals of this geometric model as implemented in the OGC *Simple Feature Access* model (Herring 2011). Each geometry type inherits from a common *Geometry* object class. The more complex geometry types are made up of instances of the simpler geometry types. In this way, a *Polyhedral Surface* consists as a collection of *Polygons*, a *Polygon* consists of an exterior *Linear Ring* and a collection of interior *Linear Rings*, and a *Linear Ring* consists of a series of *Points*. *Oracle Spatial* has developed a similar geometric model implementation, *SDO_GEOMETRY* (Kothuri, et al. 2007) also conforming with ISO 19107 (ISO 2005a), but extending to include solid geometry types (Figure 4.12).

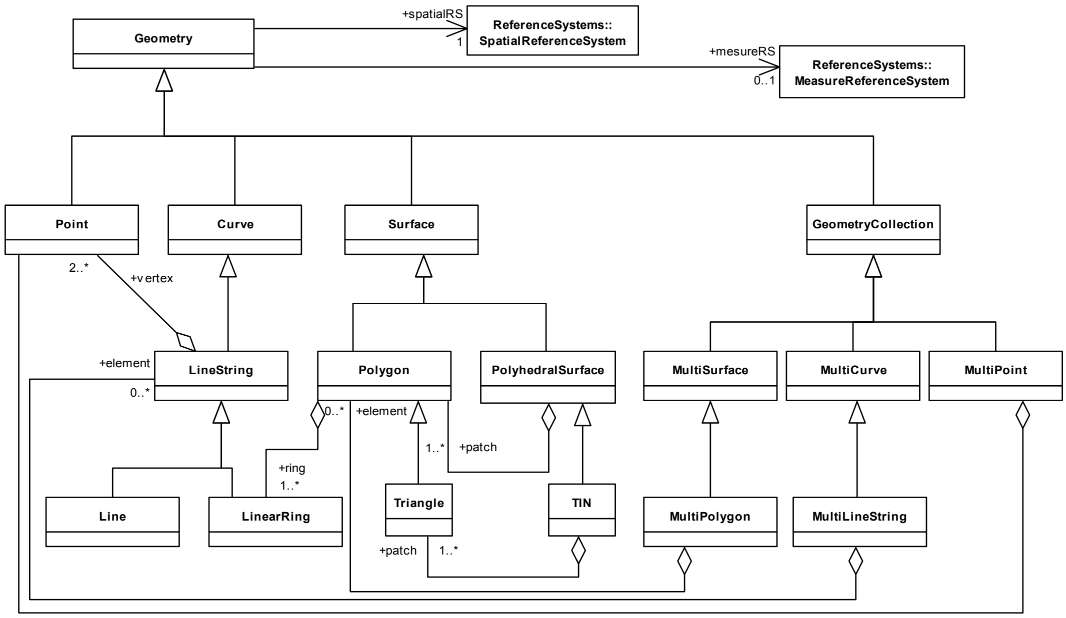


Figure 4.11: OGC Simple Feature Access geometry schema (Herring 2011)

Geospatial schema are not restricted to vector geometry. When deciding the most appropriate schema, it may be appropriate to sub-divide a 2D area in regular size pixels. When the same principle is applied to 3D sub-divisions, each element is referred to as a voxel. Each pixel (or voxel) is assigned a value that either represents a value contained within that pixel, e.g. annual rainfall or is a reference value corresponding to a defined list, e.g. land use.

The City Geographical Mark-up Language (CityGML) 2.0 standard (Gröger, et al. 2012) is an extension of the GML standard. The standard was developed to represent buildings and other built assets in an urban environment. The current schema is structured to represent building at five Levels of Detail (LoDs). The first of these, *LoD 0* is a generalised 2D building plan, *LoD 1* incorporates 2.5D building heights, *LoD 2* incorporates roof

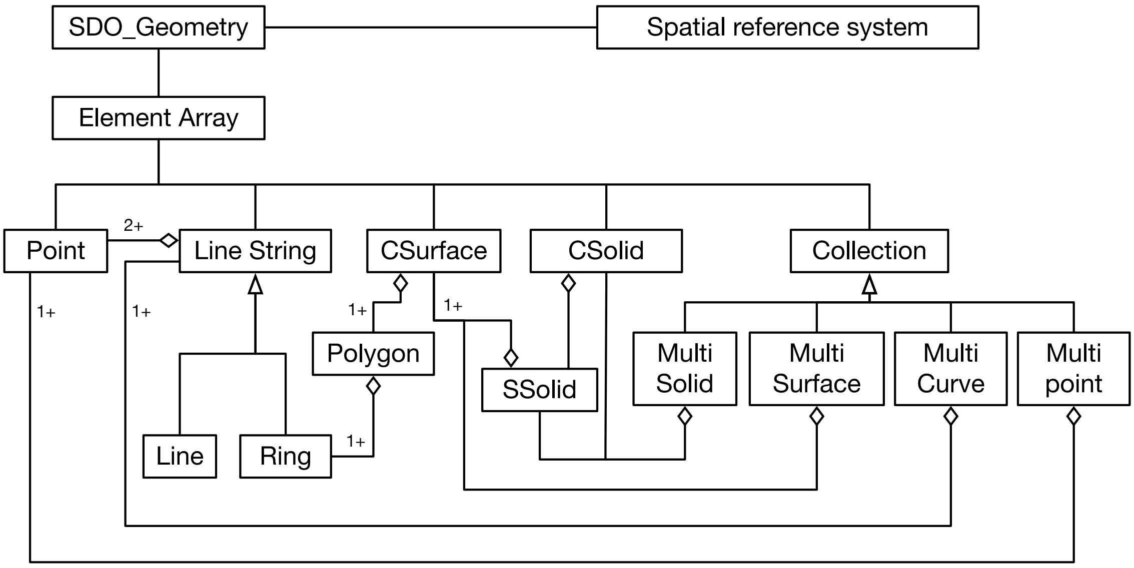


Figure 4.12: Oracle Spatial SDO_GEOMETRY schema (Kothuri, et al. 2007)

geometry and *LoD 3* incorporates façade features such as doorways and windows. The final level, *LoD 4*, provides a schema for the representing interior rooms.

The conceptual model for the CityGML standard is undergoing a major revision. In the new model, LoD is no longer associated with the top-level city object (i.e. building) but instead each component in the building has its own LoD. As such, the exterior of the building can be represented in one LoD while the interior is represented in another. As part of this revision, the original *LoDs 0-3* have been retained, but *LoD* 4_ has been dropped.

The schema of the CityGML can be extended through the use of Application Domain Extensions (ADEs); for example, Hijazi, et al. (2010) developed *UtilityNetworkADE* for representing gas, water, and electricity supplies within a CityGML model, and Agugiaro, et al. (2018) proposed an Energy ADE to support the use of building energy modelling on CityGML models.

InfraGML is the encoding standard of the Land and Infrastructure Conceptual Model Standard (LandInfra) conceptual model standard (OGC 2016). The format was developed as a replacement for the original *LandXML* format that was not fully compatible with OGC standards (Kumar, et al. 2019). LandInfra and *InfraGML* provide schema for land features, facilities and projects, alignments, roads, railways, survey data and land division. The standard was developed in conjunction with the CityGML Standards Working Group (SWG) and *buildSMART* as there are many modelled phenomena that are shared with CityGML and IFC (Kumar, et al. 2019).

InfraGML is still in its infancy and there is, as yet, no software support for the standard, whereas there is still limited support for its predecessor *LandXML*. Despite this, Kumar, et al. (2019) believe that the format has potential for solving BIM/GIS interoperability challenges by providing a GIS compatible schema for infrastructure phenomena.

Interior Spaces

With regard to the representation of interior spaces, the CityGML 2.0 schema differs from the IFC schema in that the building elements such as walls and slabs are absent. The CityGML 2.0 schema is instead focussed on the planar surfaces that enclose rooms.

A UML diagram representing the relationships between rooms and surfaces is illustrated at Figure 4.13.

As already described in Section 3.9, the soon-to-be-expected CityGML 3.0 model has undergone a major review to incorporate *AbstractSpace* objects and *AbstractSpaceBoundary* objects. An updated UML diagram representing the relationships between rooms and surfaces is illustrated at Figure 4.14.

Syntactics and File Format – *Geospatial Information Models* can be implemented using either open formats or using proprietary formats. Different open standards are available for different purposes. The principal open format for vector geometry is Geographical Markup Language (GML) (ISO 2020b) which follows the XML syntax. GML is designed to be readable by both machines and humans, but as a consequence data file sizes are larger than other formats for storing the same information. Another common open format

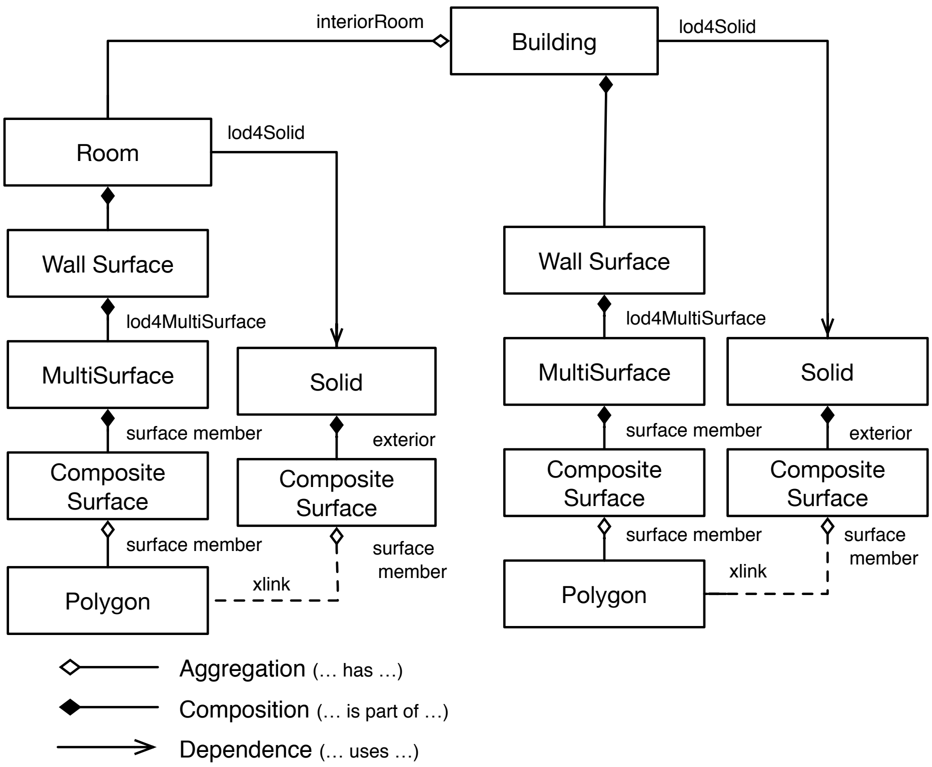


Figure 4.13: CityGML 2.0 schema (Interior Rooms)

is *GeoJSON* (Gillies, et al. 2016) which was originally designed for web mapping applications, but is now widely used in other contexts.

The de-facto standard format for exchanging vector geometry information is the *Esri Shapefile*. The format is under the proprietary ownership of *Esri* although the documentation is open and there are no technical restrictions preventing its use. The format provides for a geometry file, an index file and a database file containing a table attributes. As well as 2D geometric entities, *Esri Shapefiles* are capable of storing 3D surfaces and solids as *Multipatch* objects.

Although in everyday use, the *Shapefile* is considered to be obsolete (ESRI 2019). As such, *Esri* now promotes the *geodatabase* as their preferred means of exchanging geospatial information. As well as vector data, the format is capable of holding raster data, when accessed directly from an *Esri* application. Third-party access to vector data is read-only unless the licensed API SDK is downloaded to gain the ability to write data.

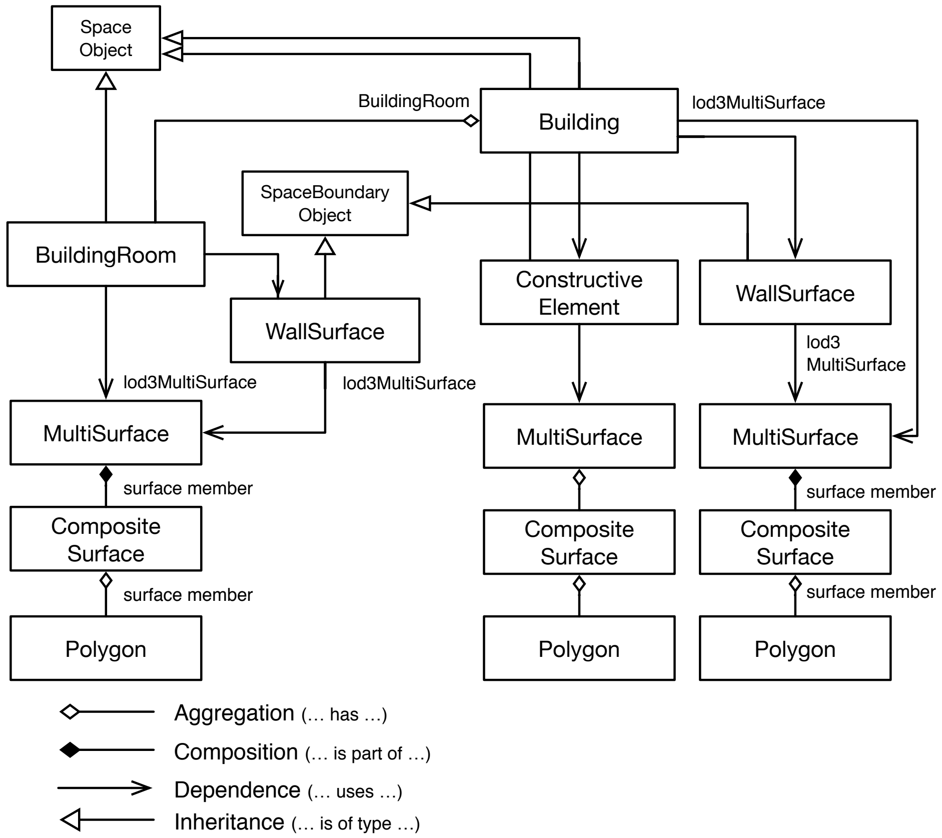


Figure 4.14: CityGML 3.0 schema (Interior Rooms)

As an alternative to the *Esri geodatabase*, the OGC provide the *geopackage* specification (OGC 2021a). The *geopackage* uses the *SQLite* database file format to provide a portable database container for storing vector and raster data. Documentation suggests that support for three-dimensional (3D) *MultiPolygons* is available, however, *FME Workbench* and *ArcGIS* appear to provide limited support for this functionality.

As well as file storage, it is common to store geospatial information in an ORDBMS. Geospatial information stored in the open-source *PostGIS* extension to *PostgreSQL* (PostGIS 2020) can be accessed using the methods described in the *Simple Feature Access*, although this standard does not support 3D surfaces and solids. Similarly, *Oracle* provides access to their *Spatial* extension through the proprietary *SDO_GEOMETRY* format.

With regard to representation using pixels, the OGC publish the *GeoTIFF* standard (OGC 2020), an extension of the regular *TIFF* raster format that geo-references the raster image by including meta-data about the origin coordinates and the Coordinate Reference System (CRS) used. 3D voxel data is used in the geological community; however, there is not as yet an accepted standard for exchanging data in this form. The regular *TIFF* format is capable of storing 3D voxel information by stacking a tower of 2D raster grid together.

4.4.3.3 Similarities and Differences

There is sufficient similarity between *Building Information Models* and *Geospatial Information Models* to enable basic information interchange from one medium to the other, albeit at the lowest common denominator. Because both information systems support the concept of a classified feature with attributed information and associated geometry, it is straightforward to extract a feature from one system, transform the syntax, and load that feature into the other system (although Chapter 7 will describe some of the practical challenges experienced with this straightforward task).

Both *Building* and *Geospatial Information Models* can be syntactically adapted for storage in ORDBMS and file-based formats. In this way, although the BIM domain favours file-based storage and GIS domain favours ORDBMS storage, interoperability issues at the *Environment* and *Platform* levels can be overcome.

The principal differences exist at the schematic and syntactic levels. The BIM domain utilises a fully-structured schema with object class inheritance and intrinsic relationships between entities. These fully structured schemas exist within both IFC and the proprietary systems, but because the models are conceptually similar, the interoperability challenges are less. The GIS domain also utilises fully-structured schema, such as CityGML and LandInfra, as well as bespoke schema developed within organisations. The semantics and structure of these schemas are sufficiently diverse, to make interchange between the BIM and GIS domains a significant challenge.

The most significant difference between *Building* and *Geospatial Information Models* is the chosen method for representing geometry. The geometric representation is chosen to suit the pragmatic needs of the industry using that geometry. For example, models in the BIM domain are created initially to support communication between designers and constructors. For this reason, BIM-based models are built using building elements such as walls and slabs, that are most easily represented as extruded 2D areas. On the other hand, GIS-based models are derived from surveys, and because only surface features are observable, the model is most easily represented in B-Rep.

Buildings are constructed on sites contained within a small footprint, enabling drawings and 3D models to be created using a local Cartesian CRS without significant errors (See Costin, et al. (2018) in Section 3.7). Geospatial information, on the other hand, covers land areas that curve around the surface of the Earth and can instead be represented using geographic coordinates expressed in degrees of latitude and longitude.

Infrastructure projects are a special case in that they are constructed over large distances but also require positional accuracy in terms of centimetres. At these scales, standard coordinate systems such as Transverse Mercator are less accurate, and errors can arise from poor scale factor. These errors can be avoided by using a bespoke *CRS* such as *London Survey Grid* (Transport for London 2011) or *Snakegrid* (Iliffe, et al. 2013).

Following on from this, because GIS-based models must be capable of representation in multiple CRS and be capable of geodetic datum transformation, GIS-based models must be represented in a Vertex-based geometric representation. Although geometric representations such as swept area may support simple transformations such as rotation, translation and scaling, they do not support complex datum transformations based on geodetic coordinate systems.

LandInfra and *InfraGML* have been developed in parallel to CityGML and IFC to serve the requirements of land administrators, civil engineers and infrastructure asset managers. However, *InfraGML* is not capable of storing detailed engineering models and it must be used in conjunction with IFC or other proprietary formats. As more than one model must be maintained, there must always be reliable methods for exchanging LandInfra information in both directions.

There are also significant differences in which proprietary BIM formats, IFC, and CityGML structure information, in particular attributes and properties. IFC elements are capable of holding information either as an attribute, as a reference to another element or by being inversely referenced by another element. CityGML, on the other hand, holds information as a sub-elements. Information from other elements can be referenced using tags. Exchange of information from IFC to CityGML requires complex schema mapping that will often be irreversible.

In summary, although syntactic heterogeneities can generally be overcome, there are significant semantic and schematic heterogeneities between *Building* and *Geospatial Information Models*. These heterogeneities arise because of the complex data relationships that exist within the schema and the choice of geometric representation, both initially chosen to support the pragmatic requirements of the model. The next subsections will discuss these pragmatic requirements concerning how the information in the models is used and managed to support the objectives of the enterprise.

4.4.4 Spatial Information Tools

The *pragmatic* level is the next level on the semiotic ladder, following on from the *empiric*, *syntactic*, *schematic* and *semantic* levels. Within the DIHAS framework, this level concerns the interface between the Human Activity System and the Information System. The pragmatic level broadly corresponds to a level in the proposed framework that was originally to be identified as *Spatial Information Processes*, however, for reasons explained in Section 4.3, this level has been split into *Spatial Information Tools* and *Spatial Information Activities*.

Practical software applications are typically composed of the *Spatial Information Platform* level and the *Spatial Information Tools* level. The two are distinguished in that the former provides access to the model, whereas the latter provides functionality including Human Computer Interface (HCI) tools and analytical algorithms.

4.4.4.1 Building Information Tools

According to the *BIM Handbook* (Eastman, et al. 2011), BIM software applications are classified as being either a *BIM Environment*, a *BIM Platform* or a *BIM Tool* (Section 4.4.1). The *Building Information Platform* application accesses information in the model stored in the *Building Information Environment*. As well as typically providing a graphical user interface for accessing model information, the *Building Information Platform* enables a place for running *BIM Tools*. Eastman, et al. (2011) considered that the functions of a *BIM Platform* should include basic tools for creating and visualising a model. Instead, according to the framework proposed here, all functions, other than digital access to the model, should be considered as *BIM Tools*.

A *BIM Tools* application includes functions for creating and editing elements, updating semantic information and attributes, and visualising the model either as a rendered scene or as plans and sections (Eastman, et al. 2011). Other tools exist for checking the model for geometric errors, in particular, performing clash detection, highlighting locations where elements stored in different files have been created in the same 3D location.

Proprietary software provides tools for the parametric design of buildings. These tools are capable of determining the layout and specification of structural columns and beams based on the parameters provided by the user; for example, tools written for the rail industry are capable of calculating clothoid curves required for gradual changes in angular acceleration around corners.

Analytical tools provide the functionality to analyse space volume topology and perform building energy analysis calculating the capacity required from Heating, Ventilation and Air Conditioning (HVAC) equipment to maintain comfortable indoor environmental conditions.

4.4.4.2 Geospatial Information Tools

The classical structure of GIS has a database at its core with an outer shell of functions that interact with that database (Bartelme 2012). The *Geospatial Information Platform* (Section 4.4.2.2) provides a stage on which the *Tools* have access to the *Geospatial Information Model* in order to perform functions on that information.

These *Geospatial Information Tools* can be broadly categorised depending on the function that they perform (Bartelme 2012). *Data Capture* tools are written to create information from inputted data such as a survey data or a remote sensing image; *Update* tools enable the user to correct geometry and provide semantic class and attribute information; *Structuring* tools convert geometry between different forms, e.g. conversion from vector to raster; *Transformation* tools convert between coordinate reference systems; *Data Handling* tools check and validate the information and provide functions to interchange information between systems; *Request and Retrieval* tools provide the ability to query information based on attributes and spatial properties; *Analytical* tools enable spatial relationships to be explored and answer questions beyond a simple query; *Visualisation and Presentation* tools interpret the information into visual form (Bartelme 2012).

Invariably, each tool cannot operate independently within its category and relies on functions provided by other tools in the toolbox to provide meaningful results. For example, a data interchange tool will use *Retrieval*, *Transformation* and *Restructuring* tools to convert from one format to another. With these tools, a GIS can read information acquired from different sources, transform and restructure that information into a consistent format so that it can be overlaid on the same platform. The visual overlay of information on a graphical display is an incredibly useful tool. However, the power of a GI System lies in the ability for generic *Geospatial Information Tools* to generate new information by combining disparate information together (Cowen 1988).

4.4.4.3 Similarities and Differences

There are many similarities between *Building* and *Geospatial Information Tools*. The first is that the types of tool follow the same pattern of categorisation. Tools exist for the creation of features, for editing information, for running queries and for visualising features.

The main difference is that the tools written for *Building Information Modelling* are limited to running on *Building Information Model* elements and vice versa for geospatial information. This difference arises because the geometry formats differ significantly between the two systems as *Building Information Models* use a variety of *Sweep Representation*, *Constructive Solid Geometry (CSG)* and *B-Rep* to model solid geometry while 3D GIS uses only B-Rep. The second difference follows on from this, in that *Geospatial Information Tools* are configured to work with generic geometric information in order to support the agnostic nature of GIS, whereas *Building Information Tools* are written explicitly for the AEC domain.

Thirdly, *Geospatial Information Tools* are predominantly written for 2D geometric features, while *Building Information Tools* are founded on the use of a 3D representation. In BIM, clash detection tools are written for analysing 3D models to detect the intersection of 3D geometries. While 3D analytical *Geospatial Information Tools* do exist, there is still limited analytical capability in comparison to 2D.

The fourth difference concerns scale and the nature of spatial relationships. BIM is generally used for small sites with linear distances rarely over 100 m, except when BIM is used for infrastructure. GIS, on the other hand, is used over much larger scales and is capable of handling geodetic geometric features which curve around the surface of the planet. Furthermore, spatial relationships in GI Science, founded on Tobler's first law of Geography (Tobler 1970), are predicated on the Euclidean distance between features. Relationships are established based on simple distance calculations. Euclidean relationships break down in urban settings whenever linear distances must follow the road network in which case spatial analytical tools must use the Manhattan distance between features. Within built assets, the Euclidean distance between features has limited value as there architectural features such as walls and floors are likely to interfere with spatial relationships. Distance between objects will instead be reliant on the navigational network through the maze of rooms, corridors and stairwells.

4.4.5 Spatial Information Activities

In the previous subsection, it was explained that a *Spatial Information Processes* level was divided into technical *Spatial Information Tools* level and a social non-technical *Spatial Information Activities* level. The *Spatial Information Activities* levels include a range of activities that support the Human Activity System (HAS) that the spatial information system has been established to support. Some of these tasks will be low-level and routine while other activities will involve negotiation between major stakeholders to approve the finalised design of a product. In the DIHAS framework these are distinguished by the terms informative acts that concern the communication of information and performative acts that concern interaction of the information system with the HAS.

It may be that some of these activities will be carried out *ad hoc* and will rely on the experience and skill of a professional person to perform a bespoke task. On other occasions, these activities will be formally documented and administered within a quality system.

4.4.5.1 Building Information Activities

Returning to the definitions of BIM set out in Section 4.2, BIM facilitates the design, construction and operation processes (according to the ISO definition), or it is the process of designing, constructing or operating (according to the PAS definition), or it is the process of using information to design, construct and operate (according to the NBIMS definition). Whichever way the semantics are interpreted, the process of designing, constructing and operating built assets are fundamental to the practice of BIM.

In order to achieve effective collaborative working, the process of designing, constructing and operating is broken down into a system of activities with the various stakeholders accepting responsibility to perform individual activities to a certain standard. A methodology for documenting these activities and the information delivery requirement at each stage in the process has been standardised in ISO 29481-1:2010 (ISO 2010). The document specifying these requirements is referred to as the Information Delivery Manual (IDM).

The *Building Information Management* process described in Section 4.4.6.1 will agree (and review) stakeholder information requirements throughout the life cycle of the built asset. Depending on the requirements of the employer (i.e. the ultimate owner), the IDM may contain process maps specifying a detailed process to be followed at every step, or it will contain transaction maps that only focus on the exchange of information between major parties. In either case, the IDM will contain a collection of formalised Employer's Information Requirements (EIRs) to be handed over at each stage of the process.

Developers use the EIRs drawn up from *Process and Transaction Maps* to write Model View Definitions (MVDs). The MVD is a formatted specification to be used by BIM applications to extract an IFC file containing only the required information from a *Building Information Model*. The MVD can also be used as a specification for validating the quality of information exchange.

4.4.5.2 Geospatial Information Activities

The term *Geospatial Information Activity* is used here to describe the tasks that make up a managed workflow. It covers a range of activities from low-level manual processing to high-level collaborative decision-making. In this context, it must be distinguished from the term geoprocessing workflow development, which is used in the GIS domain to describe the concatenation of GIS-based *Tools* to create automated routines.

Within academic literature, researchers regularly publish the workflows that have been developed to solve particular challenges (Campanaro, et al. 2016; Hjelseth and Thiis 2008). However, there are no standard methods for documenting these workflows. Li and Coleman (2005) developed a workflow model to be used in the context of GIS data production and recommended the creation of model repositories to standardise processes that encapsulate industry best practice and retain knowledge learned from previous projects. Similarly, Chai, et al. (2008) researched the use of a documented workflow for the inspection of spatial data to maintain the quality of information.

In the absence of standard workflow models, there are software solutions that implement workflow management for *Geospatial Information Activities*. For example, *Esri* includes a workflow management application that provides organisations with the ability to track the progress of tasks in a multi-user environment (ESRI 2020a).

4.4.5.3 Similarities and Differences

The documentation and supervision of *Spatial Information Activities* within a managed workflow is relevant within both Building and Geospatial Information domains. The two systems differ in that the *buildingSMART* has established an advanced system for documenting processes and the information requirements at each stage. There may be well-documented procedures for geospatial information, but these may only exist at an organisational level.

4.4.6 Spatial Information Management

In the last subsection, the *Spatial Information Activities* are described as a layer representing the pragmatic interaction between humans. In this layer, information is exchanged to complete tasks to achieve certain goals and objectives. These activities, as well as the tools, models and platforms that these activities rely on, must be tailored to meet the needs of the project that they serve. Furthermore, given that a system is required throughout the life-cycle of a project, the system must be sufficiently dynamic to respond to changes over a period measured in decades. It is in this context that the LCIM refers to the challenge of changing information requirements as the dynamic level of interoperability.

Beynon-Davies (2009) defines information management as the planning, regulation and coordination of information policies by senior management within an enterprise. Similarly Detlor (2010) describes information management as the "*control over how information is* *created, acquired, organised, stored, distributed, and used as a means of promoting, efficient and effective information access, processing, and use"*.

The role of the *Spatial Information Management* level is to understand the needs of the project or social enterprise and negotiate the most appropriate architecture of environments, platforms, models, tools and activities that are required to support those needs throughout the life of the project. The protocols for agreeing how to achieve this architecture vary between BIM and GIS and the variation is reflected by the differences between the two disciplines.

4.4.6.1 Building Information Management

Protocols for *Building Information Management* have been developed within the context of the AEC and AM/FM industries to enable the effective use of collaborative working among stakeholders. Without these protocols in place, stakeholders will develop their own procedures on how to use information within their own organisations. Bilateral protocols are established between individual stakeholders whenever they need to share information to meet a particular need.

The act of handing over of digital information inevitably leads to some information being held back. This is done for practical reasons to limit the volume information for which there is no contractual obligation to provide. However, it is reasonable to presume that if original data sources is still required, the receiving organisation must either re-obtain or re-invent the information, incurring unnecessary expense as it does so.

The UK Government has supported the development of a standardised approach to implementing *Building Information Management* protocols within the UK through the issuance of the UK BIM Mandate which came into being in 2016 (Section 3.4). As the largest customer of building and infrastructure projects in the country, the government was able to lead the development of *Building Information Management* protocols without resorting to legislation and regulation.

The specifications for *Building Information Management* in the UK have been developed by industry in conjunction with the British Standards International (BSI) and are published as the 1192 suite of standards. Organisations outside of the UK have also adopted the 1192 suite within their projects and these documents have since formed the foundation of the ISO 19650 suite of standards on *Building Information Management*.

4.4.6.2 Geospatial Information Management

Whereas the practice of *Building Information Management* has been standardised with the specific context of the AEC industry for the design and construction of built assets and the handover of building information for asset and facility management during the operational life of those built assets, the requirement to manage geospatial information is more difficult to standardise because of the range of applications for geospatial information is much higher. Nevertheless, the management of how geospatial information is created and used effectively in support of the goals of a project or enterprise is an important activity within a socio-technical spatial information system.

There are many academic publications, technical reports and textbooks offering guidance on how to identify information requirements and implement a GI System within an organisation (Somers 1998; Erie County Water Authority, et al. 1996; Tomlinson 2013). Typical guidance includes advice on delegating responsibilities, identifying information requirements and outputs, choosing information models, choosing a suitable architecture and populating the GI System with information.

Different types and sizes of organisation will have differing requirements for GIS. A large government agency using GIS to support its principal activities will be very different from that of a small local business that may use GIS to perform a single task. Consequently, an organisation must develop its own geographic information management strategy to suit its own needs (Somers 1998).

A Spatial Data Infrastructure (SDI) is a socio-technical system that is similar to a GIS. However, instead of being designed and dedicated to supporting particular objectives, the SDI is a collaborative framework of disparate information systems established to share spatial information that is agnostic to how that information will be used (CEN 2012). An

SDI reduces the duplication of spatial data collection and enable better utilisation of data and services (Grus, et al. 2010).

PD CEN/TR 15449-1:2012 defines the components of an SDI to be the "*metadata, spatial data sets and spatial data services; network services and technologies; agreements on sharing, access and use; coordination and monitoring mechanisms, processes and procedures, established, operated or made available in an interoperable manner"* (CEN 2012). From this latter definition alone, it can be seen that there is a significant role for the senior management functions of an SDI for the initial agreement on how information should be shared and accessed, followed by the ongoing coordination and monitoring of the rest of the socio-technical system.

4.4.6.3 Similarities and Differences

Building Information Management differs significantly from *Geospatial Information Management* as to the range of applications that the socio-technical system is expected to support. The *Building Information Management* protocols have been written to enable stakeholders to design, construct and handover a built asset in a commercial setting to an owner who will then manage the asset. Whereas the principles of *Geospatial Information Management*, have been formulated without a specific purpose in mind.

Within a particular project, it is normal for the *Spatial Information Management* level to govern multiple forms of information. Within the context of an information system for a built asset, *Information Management* will set the standards for both CAD-based building information and GIS-based land information as well as non-spatial asset information. It is in this context that a layperson might say that GIS is part of the BIM. It is, however, more precise to say that *Geospatial Information Environments*, *Platforms*, *Models*, *Tools* and *Activities* are vertically integrated (along with the corresponding *Building Information* levels) into a *Building Information Management* system.

Within the context of *Building Information Management*, the *Environments*, *Platforms*, *Models*, *Tools* and *Activities* required to support the design and construction of a built asset are different to the levels required to support the AM/FM activities of the same asset.

The handover of information from Capital Expenditure (CAPEX) to Operational Expenditure (OPEX) phases may, therefore, demand a transformation of information that is optimised to meet the needs of the CAPEX phase into a new format suited to the OPEX phase. This is the fundamental gap in the Crossrail Technical Information Systems as identified in Section 2.1.2.

Attention must also be drawn to a common characteristic shared by both forms of *Spatial Information Management* in that both systems require information to be managed and processed by users across a range of stakeholder organisations. It is very often the case that the gauntlet of effective *Information Management* arises not in the homogenisation of technical standards, but instead in the effective leadership of diverse individuals. The individuals concerned may not only be drawn from diverse disciplinary backgrounds but may also maintain their allegiance to competing corporate entities.

4.4.7 Spatial Information Disciplinary Field

The final outer level in this framework has been labelled as the *Spatial Information Disciplinary Field*. It is conceptually based on the outer level in the infographic used by Kennerley (2013). However, Kennerley did not accompany his infographic with any explanation as to what is meant by the *Institutional and Cultural Framework*. However, the outer rings of the Kennerley framework appear to based on a similar GIS-based framework described by Chrisman (1999). According to that framework, two outer rings belong to the socio-technical system, which are referred to as *Institutional Framework* and *Social and Cultural Framework*. The *Institutional Framework* sets goals for the GIS to achieve, whereas the *Cultural Framework* provides the worldview that motivates those goals. If Chrisman's understanding is followed, these outer rings of the frameworks represent the HAS served by the information system.

There is, however, a different way of perceiving the concept of a *Cultural Framework* than to the explanation provided by Chrisman (1999). It is proposed that this outer ring might correspond to the *People* component that is found in the classic definition of an information system (Kroenke, et al. 2013). This outer ring provides the worldview that is formed through knowledge and experience in working with spatial information modelling systems. This worldview is collectively nurtured through the evolution of a disciplinary field, that may be both professional and academic.

Writing from the perspective of being a GIS academic, Obermeyer (1994) observed the emergence of a GIS profession in 1994. Thirteen years later in 2007, she concluded that this same GIS profession was approaching maturity (Obermeyer 2007). In the former paper, she developed five criteria with which to assess the presence of a GIS profession, namely: a unique *body of knowledge*, a *professional organisation*, a *shared language*, a *professional culture and lore*, and a *code of ethics*. In addition to these, it may be prudent to include one other criterion, namely the *early career path* that practitioners need to follow to enter the discipline.

Although a functioning professional discipline is likely to fulfil the requirement for a *Cultural Framework*, it is not clear whether every criterion identified above is necessary. In the case of BIM, the socio-technical system brings together a range of formal professions namely architects, engineers, surveyors and asset managers, and it may be that an inter-disciplinary field has evolved spanning multiple professions.

In the sub-sections that follow, the five criteria will be applied to the GIS and BIM communities to explore the issues that might be relevant to a *Spatial Information Disciplinary Field* level. The sub-section headings for this topic have been intentionally switched around in this section, recognising the maturity of GIS as a disciplinary field and profession in comparison to BIM.

4.4.7.1 Geospatial Information Disciplinary Field

Body of Knowledge – The first criterion needed to recognise a profession is a *body of knowledge* (Obermeyer 1994). Although it could be argued that there is a distinction between the industrial body of knowledge and the academic body of knowledge, it can also be argued that each relies on the other in that research supports practical applications which are then in turn studied by academia.

The field of GIS is well established, the first practical GIS being pioneered in the 1960s (Longley, et al. 2005). From its inception, the computational power of GIS was quickly recognised by spatial analysts within the field of geography, and it is here that the practice of GIS found its home (Johnston 2005). Academic geographers developed the original systems, and so consequently the academic community provided an alma mater for GIS professionals.

After 30 years of solving problems related to implementing GI Systems as a practical application, Goodchild (1992) asked the GIS community asking whether the study of geographic information needed to be regarded as a scientific field in its own right, and not just a sub-discipline of Geography. A standalone field arises because spatial information that has certain inherent characteristics, namely that geographic information is keyed using a 2D spatial index, that geographic phenomena are spatially dependent, and that geographic features are located on or near the surface of an irregular ellipsoidal geoid. As such, a coordinated programme of funding was required to investigate not just the technical challenges of managing spatial information, but also data capture, conceptual modelling, statistical methods, providence of information, as well as institutional, managerial and ethical considerations. The academic community responded, and a new standalone field came to be recognised referred to as *Geographic Information Science*.

Together, the academic body of knowledge of *Geographic Information Science* and the practical body of knowledge of *Geographic Information Systems* make a substantial contribution to the establishment of a *Geospatial Information Disciplinary Field*.

Professional organisation – In addition to a body of knowledge, Obermeyer (1994) argued that the members of a profession need to associate together. The association of professionals provides several functions, although not necessarily by a single organisation. Various associations, individually and collectively, promote networking, sharing of knowledge and experience, the arrangement of research funding, marketing of skills and technologies. These associations also provide a system of certification and registration. However, this shall be discussed in the last subsection.

In the UK, the *Association of Geographic Information* (AGI) is the industry body that promotes the interests of the geospatial information community. Membership is open to organisations and individuals, and the association arranges seminars and conferences to promote networking and share recent achievements and developments.

Professional organisations and industry bodies in other countries provide similar services to promote the interests of their members and recognise the professionalism in their ranks. For example, the *Urban and Regional Information Systems Association* (URISA) provides a similar function in the US to the AGI, while the *University Consortium for Geographic Information Science* (UCGIS) advances GI Science research in American universities. Within the UK and internationally, these bodies actively promote a Geospatial Information disciplinary field and spread the body of knowledge among their members.

Shared Language – Obermeyer's third criterion evidencing the convocation of a profession is the use of a *shared language* among its members (Obermeyer 1994). While a shared language may be evidence of a profession, it may be harder to reason that it is necessary to cause a profession into being. In the same way, although a shared language is unlikely to be necessary for a *Geospatial Information Disciplinary Field*, the evolution of such a language among like-minded professionals is likely to be a natural consequence.

Within the context of the GIS profession, a shared vocabulary has arisen amongst its members through operational practice, shared research and shared education. Without getting too hung up on whether a shared language is a necessity or a consequence, the role of the profession is necessary for documenting that shared language. The profession will endeavour to recognise and clarify that terminology as it evolves through published journals, textbooks and other literature.

Further to the presence of a shared language as described by Obermeyer (1994), it is proposed that providing a system of standardisation is an essential function of a disciplinary field. Standards organisations, convened from members of the profession, agree the terminology, specifications, protocols and schema that are the backbone of the other levels in the socio-technical system.

Within the context of GIS, the OGC brings together organisations, public and commercial, to form technical committees that develop and publish technical specifications that advance geospatial location information and services. The International Standards Organisation (ISO) provide a similar function to the OGC but bring together national standards agencies rather than industrial bodies.

Professional Culture and Lore – The Obermeyer's fourth criterion for evidencing a profession is that it must have a distinctive *culture and lore* within its ranks (Obermeyer 2007). Again, a similar question to the one in the previous subsection can be asked as to whether a professional culture is a consequence or a requirement for a profession. Whereas it is expected that *culture and lore* will evolve just as a shared language evolves among a society of like-minded professionals, it could also be argued that the *culture and lore* are necessary to maintain a disciplinary field. Inducted into a profession, members will embark with missionary zeal to convince others as to the benefits of their professional field. Furthermore, Obermeyer (1994) identified that the principal players within a profession are recognised as such and are elevated above their peers to a *hall of fame*.

Obermeyer (1994) argued that the GIS professional community has a distinctive culture as described in the previous paragraph. Furthermore, geospatial professionals cultivate a community among their members, recognising principals such as Roger Tomlinson, Jack Dangermond, and Mike Goodchild, as icons for inclusion in the GIS pantheon.

Code of Ethics – All of this builds up to the last of Obermeyer's criterion requiring a profession to hold its members to account. In the UK this is achieved through chartership either through the Royal Geographic Society (RGS) or the British Computer Society (BCS). As well as being able to demonstrate a framework of competencies, chartered members must abide by the code of conduct of the respective society. Members maintain their chartered status through participating in Continuing Professional Development (CPD) activities and renewing their registration every year. There is a gap in that although a chartership enhances an individual's employability, it is not normal for employers to specify it as a condition of employment and GIS is not a regulated profession in the same way as medicine and law. As there is nothing unusual about an unchartered GIS practitioner, it is almost impossible to sanction unethical behaviour.

In the US, a similar system is run by the GIS Certification Institute (GIS Certification Institute 2020). Having proved their experience and competence, members agree to be bound by a code of ethics and rules of conduct. Members maintain their certification through CPD activities and renewing their certification, but as with the UK, there is a no legal requirement for certification.

Early Career Path – The early career path into geospatial information in the UK generally requires completing a period of postgraduate study followed by experience working as a GIS Technician, Analyst or Developer, before taking on a managerial position (Prospects 2020). The early career path is important not just for educating entrants with the knowledge and skills required to participate in the disciplinary field but also introducing them to the professional organisations, culture and ethics.

All in all, Obermeyer's criteria provide a convenient starting point for describing the range of functions provided by the *Geospatial Information Disciplinary Field*. The *Disciplinary Field* has a strong academic base and a community of professionals who hold the rest of the socio-technical system together.

4.4.7.2 Building Information Disciplinary Field

In this subsection, the same criteria developed by Obermeyer that were applied to the *Geospatial Information Disciplinary Field* shall be applied to the *Building Information Disciplinary Field*. Because the building industry was late to switch from unintelligent CAD to intelligent spatial information, the *Building Information Disciplinary Field* is less mature than its geospatial cousin.

Body of Knowledge – Although Building Information Modelling was not titled as such until the early 2000s, the principles behind the technology have been in development since 1975 (Eastman 1975). The next decade saw steady development leading up to the publication of a General AEC Reference Model in 1988 (Gielingh 1988). Although the benefits of BIM were evident as a future technology, industry did not show particular interest in BIM until 2005 (Santos, et al. 2017). Once the major software publishers were able to market Building Information Models within their own proprietary software, research and development into how to implement the technology into the project were undertaken by industry.

Since 2010 there has been heightened interest in the construction industry with the major software publishers and engineering consultancies organising trade conferences on BIM. During this time, conference papers on BIM have been presented at a more general academic conferences, but the past five years have seen the organisation of conferences dedicated to the field of BIM.

Academics have also published numerous articles on BIM in peer-reviewed journals such as *Automation in Construction*, *Advanced Engineering Informatics*, and *Journal of Computing in Civil Engineering* (Santos, et al. 2017). From the range of published academic and industry literature, it evident that there is a growing body of knowledge accessible within the *Building Information Disciplinary Field*.

Professional organisation – In 2011, there were still commercial bottlenecks preventing widespread adoption of BIM in the UK. In response, the UK Government mandated that all government construction contracts must be delivered using BIM. In the absence of a standard requirement for Building Information, the UK Government sponsored the formation of the UK BIM Task Group to research and implement a suite of standards and protocols in support of the mandate. Having completed this directive, the task group has reinvented itself as the UK BIM Alliance with a mission to represent the interests of the UK BIM organisations. There is, however, no single organisation that exists to represent individual members of the BIM community.

Concerning research funding, the *Centre for Digital Built Britain* was established as a partnership between the UK Government and the University of Cambridge to provide oversight of an integrated UK BIM programme. The Centre is charged with ensuring that research findings inform the national strategy.

Shared Language – Bearing in mind the caveats explained in Section 4.4.7.1 regarding whether shared language is a consequence or a requirement of a profession, the field of BIM has its own distinctive language that is used among its practitioners. In the UK, references to *PAS 1192* and *BIM Level 2* are commonly understood. National dialects do exist, however, as the disciplinary field matures, this language is being harmonised.

With regards to international standardisation, work is currently in progress by ISO to interpret the standards published in the BS 1192 suite as the ISO 19650 suite. Meanwhile, buildingSMART, formerly the *International Alliance for Interoperability*, was formed in 1995 to bring together stakeholder organisations in the construction industry to develop standards classes for interoperable exchange of information between software applications.

Professional Culture and Lore – Whereas there is no doubt that the BIM community has an underlying professional culture, it is beyond the scope of this thesis to gather the necessary supporting evidence. Emergence of this professional culture can be found on the BIM+ website (BIMplus 2021). The website, owned by the Chartered Institute of Building is an enthusiastic source of news, articles and opinions that aims to inform and galvanise the BIM community. With regards to a *Hall of Fame*, the author would include iconic personalities such as Charles Eastman and Mervyn Richards into such a position.

Code of Ethics – In the UK, there is evidence of an emerging BIM profession, but it is focussed on one particular role, the BIM Manager, who has responsibility for managing projects using BIM methodology. The *Royal Institution of Chartered Surveyors*(RICS) provides a system for certification for the BIM Manager role, along with rules of conduct for RICS members. However, membership is not necessary for certification, and the certificated practitioner may be a member of another professional body. There is, therefore, no single BIM Code of Ethics.

Although a defined role exists for the professional BIM Manager, there are many other people engaged in the discipline, such as technicians, analysts, asset managers, and developers who have no formal professional body to join.

Early Career Path – The BIM disciplinary field is still very young, and therefore the practitioners that make up the community will have embarked on a range of career paths to reach their current positions. A range of professional training courses and academic qualifications are being provided such as diploma-level training to be a BIM technician (University of Derby 2020) or studying at Masters-level in support of career development (University College London 2020). However, there is no agreed core syllabus on which to base training and education.

Using Obermeyer's criteria, the *Building Information Disciplinary Field* appears to be a loose confederation of practitioners who are domiciled in a range of other professions, such as architecture, surveying and asset management. Although the *UK BIM Alliance* has been formed to represent the organisational stakeholders, there is no professional body to which individual practitioners can belong. However, through training and other means of professional development, there is evidence that a disciplinary field is emerging.

The professional role of BIM Manager has become recognised in the AEC profession working in the design and construction phase of the project (Institution of Civil Engineers 2020). However, it should be noted that a BIM Manager is a specialist position that exists to manage the common data environment and support other engineers who make management decisions concerning design and construction. It is the role of every professional to understand the tenets of Building Information Modelling and their part in contributing to its success.

4.4.7.3 Similarities and Differences

Having reviewed the disciplinary fields for both BIM and GIS using Obermeyer's criteria for evidence of profession, there are significant similarities and differences. The first observation is that the *Geospatial Information Disciplinary Field* is more established than the *Building Information Disciplinary Field*. Although conceptual foundations of building information only followed geospatial information by about ten years, geospatial information technologies were quickly adopted by governmental organisations responsible for land management. A professional caste of geospatial information scientists heeded the call to undertake the academic research and commercial development required to solve new challenges. Although the academic research and commercial development were undertaken in building information, the catalyst to deliver the technology to market was perhaps 30 years behind geospatial information.

Although both BIM-based systems and GIS-based systems provide practical tools for the management of spatial information, it should be observed that geospatial information is more amenable to spatial analysis. Because of this, a second observation is that the *Geospatial Information Disciplinary Field* tends to include a contingent of professionals rooted in academic science and the humanities in addition to those grounded in engineering.

A third observation is that the *Building Information Disciplinary Field* has its foundation in the AEC industry and, to a lesser extent, the AM/FM industry. The discipline is dominated by and is subservient to the requirements of those industries. In contrast, the *Geospatial Information Disciplinary Field* has branched out from its origin in land management and now serves a range of industries and governmental organisations, for example, demographics, epidemiology, geology and oceanography.

Fourthly, the geospatial information community appears to have a more substantial and more identifiable core profession in comparison to the *Building Information Disciplinary Field*. The latter is comprised of a dedicated inner circle of BIM professionals and a loosely associated outer circle drawn from a range of multi-disciplinary professions, including the GIS profession. This characteristic, combined with the previous observation, has the potential to cause the two professions to merge.

There are, however, many similarities between the two disciplinary fields. These similarities can be predominantly observed in an overlap in the body of knowledge shared between building information and geospatial information in areas such as computer science, information modelling and information management. There is an opportunity for professionals to learn from each other and even navigate the turbulent waters that exist at their confluence. If the nature of the BIM profession becomes less focussed on project and asset management and more focussed on spatial information system implementation, analysis and simulation, then it is only natural for BIM and GIS professionals to crossover.

It should be noted that there is still some debate with regards to the future of the BIM profession. There is a view that BIM will become business as usual in the AEC and Asset Management (AM)/Facilities Management (FM) communities with all architects, civil engineers and asset managers becoming "*BIM professionals*" (Geospatial World 2016). On the other hand, as the adoption of BIM grows, the implementation and maintenance of effective and reliable BIM systems will become a critical business function. As people enter the profession who have a passion for not just managing projects, but also for conducting simulation and analysis, then there may still be a need for a dedicated cadre of spatial information specialists.

In this section, the similarities and differences of BIM and GIS have been studied at each of the seven levels in the proposed Spatial Information System Framework. The foundation has now been laid to consider how the Framework can be applied to existing examples of interoperability and integration between BIM and GIS in literature.

4.5 Applying the Framework to BIM/GIS Interoperability

In Section 4.3, a framework of seven levels was proposed describing the constitution of a spatial information system. Earlier proponents of information system frameworks, Chrisman (1999) and then Kennerley (2013), illustrated their frameworks using a series of concentric circles as illustrated in Figure 4.2. The illustrations recognise a concentric relationship between each level, whereby the lower levels are constrained by the requirements of the higher levels, meanwhile, the efficacy of the higher levels is dependent on the most appropriate implementation of the lower levels. Adopting a similar approach, the proposed framework of seven levels is illustrated at Figure 4.15 for both the building and geospatial domains using the harmonised terminology.

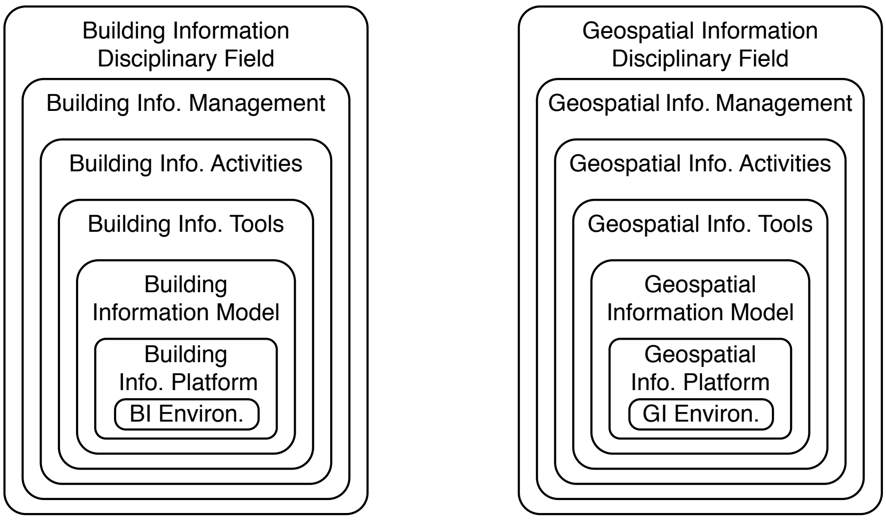


Figure 4.15: Framework applied to building and geospatial domains

In Section 3.6 of the literature review, it was revealed that some authors had developed frameworks for categorising research into BIM/GIS interoperability. Work was done to align the levels of these frameworks and consolidated them together into a new framework consisting of three principal levels, that are the *Data* level, the *Process* level and the *Application* level. The framework is illustrated in Figure 3.18 with published research articles providing examples of how BIM/GIS interoperability is approached at each sub-level.

Each of the published research articles Figure 3.18 will be used to test the framework proposed in this chapter as a means of describing how BIM/GIS interoperability is approached in that publication. This test will be achieved by marking up the diagram of concentric circles in Figure 4.15 to illustrate the level at which interoperability is being achieved.

4.5.1 Data Level

4.5.1.1 Information Integration

The first example illustrated in Figure 4.16 is exemplified by the research published by Hijazi, et al. (2010) concerning the mapping of IFC utility network information into an ADE in CityGML. Specialised information is directly imported into CityGML format, and there is limited interaction with the core schema of the CityGML model.

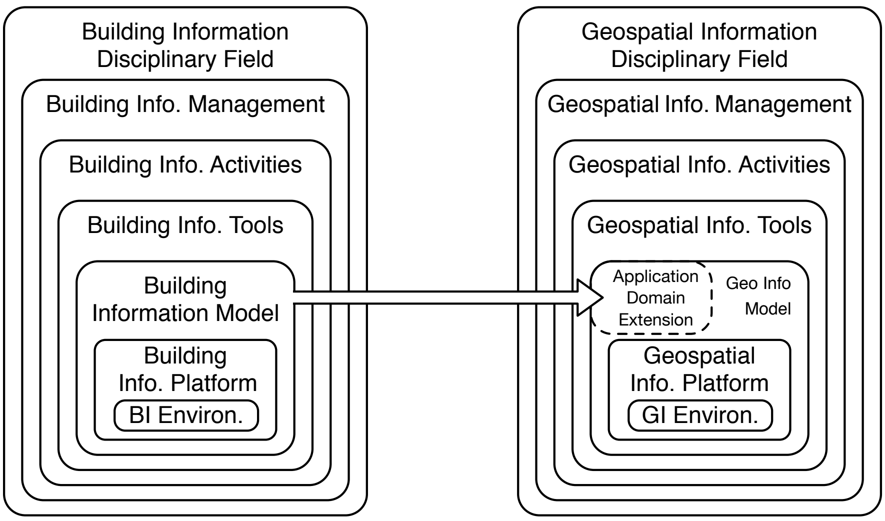


Figure 4.16: Integration of information at Data level

4.5.1.2 Information Conversion

The next example concerns the conversion of IFC geometric and semantic information into the CityGML schema as researched by Donkers, et al. (2015) as illustrated in Figure 4.17. Nagel, et al. (2009) provides a similar example concerning the conversion in the opposite direction of CityGML geometry into IFC.

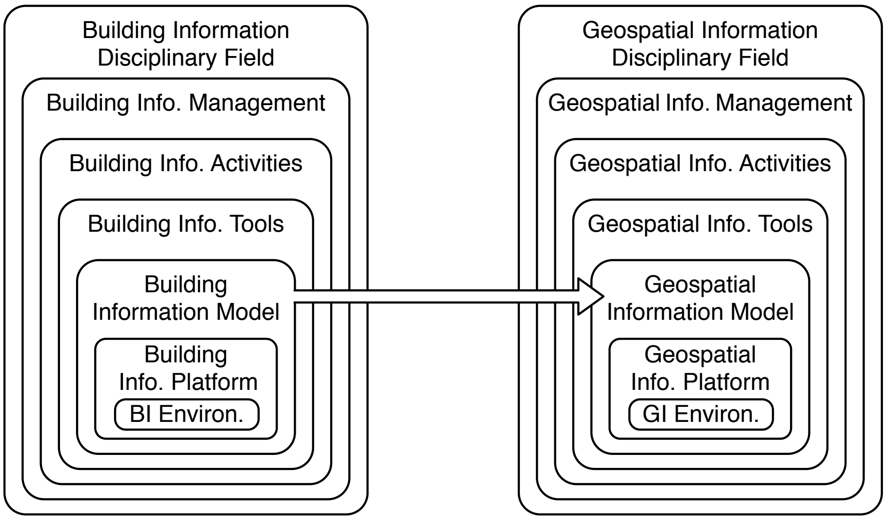


Figure 4.17: Conversion of information at Data level

4.5.1.3 Unified Information Model

The third approach illustrated in Figure 4.18 is the development of a *Unified Building Model* as advocated by El-Mekawy, et al. (2012). A *Unified Building Model* could not be hosted on either a BIM or GIS platform, and therefore a dedicated platform would be required to hold the model. In this way, a *Unified Information Model* (using neutral terminology) would hold all the necessary information to extract information in a form accessible to both *Building* and *Geospatial Information Tools*.

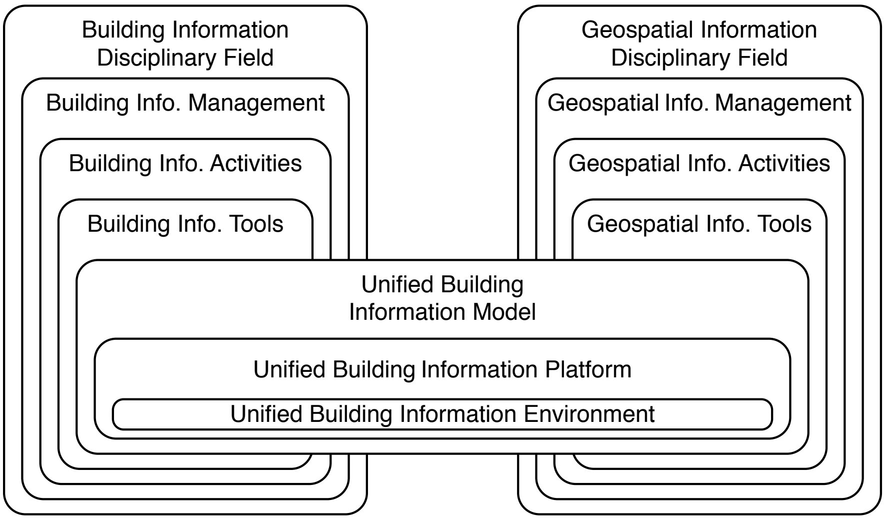


Figure 4.18: Unified Information Model at Data level

4.5.1.4 Data Mapping

Kang and Hong (2015) identified a final form of system interoperability referred to as *Data Mapping*. They did not find a specific example relevant to BIM/GIS interoperability although similar research was found in the field of product modelling (Scherer 2007). This form of interoperability, as illustrated in Figure 4.19, concerns the exchange of information from one system to another, the processing of information in that other system, and then the full incorporation of that processed information in the original system.

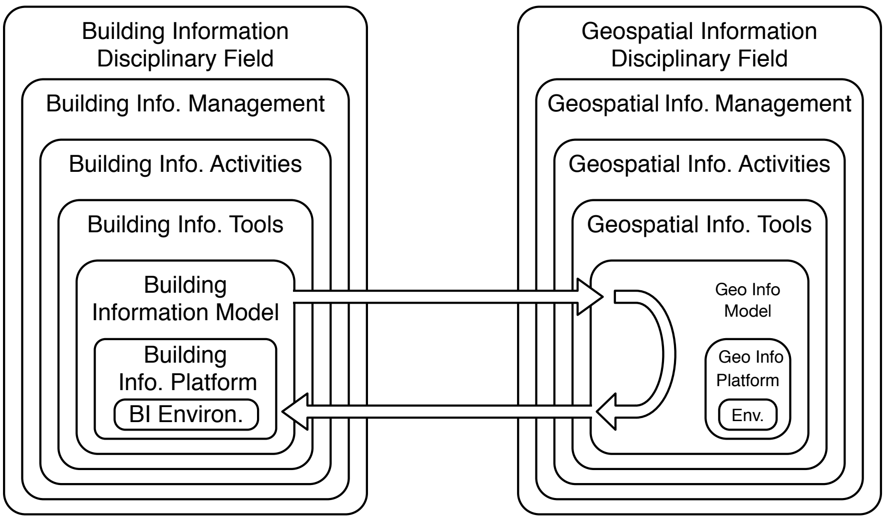


Figure 4.19: Data mapping at Data level

4.5.2 Process Level

4.5.2.1 Visualisation Tools

Döllner and Hagedorn (2007) published their work into visualising geometry sourced from both *Building Information Models* and *Geospatial Information Models* in a web browser. This work is classified as belonging to the process level because the information is not permanently changed at the data level in the *Spatial Information Models*. This loose integration is illustrated in Figure 4.20.

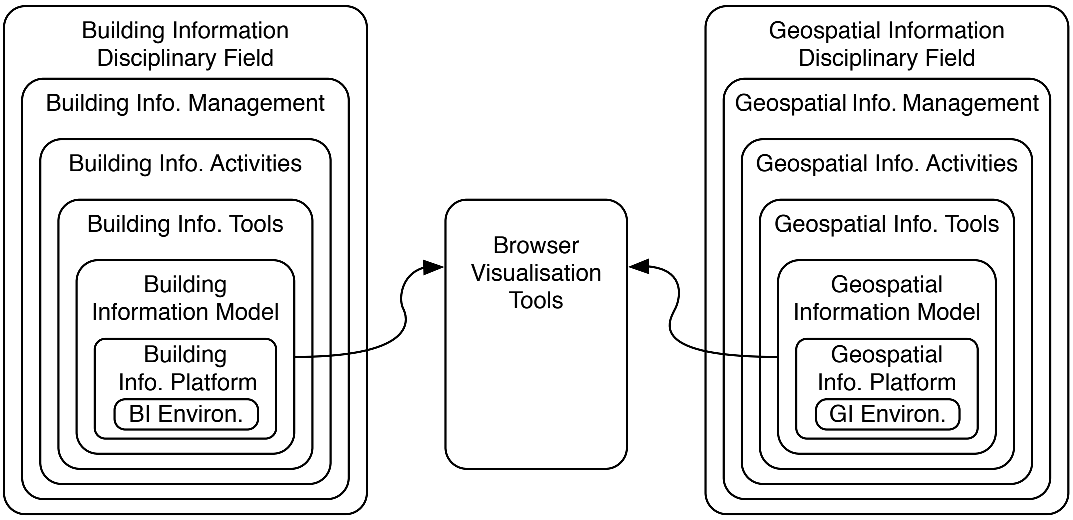


Figure 4.20: Visualisation tools at Process level

4.5.2.2 Ontological Tools

Another example of interoperability at the *Process* level, as illustrated in Figure 4.21, is provided by the work of Beetz, et al. (2006) who proposed a topological reasoning service for extracting data in the form of RDF triples suitable for use in semantic web applications. A similar approach to accessing data via RDF was exploited by Mignard and Nicolle (2014) in the development of an integrated tool for managing AM/FM data as explained in Section 4.5.3.2.

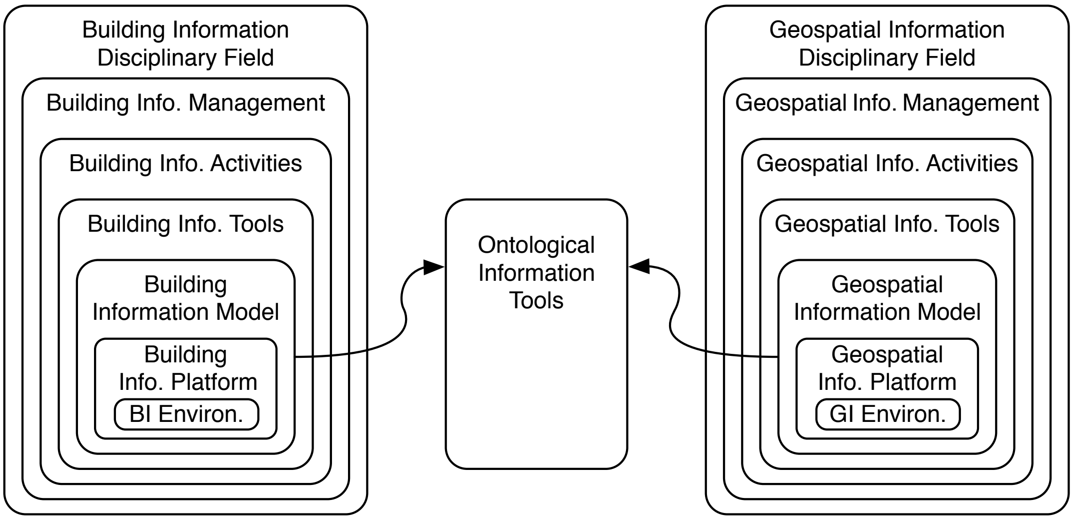


Figure 4.21: Ontological tools at Process level

4.5.3 Application Level

4.5.3.1 Spatial Data Warehouse

The first illustration at the *Application* level (Figure 4.22) is the Spatial Data Warehouse (SDW) as a means of tight integration as published by Kang and Hong (2015). In this diagram, the *Activities*, *Management* and *Disciplinary Field* circles have been drawn around the SDW to recognise that the purpose of the SDW in Kang and Hong (2015) is to support FM activities as governed by *Building Information Management* requirements.

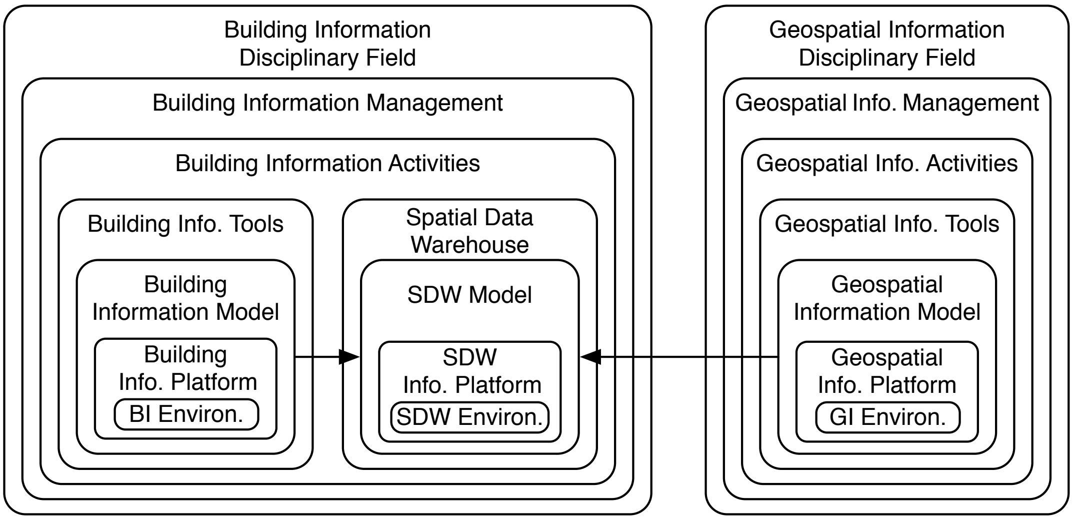


Figure 4.22: Spatial Data Warehouse at Application level

4.5.3.2 Integrated Information Application

The integrated information application, as illustrated in Figure 4.23, utilises the ontological tools, as explained in Section 4.5.2.2. Mignard and Nicolle (2014) used such a system to develop an integrated application for accessing FM information within a built asset.

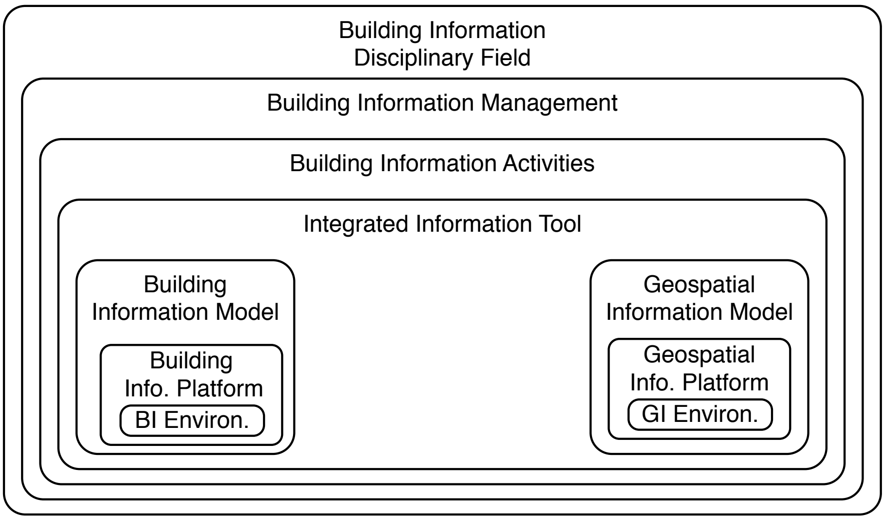


Figure 4.23: Integrated Information Application at Application level

4.5.3.3 Integrated Information Activities

The work published by Schaller, et al. (2017) is an example of how engineers use *Building* and *Geospatial Information Tools* together within the activities required to plan the route of a highway. In Figure 4.24, the *Geospatial Information Model* is included within the envelope of *Building Information Management* to illustrate the example of how GIS has been vertically integrated into BIM.

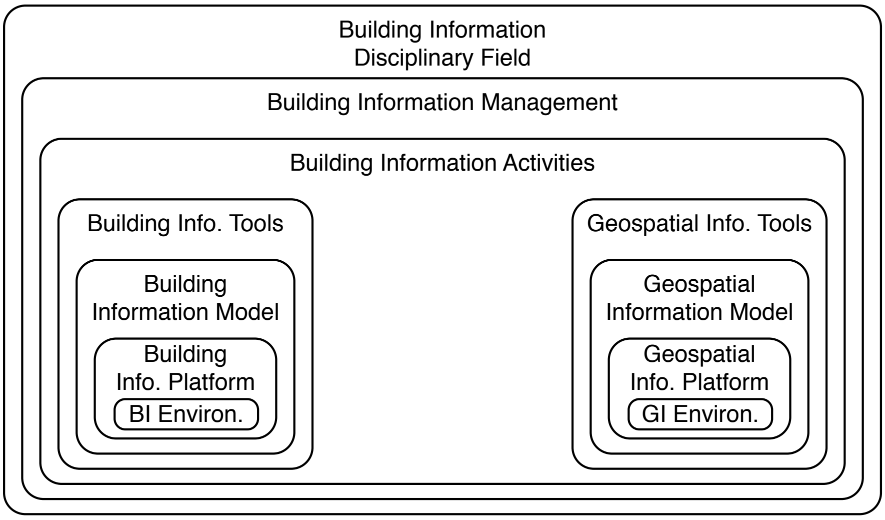


Figure 4.24: Integrated Information Activities at Application level

The first sections of this chapter demonstrated a requirement for an agnostic Spatial Information System Framework that could be used to describe the various levels of BIM and GIS. Using existing frameworks as a foundation, a novel framework was developed and used to describe the full socio-technical range of BIM and GIS systems and analyse various integration methods described in literature. The framework is now ready to be applied to the Technical Information Systems used in the Crossrail project.

5 Application of Framework to Crossrail Systems

In Chapter 4, a novel Spatial Information Systems Framework was developed that provides a means for describing the decomposition of both Building Information Modelling (BIM) and Geographic Information Systems (GIS) while using the same terminology. Within the Crossrail project, the Computer Aided Design (CAD) system, Geographic Information System (GI System) Asset Information Management System (AIMS) and other information systems under the management of the Technical Information Group are referred to as the Technical Information Systems. In this chapter, the seven levels of this framework will be applied to the Crossrail Technical Information Systems in order to describe the socio-technical nature of each system. Where not cited otherwise, the information in this chapter has been gathered from directly working with the systems while attending on-site at the Crossrail head office and through discussions with the managers and staff within the Crossrail Technical Information Department, under the supervision of the GIS Manager, Daniel Irwin, the industrial supervisor of this doctoral research [1].

Having broken each system up into a series of levels in Section 5.1, Section 5.2 aims to compare the systems at each level and identify the different types of system heterogeneity. As explained in Section 3.3.1, system heterogeneity is the prime cause of interoperability affecting the systems on different levels.

The practical application of the Spatial Information Systems Framework to the Crossrail Technical Information Systems will be used as a further method for validating the framework, which will be in addition to work done applying the Framework to examples

[1] As part of the studentship programme, Crossrail provided the author with an office pass, a dedicated workstation and network access to the Technical Information Systems. This provided the opportunity for regular attendance approximately one day per week over a three year period.

found in literature as described in Section 4.5. The suitability of the Framework in this regard will be discussed in Chapter 10.

5.1 Crossrail Technical Information Systems

As part of its commitment to technological innovation (Section 1.2), Crossrail has had a fundamental vision to build a virtual railway in parallel with the physical railway. This virtual railway would enable collaborative methods of working throughout the delivery phase of the project. This vision was formulated at the initial stages of the project once parliamentary approval had been granted.

During the planning stage prior to parliamentary approval, GIS was used by Crossrail planners to optimise the route, assess the impact of the route and address parliamentary concerns. Alongside this, the planners mocked up three-dimensional (3D) CAD models to provide Parliament with a conceptual visualisation of the route and stations. These 3D models were disjointed and not integrated with the GIS. Once approval was given to start the engineering design, the importance of integrating these 3D CAD models with GIS plans was recognised, and all models were converted into *Bentley MicroStation DGN* (the standard format used by Transport for London (TfL)) and were geo-referenced using London Survey Grid as a common Coordinate Reference System (CRS) (Taylor 2017).

At about the same time as the start of the Crossrail design phase, BS 1192:2007, a national code of practice on collaborative working (BSI 2007) was in the final stages of publication. Being committed to innovation, Crossrail co-operated with the British Standards International (BSI) authors and pioneered the implementation of this new standard. Incorporating the principles of BS 1192:2007, a Common Data Environment (CDE) was established as a single repository for storing all CAD models and asset information, regardless of which stakeholder had ownership of the information. In time this CDE would become the foundation of the enterprise management system used for managing all contractual information requests and design approval.

Before BS 1192:2007, it had been standard practice for the design contractor to lead the project in choosing the configuration of the information systems and to host the hardware and software for managing the information model. Although beneficial and profitable for the design contractor, the client had the burden of working with a multitude of standards forced upon them by multiple contractors. With the Crossrail CDE, the central Crossrail organisation took on the responsibility for establishing common information standards and hosting the network architecture to be accessed by all contractors.

The CDE hosts all structured and unstructured information (ISO 2018). Structured information includes CAD models and Relational Database Management System (RDBMS) data, while unstructured information includes documents and photographs. Ideally, an integrated CDE should be established, capable of handling all types of information. However, the practical implementation of a CDE requires the use of different commercially available software applications suited to different types of information.

The Crossrail CDE is built upon four specialised technical information systems namely the Electronic CAD Management System (ECMS) (*ProjectWise*), the Electronic Document Management System (EDMS) (*AssetWise/eB*), the GIS and the Master Data Model (MDM). The implementation and utilisation of these technical information systems are key to the Crossrail BIM strategy.

5.1.1 Crossrail Spatial Information Environment

The first level in the Spatial Information System framework is the *Spatial Information Environment*. This level provides the basis for one of the primary objectives of BIM, that is to promote collaborative working among stakeholders. The *Spatial Information Environment* consists of the CDEs through which all Crossrail staff, framework design consultants, primary contractors and infrastructure managers have access to the information models. Configuration of the CDE involves designing the network architecture with suitable bandwidth and providing the security protocols for the appropriate people to access information with the appropriate privilege.

The information in the CDE is managed following the principles set out in BS 1192:2007 (BSI 2007) to ensure that only information authorised for access can be retrieved and edited by the appropriate stakeholders (Irwin and Tamash 2016). The Crossrail *Spatial Information Environment* is hosted on servers managed by Crossrail and is made up of four separate environments: (1) *ProjectWise*, (2) an *Oracle* Object Relational Database Management System (ORDBMS), (3) *AssetWise/Enterprise Bridge* and (4) the *Master Data Model Environment*.

5.1.1.1 ProjectWise

Access to all CAD drawings and models that make up the Crossrail Project Information Model (PIM) is managed by *Bentley ProjectWise*, an ECMS file management system tailored to the management workflows specified by BS 1192:2007 (Crossrail 2013). Using ProjectWise ensures that CAD models are checked out before editing and back in afterwards preventing version clash.

Metadata on each CAD file is maintained in an *Oracle* database that references the location of the file saved in the networked file operating system. Every CAD file created by design contractors is saved using a concatenated filename structured in the format illustrated in Figure 5.1 (Crossrail 2016a).

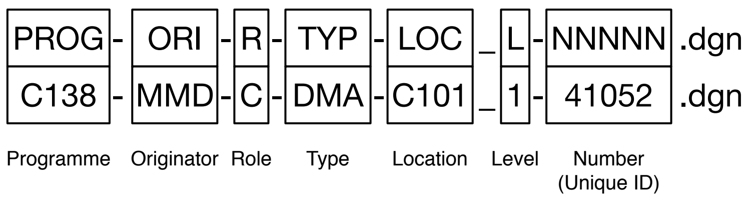


Figure 5.1: Crossrail filename format (Crossrail 2016a)

5.1.1.2 Geospatial Information Environment

Geospatial information in Crossrail is centrally stored in an ORDBMS environment running in *Oracle*. The architecture of this ORDBMS environment is depicted in the diagram at Figure 5.2 (Irwin and Tamash 2016). Crossrail use Oracle technology because of its robustness and resilience when storing large datasets of critical information and because of level of support provided to maintain the database and resolve issues.

The predominant method for *Geospatial Information Tools* to access the information in the *Geospatial Information Environment* is through the *Crossrail Maps* application which serves live data over the Crossrail intranet for users to view the information in a client browser (Irwin and Tamash 2016). The information can also be viewed by Crossrail users in mobile applications accessed over the internet, and a limited amount of information can be accessed on the public website. To maintain the security of the Crossrail network, these users access a mirror server that is regularly updated through a firewall (Irwin and Tamash 2016).

As with the information in the CAD CDE, the *Geospatial Information Environment* is managed following the principles set out in BS 1192:2007 (BSI 2007) to maintain the integrity of the information (Irwin and Tamash 2016).

In addition to the central geospatial CDE, Crossrail maintained ancillary data environments for particular business tasks, such as a dedicated registry for storing information relating to legal ownership and rights over property and land. These data environments are maintained outside of the CDE and are not governed by the Crossrail BIM protocols.

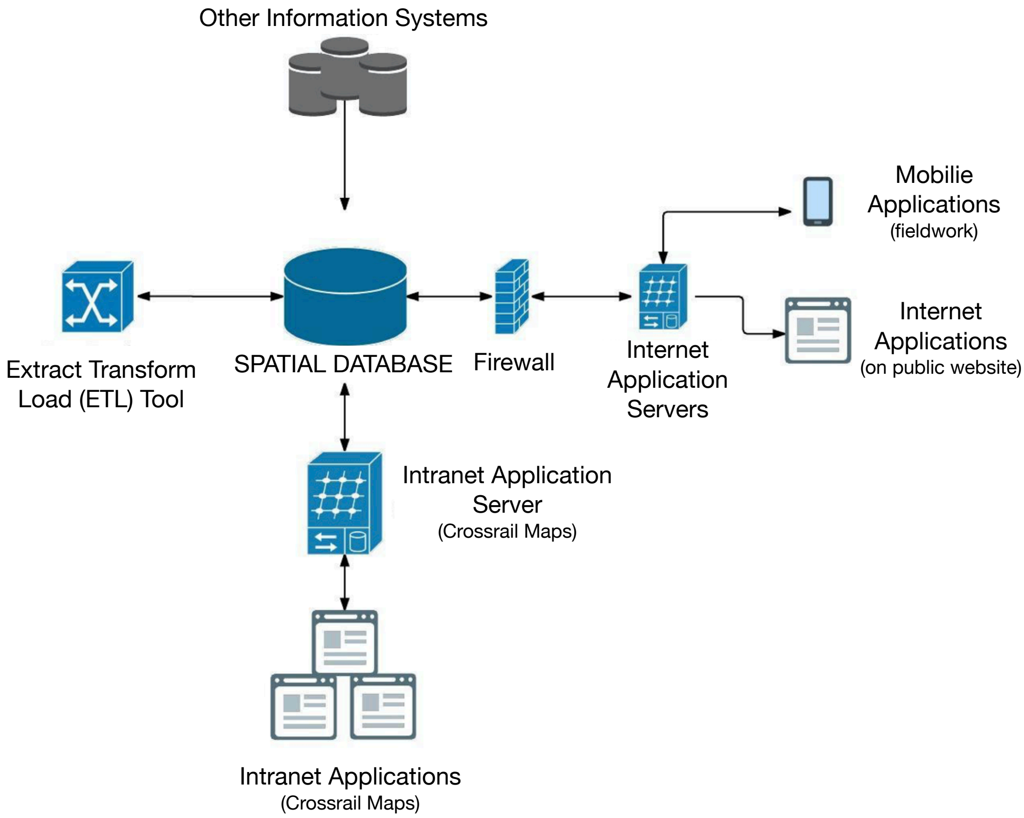


Figure 5.2: Crossrail Geospatial Information Environment (Irwin and Tamash 2016)

5.1.1.3 AssetWise / Enterprise Bridge

Information on Crossrail assets and associated documentation is stored in *AssetWise*, a database product developed by *Bentley*, hosted on a *SQLServer* database. *AssetWise* is hosted on Crossrail managed servers and accessed through a browser-based client. It is moot as to whether *AssetWise* can be classified as a Spatial Information System; on the one hand, it does not contain geometric information, while on the other, assets are broadly positioned using named location (i.e. it is not certain where the asset is except that is contained within a geometric boundary). The levels of the Spatial Information System are applied to *AssetWise* in order to compare it with the other Crossrail Technical Information Systems.

The *AssetWise Spatial Information Environment* was originally procured to be a register of assets with the ability to cross-reference associated documentation. However, as the

Crossrail project needed a general-purpose EDMS, it was decided to extend the scope of *AssetWise* rather than procure another information system. In this guise, *AssetWise* is also referred to as eB (Taylor 2017). eB is used to manage standardised information workflows, such as the CAD data team drawing request (Figure 5.3), allowing transmittal of information for review by designated persons and enable requests to be authorised following documented procedures.

5.1.1.4 Master Data Model Environment

The *Master Data Model* and the *Master Data Management Data Warehouse*, which shall be described in further detail in Section 5.1.3.4, are hosted on an RDBMS running on *Microsoft SQL Server* (Palmgren 2017).

5.1.2 Crossrail Spatial Information Platforms

A *Spatial Information Platform* is the software application that enables access to the *Spatial Information Model* stored in the *Spatial Information Environment*. In the case of information stored in an ORDBMS, there is a nuanced distinction between the *Environment* and *Platform* levels.

5.1.2.1 Bentley MicroStation

Crossrail uses *Bentley MicroStation* as the *Spatial Information Platform* to create and edit 2D CAD drawings and 3D models. The *Spatial Information Platform* is extended with the *Bentley* BIM extensions, namely *Bentley Architecture*, *Bentley Structural Modeller*, *Bentley Building Electrical Systems* and *Bentley Building Mechanical Systems*.

Bentley MicroStation is a popular CAD platform that is traditionally preferred by the rail industry. The choice of *MicroStation* was driven by the fact that it is already used extensively by London Underground and Network Rail.

5.1.2.2 Oracle Spatial

The *Spatial extension* to the *Oracle* ORDBMS environment provides a platform for spatial tools to access the model. These tools perform spatial operations such as buffering and spatial queries such as intersections.

5.1.2.3 ArcGIS

In addition to the *Oracle Spatial*, the Crossrail GIS team also use the *Esri ArcGIS Spatial Information Platform*, installed as the *ArcMap* and *ArcScene* applications, to access geospatial information held within the ORDBS. *ArcMap* provides a Graphical User Interface (GUI) platform with visualisation tools for viewing two-dimensional (2D) information, while *ArcScene* provides a GUI platform with the visualisation tools for viewing two-and-a-half-dimensional (2.5D) and 3D information.

5.1.2.4 Master Data Model Platform

A custom front-end application was developed in-house for Crossrail staff and contractors to access the *Master Data Model* (Palmgren 2017).

5.1.3 Crossrail Spatial Information Model

5.1.3.1 CAD Information Models

All Crossrail contractors are contractually required to deliver their digital output according to the Crossrail CAD standards (Crossrail 2016a). According to this standard, all CAD drawings and 3D Models shall be created and edited using *Bentley MicroStation* and the *Bentley* BIM extensions. Any contractor or sub-contractor using another standard shall be responsible for converting work into *Bentley MicroStation* and accepting any risk associated with format conversion.

Detailed engineering information within the Crossrail 3D model is held in *MicroStation* CAD files (as identified by the *DGN* file extension). The information in these files contains general geometry that can be read by the underlying *MicroStation* platform but also contains information that can only be created, edited and visualised if the specific BIM extension is installed (Crossrail 2016a).

The Crossrail 3D model is constituted as a federated model. A distinct file is created to contain elements according to the contractor tasked with its creation, the site, the storey level, and the *BIM* extension used. Coordination files are created that pull information together by referencing the individual files in the federated model (Crossrail 2016a).

Every element in the file is held within a *MicroStation* level (akin to the term *layer* used in *AutoCAD*). The levels in the Crossrail CAD model are established in the Crossrail seed file that provides a template as *DGN* files are created (Crossrail 2016a). These levels are structured using a scheme of classification modified from *Uniclass 1.4* (CPIC 2014) providing a universal method for grouping elements according to their class.

Every element created in *MicroStation*, whether created using the basic *MicroStation* tools or the BIM authoring extensions, is attributed with an persistent *Element ID* that is unique within its model file. The *Element ID* together with the file name uniquely identify elements within the project. As such, this *Element ID* can be used to trace elements and maintain an inventory of elements.

Within the BIM authoring extensions, elements are created using a system of *families* and *parts* (Bentley Systems 2020). As well as providing semantic classification, these *families* and *parts* are used to group property sets.

There is a range of geometry types available in *MicroStation*. Simple geometry types include *PointString*, *LineString* and *Surface*. The geometries of the elements created using the BIM authoring extensions typically belong to a proprietary geometry type referred to as *SmartSolid* that uses Constructive Solid Geometry (CSG) to define solid geometry.

As well as deciding on a standard CAD format, a single coordinate reference system was adopted. Early planning work had used British National Grid (BNG), however, the use of the national CRS is not suitable for civil engineering work across London due to unacceptable scale factor distortion. Instead, a bespoke CRS tailored for use in central London was developed and published as *London Survey Grid* (Transport for London 2011). The production of all 3D models in a single CRS and a single CAD format enables two or more geometric models to be loaded alongside each other. Produced in the form, the individual standalone models come together to make up the federated model.

5.1.3.2 Geospatial Information Models

Geospatial information in Crossrail is stored centrally on an *Oracle Spatial* ORDBMS. All information is primarily held as 2D information (i.e. without height information) as *points*, *linestrings* or *polygons* in a format compliant with the Open Geospatial Consortium (OGC) *Simple Feature Access* (Figure 4.11).

Information is stored as layers, for which there are over 700. In order to manage the content of these layers, the UK *Gemini* standard is used for attaching metadata to each data layer captured (Irwin and Tamash 2016). The data in these layers may be information created internally by Crossrail such as route information, legal boundaries and logistical information, or it may be externally sourced information, such as utility information and OS MasterMap base map, provided centrally under a collective licence agreement (Irwin and Tamash 2016).

As part of Crossrail's commitment to innovation, the GIS team is evaluating the use of 3D representations (i.e. surface and solids). This information is predominantly stored as *Esri MultiPatch* geometries saved on the *Oracle* database using *ArcSDE* technology.

5.1.3.3 Asset Information Management System Model

Although a wealth of asset information exists within the *MicroStation* BIM files, this information is under the oversight of the design contractors for the duration of the project. In preparation for handover, a register of every asset with the Crossrail project needed to be populated in AIMS. In order for AIMS to be ready in time for handover, it had been necessary to start populating information before the *MicroStation* BIM files were finalised.

By establishing an independent AIMS, under the direct oversight of the Crossrail Asset and Configuration Management Team, the register could be furnished with information without being hostage to the development of the federated CAD model.

It should be noted that information in AIMS is not the same as the Asset Information Model (AIM) as defined by PAS 1192:2014 (Section 3.4.2). Although the information in AIMS, as structured non-graphical information, is technically considered a part of the AIM, the AIM predominantly prescribes that the model shall be composed of graphical information in a federated 3D model.

The information in AIMS is structured according to the Asset Breakdown Structure (ABS) as outlined in Figure 5.4. The Crossrail *Complex* is broken down into *Facilities*, which may be either be located across the entire network, e.g. track or at a particular station, e.g. Liverpool Street Station. Facilities are broken down into *Primary Functional Units (PFUs)*, and then into *Functional Units (FUs)* and then into *Assets* as depicted in Figure 5.4. *Primary Functional Units* that do not belong to a particular *Facility* are broken down into *Locational Functional Units*. The term *Asset* is used to describe an object that represents a particular function. The term *Equipment* is used to refer to the item that fulfils the function of the *Asset*. As an example, the requirement for a control unit cabinet can initially be satisfied by a Mk 1 Cabinet, and then later be replaced by a Mk 2 Cabinet. Furthermore, each cabinet will have a serial number against which its maintenance history can be logged.

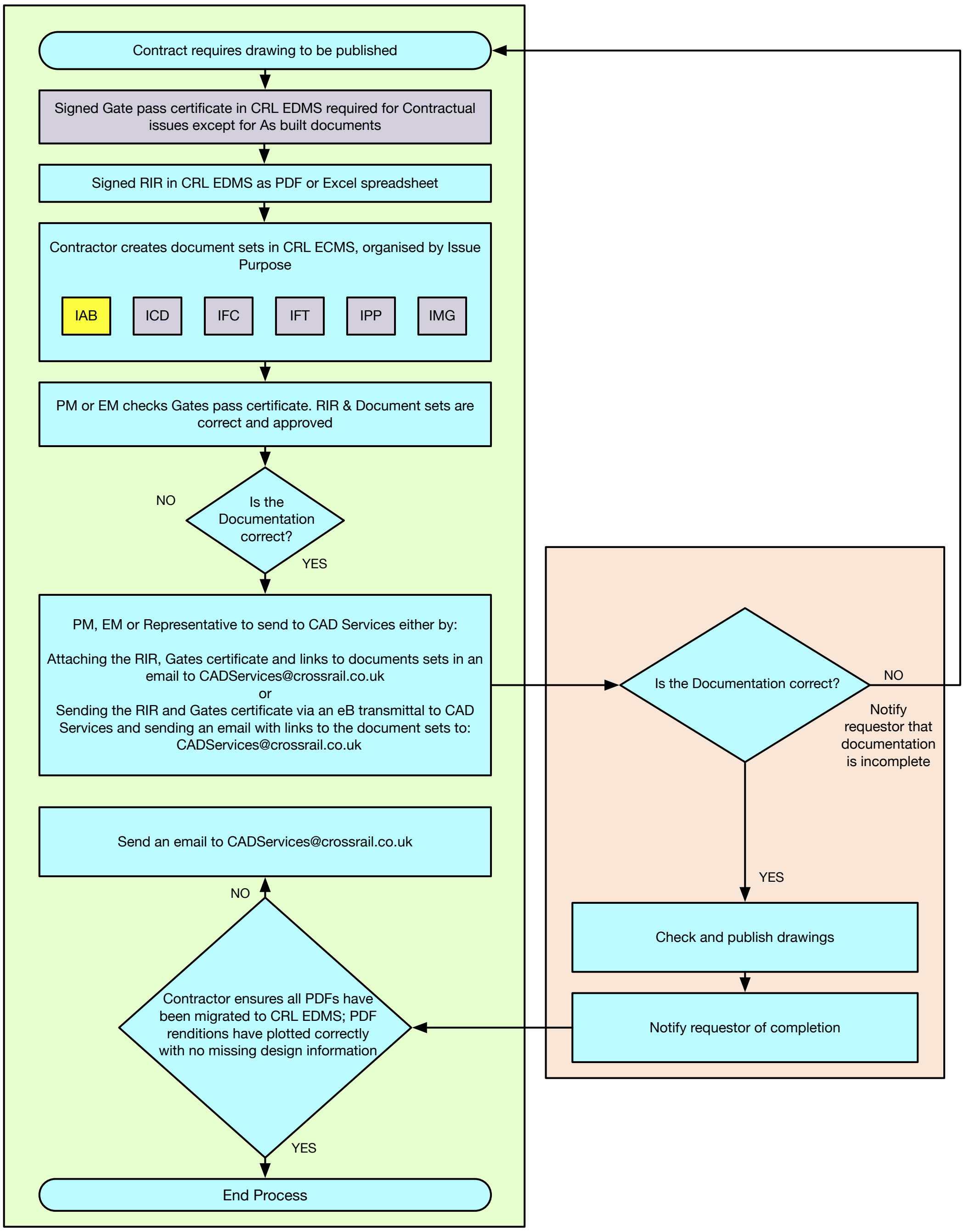


Figure 5.3: Flow diagram of CAD data team drawing request (Patel 2018)

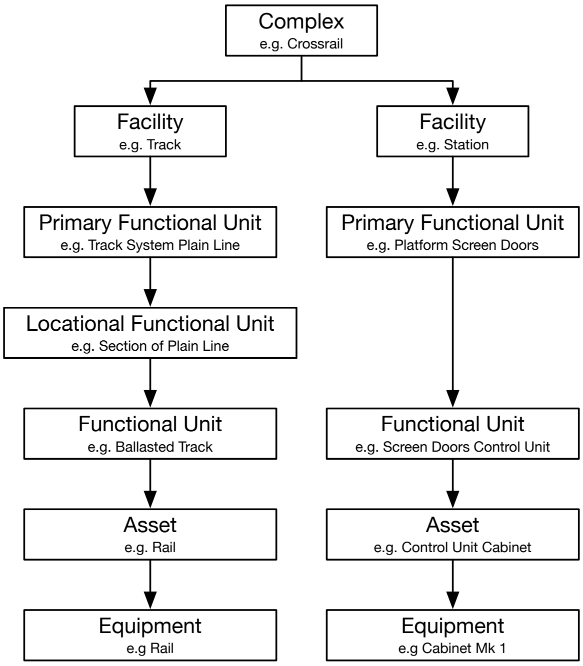


Figure 5.4: Crossrail Asset Breakdown Structure (adapted from (Crossrail 2017))

Every *PFU* and *FU* and *Asset* is identified with a unique *Asset ID* created from chaining a location code, a function code and a unique identifier, for example, *CR501-BAF-00001* (Crossrail 2017).

Three elements of information are provided for every asset. The first describes where the asset is located using a hierarchical format starting with the facility, the storey level, and then the space in which the asset can be found. The second describes the function that the asset supports such as TRK (*Track Plain Line*) or PSD (*Platform Screen Doors*). The third describes the class of the asset using a scheme of classification that is a modified version of *Uniclass 1.4 Table L* (CPIC 2014).

Detailed descriptions of each class of asset together with details on the information required to be held on each class of asset are specified in a suite of documents referred to as Asset Data Dictionary Definition Documents (AD4s).

The geometry of AIMS assets is not defined, however, the general location of assets is attributed by reference to the space in which the asset is located inside, or by the linear referencing along the track in the case of trackside assets.

5.1.3.4 Master Data Model

An in-house data warehouse was developed by Crossrail to pull together business information from various systems from which to produce management reports. At the early stages of the project, the quality of the reports suffered due to inconsistent use of identifiers within the various systems. To overcome this a *Master Data Model* was created to ensure that every information model used across the Crossrail Technical Information System, including non-spatial systems used for financial data and risk management, is configured with the same *Master Data* (Palmgren 2017). Each element in Figure 5.5 contains a list of terms that have a universal meaning across the project (Taylor 2017).

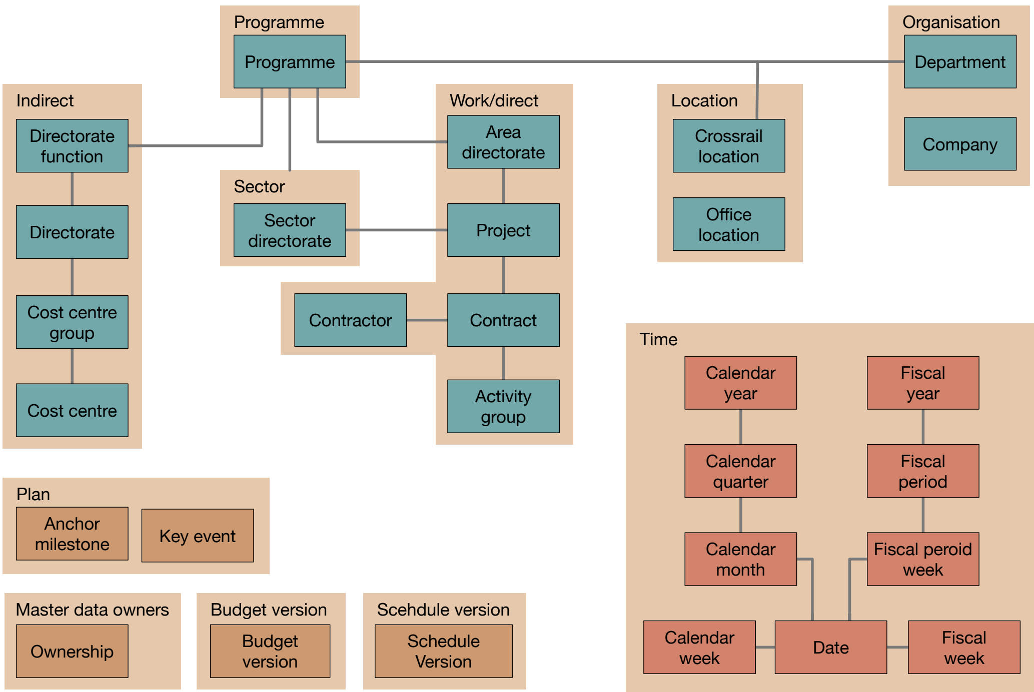


Figure 5.5: Master Data Model (Taylor 2017)

5.1.4 Crossrail Spatial Information Tools

5.1.4.1 Building Information Tools

The Building Information Tools are capable of accessing and interacting with the *Building Information Model* stored in the *Bentley MicroStation* platform. Initially, this is done using the BIM authoring tools provided by the *MicroStation* BIM extensions. There are, however, other tools used by Crossrail that can access files saved in the *Bentley MicroStation DGN* format.

Bentley Navigator is a clash detection tool that is used to identify instances where elements saved in disparate files in the federated model occupy the same volume in space. It also provides tools for collaboration.

Bentley Synchro is a 4D planning tool that is used to visualise construction along a timeline. It can be used to plan the order of construction in particular planning the incoming shipment of pre-constructed assets and equipment.

Bentley MicroStation is capable of exporting 3D geometry in *AutoCAD DWG* format and *Trimble SketchUp SKP* format. Furthermore, the BIM authoring extensions are capable of exporting BIM-related information and geometry in the Industry Foundation Classes (IFC) format. These model formats are used for information exchange to other software applications, thus extending the range of tools available.

5.1.4.2 Geospatial Information Tools

Crossrail staff and contractors are able to view site plans and logistical information through the *Crossrail Maps* application through a browser-based client accessing the *Geospatial Information Environment* through the Crossrail intranet.

More advanced spatial query tools and spatial analytic functionality can be run from within the toolboxes that are shipped with the *ArcGIS* software package. These tools include the *3D Analyst* toolbox that is licensed separately. Additionally, FME Workbench is used by the Crossrail GIS team as it provides an extensive suite of tools for manipulating and transforming element geometry, attributes and properties.

5.1.4.3 Asset Information Tools

Contractors populate the information in AIMS by uploading spreadsheets with the data to be inputted. Crossrail staff are then able to view asset information through a browser-based client accessing the server within the Crossrail intranet. Once operational, the information will be accessed through Engineering Asset Management (EAM) software capable of scheduling maintenance, such as *Maximo*.

5.1.5 Crossrail Spatial Information Activities

5.1.5.1 Building Information Activities

Primary contractors were responsible for the design and analysis of the infrastructure governed by their contracts. They are responsible for maintaining a quality management system and it is expected that they will have standard working procedures.

Crossrail is responsible for setting procedures for contractors to follow to quality check the CAD models that they produce (Patel 2018). Using the definitions set out in Chapter 4 the exchange and approval of building information between project stakeholders falls within the remit of *Spatial Information Management*.

5.1.5.2 Geospatial Information Activities

In the early days of the Crossrail project, GIS was predominantly used to plan the location of the route, stations and engineering infrastructure in preparation of gaining parliamentary approval in 2008. Geospatial information is now provided to the project stakeholders as a visual tool to provide them with spatial awareness so support day-to-day decision-making and logistical planning purposes (Irwin and Tamash 2016).

Information in the Crossrail *Geospatial Information Model* must be maintained on a regular basis. The Crossrail GIS Team author documentation giving instructions for technically skilled users to follow to maintain information in a consistent manner (Irwin and Tamash 2016).

5.1.5.3 Master Data Model Activities

Any request to change the master data model can only be implemented following approval from the owners of the relevant source systems that are managed by the *Master Data Model*. The relevant source systems are not automatically updated upon authorisation, but must be changed by the system owners (Palmgren 2017).

5.1.6 Crossrail Spatial Information Management

As explained in the preamble to this chapter, Crossrail was instrumental in the concurrent development and adoption of BS 1192:2007 in setting up a CDE. The publication of BS 1192:2007 was the first step in the national implementation of BIM as standard practice across the UK. Being an early adopter of BS 1192:2007, Crossrail pioneered the fundamental practices of BIM eight years before the implementation of the UK BIM mandate. It should be remembered that Crossrail is not expected to comply with the full suite of UK BIM standards, most of which were published after the project was in full swing.

The other standards in the UK 1192 suite, such as PAS 1192:2013 for the adoption of BIM in design and construction and PAS 1192:2014 concerning the use of BIM in asset management PAS 1192:2014, were published towards the middle of the Crossrail project. There is no requirement to achieve *Level 2* in compliance with the UK Government BIM mandate, however, there has been an endeavour within the Crossrail project to take on the principles of the UK 1192 suite of specifications as compelling guidelines. Although it is not always possible to apply current standards retrospectively to a project that was already underway for five years, this exercise has been useful for setting a benchmark.

The technical system architecture, security protocols, software, modelling standards described in the previous subsections all come together as a system for the management of information. Adherence to the system specification is essential for enabling the meaningful and efficient exchange of information within the project. These standards were written into the works specification contracts to achieve this meaningful exchange of information.

The information in the CDE is managed following the principles set out in BS 1192:2007 to ensure that only information authorised for access can be retrieved and edited by the appropriate stakeholders (Irwin and Tamash 2016). Tightly controlled security privileges prevent Crossrail employees from accessing documents designated as *Work-in-Progress* in the contractor's domain to all but the smallest number of technical staff at Crossrail.

While still in the contractor's domain, CAD documents are designated as *Work-in-Progress*. Once they are ready for approval, the contractor re-designates the CAD document to *Shared* before being accepted by a Crossrail representative who re-designates the document as being *Authorised*. The progress of documents through the approvals process is managed using the EDMS.

5.1.7 Crossrail Spatial Information Disciplinary Fields

The *Spatial Information Disciplinary Field* is the term used to describe the outer layer of the Spatial Information System. This level represents the professional culture and knowledge that influences the rest of the system, and this heading will be used to describe the people that use and manage the Spatial Information Systems within the Crossrail organisation.

The Spatial Information Systems are managed by the Technical Information Department under the Head of Technical Information. The group consists of the teams listed in Section 5.1.7 co-located close to each other in an open-plan office building within the Crossrail headquarters.

Table 5.1: Teams making up Crossrail Technical Information Department

Team	Description
CAD Team	CAD professionals with a background in engineering
GIS Team	GIS professionals with a background in geography and information systems
Asset and Configuration Management Team	Asset managers and configuration managers with a background in engineering
Document and Data Compliance Team	Administrative staff with a background in quality control

5.2 Assessment of Interoperability

This chapter has so far described each of the four Crossrail Technical Information Systems using each of the seven levels developed within the Spatial Information System framework. In this section, the ECMS, GIS and AIMS will be analysed to identify system heterogeneity at each level and potential interoperability that may be found there. Note that the *Master Data Model* and the EDMS will not be included in this assessment as they are incidental to the research challenges identified in Section 2.1.2.

This assessment is carried out from two perspectives. The first concerns the relationship between the ECMS, AIMS and the 2D GIS as used for *Crossrail Maps* as illustrated in Figure 5.6; the second covers the relationship between the ECMS, AIMS and the 3D GIS as set up to resolve the research challenge identified in Section 2.1.2 as illustrated in Figure 5.7.

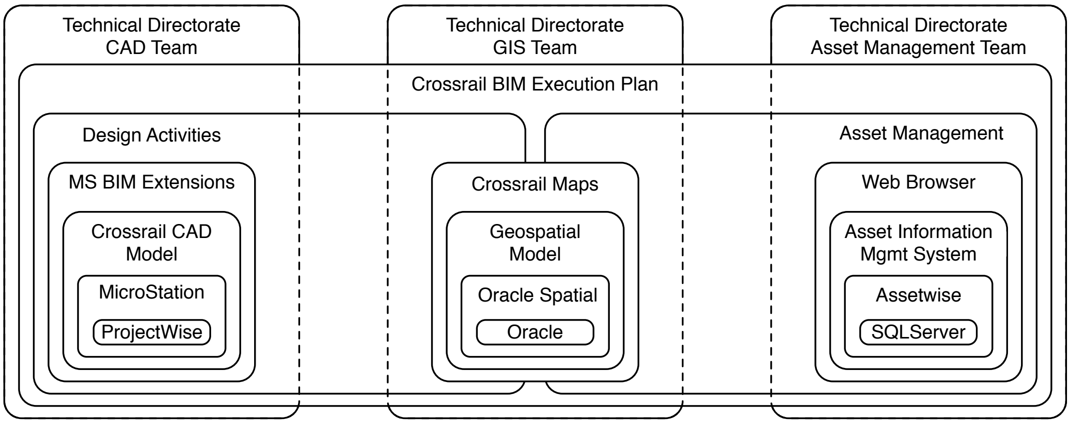


Figure 5.6: Summary of Crossrail Technical Information Systems (As implemented)

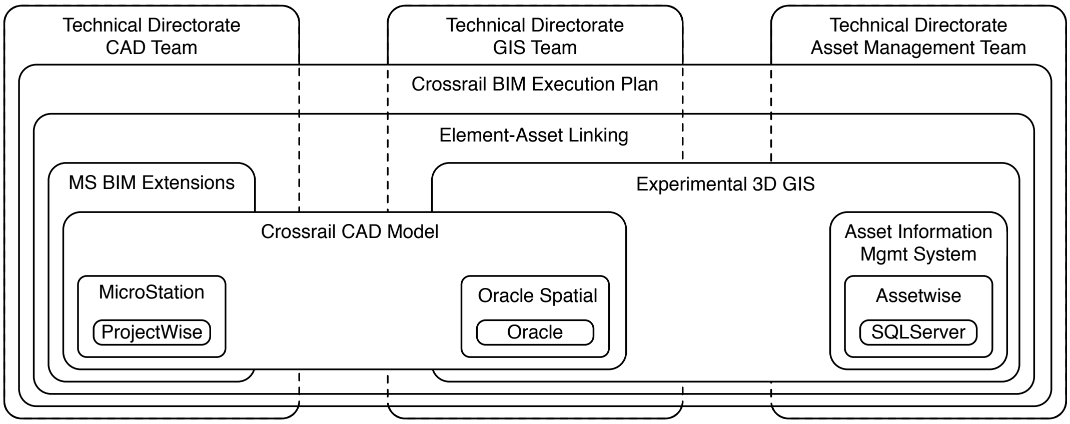


Figure 5.7: Summary of Crossrail Technical Information Systems (As required)

5.2.1 Spatial Information Environment Level

In principle, there are no significant interoperability issues at the technical levels during the delivery phase of the project. The only significant issues arise when trying to configure the network architecture to allow users from outside the Crossrail intranet to access the environments due to network security constraints.

After the handover of the information, the future Infrastructure Manager (IM) will need to implement the same *Spatial Information Environment* and *Platforms* as implemented during the delivery phase, if the information is to be interoperable over time. It should be noted that the future IM, Rail for London (RfL), does not intend to use the 3D models created during the delivery phase on a daily basis. Instead, the 3D models will be archived, and the as-built 2D drawings will be generated from the 3D models (MacDonald 2016).

The *Spatial Information Environments*, i.e *ProjectWise*, *Oracle* and *AssetWise* are configured to support their respective *Spatial Information Platforms*.

5.2.2 Spatial Information Platform Level

It is generally the case that the software applications, i.e. *MicroStation*, *Oracle Spatial* and *AssetWise*, are only capable of hosting that part of the information model that they are designed for. In the case of *MicroStation*, tools exist for importing and exporting information written in different formats; third party applications also exist, such as *FME Workbench*, to read and write information in a multitude of formats.

5.2.3 Spatial Information Model Level

Bishr (1998) identified that system interoperability primarily arises from model heterogeneity, specifically syntactic, schematic and semantic heterogeneity (Section 3.3.1). Figure 5.8 summarises these syntactic, schematic and semantic differences for the CAD models in *MicroStation* and IFC format, 2D and 3D GIS, and the AIMS asset model. The schematic differences are described in terms of functional breakdown, hierarchical structure, unique identification and geometry.

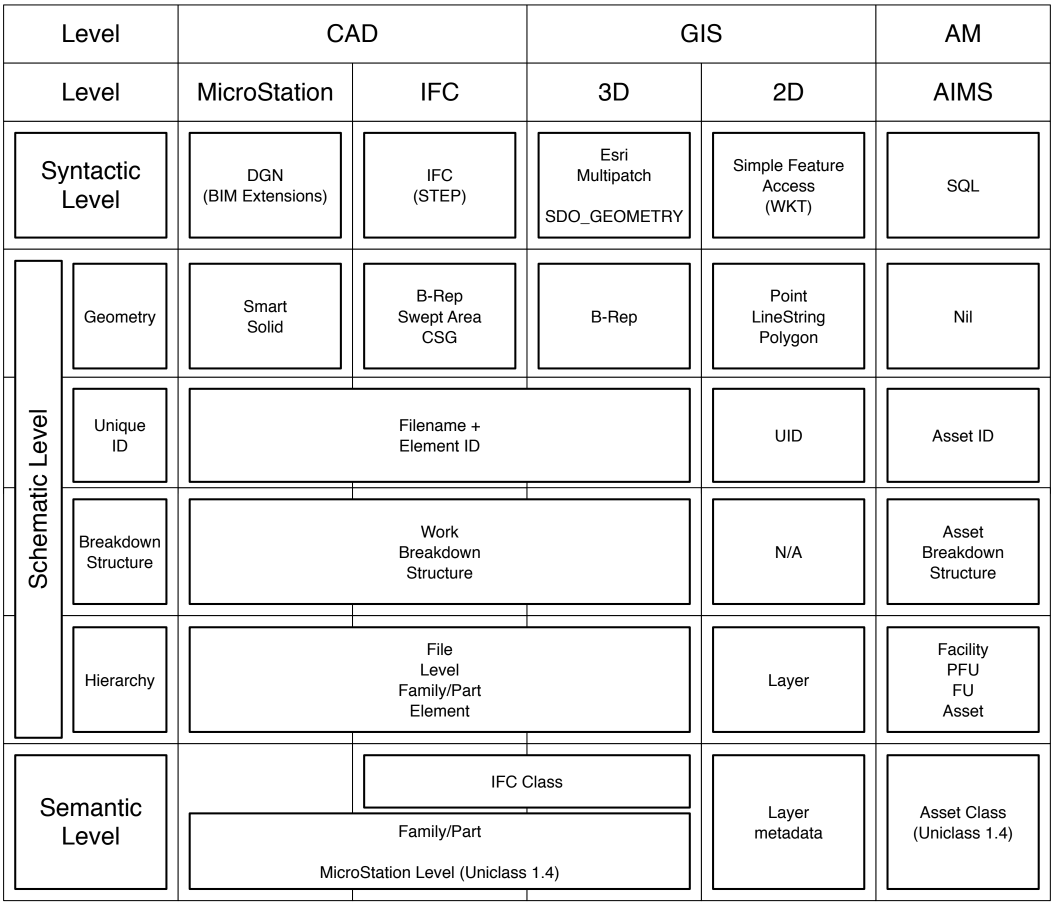


Figure 5.8: Summary of Crossrail Technical Information Systems at Spatial Information Model level)

5.2.3.1 Syntactic Level

CAD information is syntactically formatted in the proprietary *DGN* format. Although there is limited access to read and write 2D elements, *Bentley* does not freely publish the syntax for accessing 3D geometry elements. Tools do exist in *MicroStation* and the BIM extensions to import and export *AutoCAD* 3D CAD models (*DWG*), *Trimble SketchUp* (*SKP*) and IFC. These IFC files are written in the open STEP file format (STEP) syntax (or Extensible Markup Language (XML) syntax).

2D GIS features are stored in ORDBMS tables that can be accessed via Sequential Query Language (SQL) the using Well Known Text (WKT) geometry format or the *Oracle SDO_GEOMETRY* format. Alternatively, 2D and 3D features can be read directly from *Esri ArcGIS* applications using *ArcSDE*.

Information in AIMS is also formatted in ORDBMS tables accessed using SQL. Information stored in non-structured formats, e.g. PDF documents, can be accessed; however, handling the information has its own separate challenges.

Except for the *MicroStation DGN* format, syntactic heterogeneity is not an issue because the syntax is openly published and freely licensed for use. Interoperability is achieved using either import and export tools or third party applications, such as *FME Workbench*.

5.2.3.2 Schematic Level (Geometry)

Although differences in geometry are most apparent at the syntactic level, they initially arise at the schematic level (see Wise (2010) in Section 3.2.3). The *MicroStation* BIM extensions predominantly create element geometry using the proprietary *SmartSolids* geometry type. This element geometry type is a compound format that uses CSG methods to construct Boundary Representation (B-Rep) geometries from other *MicroStation* geometry types. As well as the *SmartSolid* elements, a *MicroStation* file will contain other elements including textual annotations, dimensions and construction elements. These construction elements can be distinguished from other model elements by reading the construction attribute.

The *MicroStation* BIM extensions will export these *SmartSolids* to IFC as *Sweep Representation* if the geometry is sufficiently simple, however, the export tools will otherwise revert to converting these *SmartSolids* to a planar B-Rep mesh. Any geometry not already defined using B-Rep must be converted before it is capable of being stored in a 3D GIS (either as *MultiPatch* or *Oracle SDO_GEOMETRY*). All geometry must be flattened into a polygon for handling within 2D GIS (with an optional elevation attribute to achieve 2.5D).

The conversion of *MicroStation* geometry into 3D GIS compliant geometry is a significant interoperability challenge first raised in Section 2.2 and will be addressed in Chapter 6 and Chapter 7.

AIMS assets do not contain any geometric representation, although they are attributed with their general location using the name of the space in which they are located. The lack of geometric representation is detrimental for the reasons set out in Section 2.1.2, and it is this lack of geometry that is a significant driver for establishing relationships between elements in *ProjectWise* and the assets in AIMS.

Spaces are represented as 2.5D floors plans, and there is a tool in the *AECOsim Building Designer* BIM extension for defining 2D spaces which permits manual and semi-automatic construction. However, the requirement to construct spaces within the 3D model was not contractually specified. Instead, contractors delivered floor plans as 2D CAD drawings that were not geo-referenced in *London Survey Grid*. These floor plans contain a reference to the storey level but do not otherwise contain floor elevation or ceiling height. These 2.5D floor plans are used by the asset managers to locate the space in which an asset is located. However, the lack of full 3D geometric representation is a significant interoperability challenge first raised in Section 2.2 and will be addressed in Chapter 8 and Chapter 9.

5.2.3.3 Schematic Level (Unique ID)

Every geometric element in a *MicroStation* file is attributed with an integer that represents a unique *Element ID*. If this *Element ID* is combined with the *MicroStation* filename, then an ID is generated capable of uniquely identifying every element in the ProjectWise. When a *MicroStation* file is exported in the *AutoCAD DWG* format, this *Element ID* integer is converted to a hexadecimal alpha-numeric string. It maintains an identifiable relationship with the corresponding element in the *AutoCAD* model. When exported to IFC, the *MicroStation* Element ID is concatenated with the filename (and the sub-file model) and held in the *Description* attribute.

Each feature in the 2D GIS is identified using its own UID; however, all element imported from *ProjectWise* via IFC into the 3D GIS will carry the GUID assigned by the *MicroStation* export tool and the *MicroStation* Element ID held in the *Description* attribute.

AIMS assets using a different identification system based on concatenation of the assets location code, function code and a unique identifier.

The schematic heterogeneity in uniquely identifying elements and assets is a significant interoperability challenge first identified in Section 2.1.2. It is the aim of this thesis to address interoperability issues as identified in Section 2.2 to work towards overcoming this challenge; however, this challenge will itself not be addressed.

5.2.3.4 Schematic Level (Breakdown Structure)

Information models created using the *Bentley MicroStation* BIM extensions exist as *parts* which belong to a *family* class. Component parts can be aggregated into compound parts. These aggregated relationships must be mapped to aggregate relationships in IFC. However, all component parts of a compound part must be saved within the same level of the same file in the federated model. Component parts in *MicroStation* are created by designers to direct the construction of built assets and are structured according to the Work Breakdown Structure (WBS).

The corresponding grouping in AIMS is based upon a hierarchical ABS that supports the future IM's Asset Management Strategy. The ABS consisting of *Facility*, *PFUs*, *FUs* and Assets does not necessarily correspond to the hierarchical WBS in *ProjectWise*.

The difference in the philosophy behind how the WBS and ABS are structured leads to incompatible systems for linking individual components. While it may be straightforward to establish *1:1* relationships between elements and assets, there may be other relationships that need to be considered.

By way of a hypothetical example, Figure 5.9 sets out four diagrams showing how a single floor slab might be represented differently in *MicroStation*/*ProjectWise* and AIMS. In diagram (a) a single slab is represented as a single element in CAD and single asset in AIMS; for this, a simple *1:1* relationship can be established. Over in diagram (b), however, it may be that the slab is constructed from two individual elements that are considered as a single asset in AIMS; in which case the *Element 1* and *Element 2* have an *M:1* relationship with *Asset A*. Alternatively in diagram (c), responsibility for a single floor slab *Element 1* has been allocated to two different asset managers (for example in a connecting corridor between an existing London Underground station and a new Crossrail station); this would represent a *1:N* relationship. Finally, a combination of situations in diagrams (b) and (c) are depicted in diagram (d) as an *M:N* relationship.

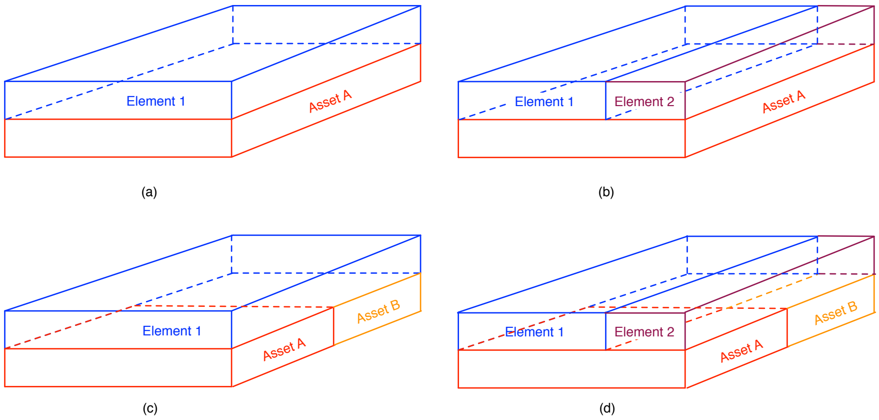


Figure 5.9: Incompatible relationships between elements and assets (a) 1:1, (b) M:1, (c) 1:N, (d) M:N.

This schematic heterogeneity creates a significant interoperability challenge that has been identified as a research gap in Section 2.1.2. It is the aim of this thesis to address interoperability issues as identified in Section 2.2 working towards overcoming this challenge.

These incompatible relationships mean that there is an ambiguous relationship between elements and assets between the two systems. This ambiguous relationship could be overcome by breaking up the geometry of the CAD elements so as to correspond with the AIMS asset breakdown, and break up AIMS asset to match the CAD element. This extensive exercise seems to have limited benefit with regard to partitioning AIMS assets, although breaking up the CAD elements would bestow AIMS assets with a geometric representation that could be used for analysis and visualisation.

5.2.3.5 Semantic Level

MicroStation CAD elements and AIMS assets are each classified using different schemes of classification system each modified differently from *Uniclass 1.4* (CPIC 2014). The method of attributing this semantic classification also differs, with a CAD element being classified by allocating the element to a *MicroStation* level, in contrast to an AIMS asset which is classified through attribution.

Although the two classification systems are similar, there are widespread differences between them. Furthermore, due to ambiguities in the class descriptions, the assignment of elements and assets to their respective classes is dependent on the subjectivity of the user creating the entity. As an example, a collection of staircase handrails at one Crossrail station were classified as being part of staircase, which subsequently had a detrimental effect when trying to visualise walkable surfaces.

The use of 3D features in GIS corresponding to *MicroStation* model elements is still experimental. These features are not classed using their own classification system but instead follow the classification of the model element from which they are derived.

5.2.4 Spatial Information Tools Level

The *Spatial Information Tools* used by Crossrail can only work with the *Spatial Information Models* for which they have been designed. The exception being those tools that are written for import and export of information. In this regard, *FME Workbench* is uniquely positioned to perform Extract-Transform-Load (ETL) operations between the *Spatial Information Platforms*.

5.2.5 Spatial Information Activities Level

Spatial Information Activities are generally carried out by Crossrail staff and primary contractors who belong to a particular disciplinary field. For example, CAD technicians will generally work with CAD tools, GIS Professionals will work with GIS, and asset managers will generally work with information in AIMS.

There may be occasions where one professional will need to use tools outside their disciplinary field; however, these activities were not researched as part of this thesis.

It should be noted that the semantics and schema of the information models will be developed to support the *Spatial Information Activities* and the aims and objectives of the project.

5.2.6 Spatial Information Management Level

As a project, Crossrail is dedicated to the principles of BIM. All Technical Information is guided by the *Crossrail BIM Principles* documents, written by the Head of Technical Information and authorised by the Crossrail Chief Executive. However, closer inspection of the document will reveal that it is split up into sections corresponding to CAD, GIS and Asset Management (AM).

5.2.7 Spatial Information Disciplinary Field Level

The Crossrail staff responsible for the Spatial Information Systems are co-located in a single Technical Information Department. Although grouped into specialised teams, good communication and team building ensure that the staff are familiar in the principles of the systems outside their discipline.

5.3 Summary

This chapter has used the novel Spatial Information System Framework developed in Chapter 4 to break up the Crossrail Technical Information Systems into seven levels. Each level is described in detail and used to identify system heterogeneity that has the potential to manifest as an issue hindering interoperability. The interoperability issues identified in Section 5.2 helped to clarify the scope of the challenge and were retrospectively used to shape up the supporting research questions in Section 2.2. In Chapter 6, a method to extract, transform and load elements from *MicroStation* files to a GIS compliant format will be developed with particular focus on overcoming the interoperability issues identified in Section 5.2.3.2.

6 Development of Extraction Transformation and Loading Workflow

As set out in Chapter 5, the information used for Building Information Modelling (BIM) at Crossrail is stored in two heterogeneous information systems, *ProjectWise* and *AssetWise*. *ProjectWise* comprises a federated model consisting of *MicroStation* Computer Aided Design (CAD) files each of themselves consisting of a collection of geometric elements. *AssetWise* is an Object Relational Database Management System (ORDBMS) containing assets stored as rows in tables.

The two information systems were populated independently (Section 5.1.3.3) and because of this, although individual entities do correspond with each other, the direct relationships between them are not yet determined. The entities neither share a common identifier nor do they share a common scheme of classification.

In Chapter 2, the requirement to establish links between CAD entities in *ProjectWise/MicroStation* and *AssetWise/AIMS* was identified as a shortcoming in the aspirations of Crossrail to comply with PAS 1192-3:2014. In pursuit of meeting this aspiration, a method is proposed in Section 1.2.1 to link entities based on their spatial location and their element and asset classes. This linking method, however, requires three prerequisite steps as illustrated in Figure 2.3; the first of these (PRS 1) is the extraction of CAD elements from *ProjectWise/MicroStation* and their transformation into a Boundary Representation (B-Rep) format compatible with three-dimensional (3D) Geographic Information Systems (GIS).

This chapter will address the interoperability challenges as highlighted in Section 2.2 that have been experienced in practice. It will thus investigate the various options for extracting CAD elements created with the *MicroStation* BIM extensions and transforming them into B-Rep. *FME Workbench* is a software application used throughout industry and academia for interchanging geometric information between different applications, and it had been hoped to take full advantage of this application to export the CAD elements, formatted as *Industry Foundation Classes (IFC)*, and upload them to a Spatial Data Warehouse (SDW). However, exporting via *IFC* was found to be more problematic than expected due to loss of information and corrupted geometry, prompting a systematic investigation into alternatives ways of doing things and the development of workaround methods for resolving persistent challenges[1].

In the earlier stages of this research, an as-then unresolved technical issue was causing *FME Workbench* to lock up when reading some geometric entities from *IFC* files. So as to overcome this bug, a workspace was developed that ignored the *IFC* geometry and substituted it with geometry from *DWG* and *SKP* files. The technical issue was eventually traced to the export function in the *MicroStation* BIM authoring extension used in *Select Series 2* on the Crossrail installation. Upgrading to the *AECOsim Building Designer*, as installed on the UCL network, resolved this issue, thus enabling *IFC* to be used as the principal export format. The alternative methods and workarounds that were developed to overcome this significant challenge will now be used to replace the small percentage of missing and deformed geometry that do still occur, albeit less frequently.

Once the various options have been evaluated, this chapter will go on to propose a practical workflow to extract, transform and load elements from *MicroStation* to a 3D GIS platform. Once implemented, the effectiveness of this workflow will be tested and analysed in Chapter 7. As well as presenting a narrative of the work carried out to obtain a working feature set, the two chapters provide a case study that will be used to assess the practical interoperability between BIM-based and GIS-based information.

[1] An initial report on the interoperability challenges experienced and an early version of the practical workflow developed is published in a preceding paper that is within the scope of this thesis (Boyes, et al. 2017).

6.1 Aim of ETL Operation

Once transformed into B-Rep, these elements will be used for the next steps in preparation towards the ultimate aim of linking asset information. For the second step (PRS 2), the transformed elements will provide input for generating 3D representative spaces. Depending on the ultimate method chosen in Chapter 8 to create spaces, the transformed B-Rep elements may need to be watertight solid features, and the workflow developed here must work towards achieving this aim.

For the third prerequisite step (PRS 3), it is intended to use the transformed elements with the 3D spatial analysis tools in either *Oracle Spatial*, in particular, the *SDO_INSIDE* and *SDO_ANYINTERACT* functions, or the *Inside3D* tool that is part of the *3D Analyst* toolbox in *ArcGIS*. As well as providing spatial analytical functions, *Oracle Spatial* has been adopted to provide the GIS platform in this project because it is a robust storage environment that can be managed as an SDW. Furthermore, *Oracle Spatial* can be easily accessed from both *FME Workbench* and *Python* scripts. *Esri ArcGIS Shapefiles* or *geodatabases* are an equally suitable alternative format, but they were not chosen as a suitable *Python* package was not available to access *MultiPatch* geometry inside *Shapefiles* or *geodatabases*. Likewise PostgreSQL/PostGIS would also be a suitable alternative to act as a repository of information, however, the platform does not provide the tools to perform 3D spatial queries on polyhedral surfaces.

In a broader context, the extraction and loading of elements from CAD to GIS need to be performed with confidence and dependability. In future applications, there is every likelihood that the output of Extract-Transform-Load (ETL) operations will be relied upon to make critical decisions concerning safety and efficiency. With this in mind, it is essential to identify every element that is a candidate for extraction and ensure that it is accounted for throughout the process, confirming that the resulting geometry matches the original geometry without unexpected distortion.

6.2 Extraction of CAD Elements

In this section, the various options for extracting CAD elements created with the *MicroStation* BIM extensions and transforming them into B-Rep will be investigated. It will consider the advantages and disadvantages of exporting the elements as *IFC*, *AutoCAD* models, *SketchUp* models as well as the native *MicroStation DGN* format.

It should first be noted that there is no functionality in *MicroStation* to export 3D elements directly into *Oracle Spatial*. Likewise, there is no functionality in *Oracle Spatial* to import *MicroStation* elements. Middleware must therefore be used to perform an ETL operation (Section 3.3) on the elements, for which *FME Workbench 2019.1* will be used here.

FME Workbench is a graphical programming language, developed by *Safe Software* (Safe Software 2017), that is widely used in GIS and BIM for solving interoperability issues that exist between different applications (Jusuf, et al. 2017; Zhu, et al. 2019). The application allows users to write workflows within a desktop Graphical User Interface (GUI) which can then be run locally, or be uploaded to a remote server for background batch processing.

The *FME* users add workflow components, referred to as *Readers*, *Writers* and *Transformers*, to a graphical workspace where they can then be connected up like an electrical circuit diagram. The workspace readers import features, which then flow through the transformers to writers which save the features to a specific file format or upload the features to a database. In between the *Readers* and the *Writers*, the *Transformers* perform bespoke operations on the geometrical features and their attributes.

6.2.1 Extraction Criteria

The proposed process will essentially involve four steps: export elements from *MicroStation*, read elements into the middleware, transform the elements into B-Rep, and load the transformed elements into *Oracle Spatial*.

Five different ways of exporting of elements from *MicroStation* will be investigated; these are the native *MicroStation DGN* format, running a script written in *MicroStation Visual*

Basic Application (MVBA) to export a *CSV* file, the *AutoCAD DWG* format, the *IFC* format, and the *Trimble SketchUp SKP* format. Each format has advantages and disadvantages, and these will be investigated and considered alongside the following set of criteria. These criteria have been chosen in consideration of the proposed linking method and the second and third prerequisite steps.

Most importantly, *MicroStation* should be able to export every element that makes up the model, and *FME* must be able to read every element with reliability. If this is not possible, the error should be handled gracefully and logged, without crashing applications.

The exported geometry should also accurately correspond to the original geometry and be fit for its intended purpose. As well as having no conversion faults involving chronic disfigurement, the distortion of the transformed element should be within a tolerance set by the intended purpose of the geometry.

As part of the second prerequisite step to generate 3D spaces, the transformed geometry will be voxelised. For the present purposes, it is not intended to use a voxel size of less than 0.25 metre. The geometric tolerance should, therefore be significantly less than this distance.

The fundamental purpose of the proposed linking method is to link the identifier of CAD elements with the identifier of the Asset Information Management System (AIMS) assets. Therefore, the third criterion is that the *MicroStation Element ID* attribute that uniquely identifies every element should survive exportation, or there must be some alternative means of reuniting the element with its identifier.

As well as linking elements/assets based on their spatial location, the proposed linking method will also use information based on the semantic class that the elements belong to. For the fourth criterion, it is desirable that as much semantic information as possible be exported.

Finally, given that the ETL operation needs to be performed on perhaps millions of elements in thousands of files, both the export and the transformation operations must be capable of batch processing.

Consequent to this, if the export is to be performed automatically, the batch process should include a self-checking operation to confirm that all elements in the federated *DGN* files are accounted for in their new format.

Before reporting on this investigation against these criteria, it should be noted that the original CAD files used in this investigation were authored using *Bentley MicroStation V8 (Select Series 2)* and BIM authoring extensions, and the initial work for this research was carried out using this version. However, problems with bugs in this outdated version prompted a move to *Bentley AECOsim Building Designer (Select Series 6)*, a release of *MicroStation* incorporating all the BIM extensions. Some of the lessons learned from working around these bugs (as reported in Boyes, et al. (2017)) are no longer relevant after upgrading the software.

6.2.2 Extracting via DGN files

FME Workbench has the ability to work with the native *MicroStation DGN* format, but it is limited to reading those core elements which are based upon the openly available Interactive Graphics Design System (IGDS) format (Safe Software 2017a). The geometry created by the BIM extensions is stored in a proprietary geometry type, referred to in *MicroStation* as a *SmartSolid*, that is not supported by *FME Workbench*, and thus *FME* is not capable of reading the CAD elements created as BIM objects.

6.2.3 Extraction using MicroStation Visual Basic

Functionality exists in *MicroStation* to automate operations and perform basic queries using scripts written in *MVBA*. This functionality can be used to count the number of elements in each *DGN* file, access metadata on each element and save the information as a *CSV* file. This element count can be used to provide a baseline from which to account for each element during the ETL operation. The data harvested includes the filename, model name, *MicroStation* level, *Element ID*, geometry type, construction status, and minimum bounding volume coordinates. Although the *MVBA* script cannot export element geometry, this method provides a way of extracting metadata that is not exported using the other techniques, i.e. the *MicroStation* level and the construction status.

6.2.4 Extraction via AutoCAD DWG

MicroStation is capable of exporting 3D CAD models in the *AutoCAD DWG* format, but without the BIM information contained in the BIM extensions. *FME Workbench* fully supports the reading and writing files in this format. Furthermore, exporting models in the *DWG* format was found, from this investigation, to be more reliable than *IFC* and *SKP* at not losing elements.

Those elements with curved surfaces in CAD formats are interpreted in *FME Workbench* as high-fidelity B-Rep, at the expense of higher data storage size.

MicroStation Element IDs are accessible in *FME Workbench* where they are known as *Entity Handles*. However, reading the *Element ID* of *SmartSolids* is not straightforward as block entities in *AutoCAD* are assigned a new *Entity Handle*. This investigation discovered an undocumented procedure for exposing the underlying *Element ID* by setting an option in the *FME Reader* to preserve block entity insertion points.

Working with geometric elements, it is sometimes more appropriate for software to break up an element into multiple parts, while still managing it as a single entity. While both *MicroStation* and *AutoCAD* have this ability to handle multi-part geometry, *FME Workbench* splits these elements up into separate *FME* features. When this occurs, the *Element ID* can no longer be relied upon to identify the element uniquely, and a part number must be created for each sub-element. Figure 6.1 depicts a single element that is part of the *Broadgate Ticket Hall* ceiling that fits around a column. The element is represented as two parts with one shown in the figure as opaque and the other transparent. In *DGN* and *IFC*, the parts are aggregated together, but in *DWG* and *SKP* the parts are separated. The use of two different hierarchical structures causes problems when attempting to account for the total number of elements as these parts share the same *Element ID*. In this thesis, the word *feature* shall refer to elements once they have been broken up into parts so as to distinguish them from elements aggregated under a unique *Element ID*.

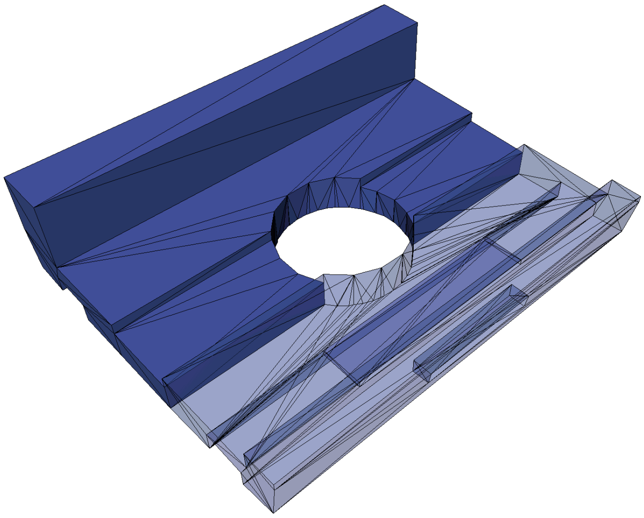


Figure 6.1: Example of multi-element decomposed into two parts

When the features read from *DWG* files are tested with the *FME GeometryValidator* transformer, practically all of the features derived from planar CAD elements are deemed to be valid or can be repaired. However, features with curved surfaces do not survive as valid solids and cannot be uploaded to *Oracle Spatial* as watertight solids. These features can still be handled and uploaded as non-watertight surfaces.

6.2.5 Extraction via IFC

The *MicroStation* BIM extensions support the export of CAD elements in the *IFC 2x3* format and provide unofficial beta support for export in *IFC 4*. Each CAD element is exported as a semantically classed object with attributes, properties and inter-element relationships. Furthermore, each element in *IFC* has the unique *MicroStation Element ID* contain within its *Description* attribute.

IFC is capable of representing each element using a variety of geometrical methods. From observing the exported IFC file, it would appear that the BIM extension chooses the most appropriate representation for each element. While the majority of *SmartSolid* elements are transcribed as B-Rep, it would appear that simple objects with planar faces are exported as Constructive Solid Geometry (CSG) or as a Sweep Representation. *FME Workbench* must resolve these representations into planar B-Rep before elements can be uploaded to a 3D GIS platform.

A number of problems were encountered using *IFC* as a conduit for extracting geometry from *MicroStation*. The first concerns the inclusion of construction elements in the *IFC* file; when elements are created in *MicroStation*, they are tagged by default as construction elements, but as the model progresses, the tag is changed from *construction* to *primary*. Although the option exists to filter out construction elements when exporting to *AutoCAD DWG*, this functionality seems to be unavailable when exporting to *IFC*. It is, therefore, necessary to identify these elements, either manually or automatically, and remove them before using exported elements.

A second problem concerns the quality of exported geometry. Although generally reliable, there are instances where geometry exported via *IFC* is lost or unrecognisably deformed. The occurrence of the error does not appear to be common; for example, only three elements were found to be unrecognisably deformed or null when exporting 1600 elements in the case study. However, deformations can be significant, as illustrated in Figure 6.2. In this illustration, the element of concern should only be the small brown wedge in the floor of the corridor to the right of the shaft, but instead, it is represented as a large semi-circular slab covering half of the shaft.

Although this deformation appears to be due to a bug in the software, incorrectly represented geometry such as this has the potential to cause consequential errors and lead to a loss of confidence in the format. Automatic checking methods are, therefore, needed to identify rare misinterpreted geometry whenever it occurs.

A third problem was encountered during the early stages of this research when using *MicroStation (Select Series 2)*. *IFC* files exported with this version caused *FME Workbench* to lock up while reading geometric representations. A workaround was possible to read semantic information about elements by selecting an option to read only the Minimum Bounding Volume. This investigation discovered that upgrading to *Bentley AECOsim Building Designer (Select Series 6)* overcame the problem; however, Boyes,

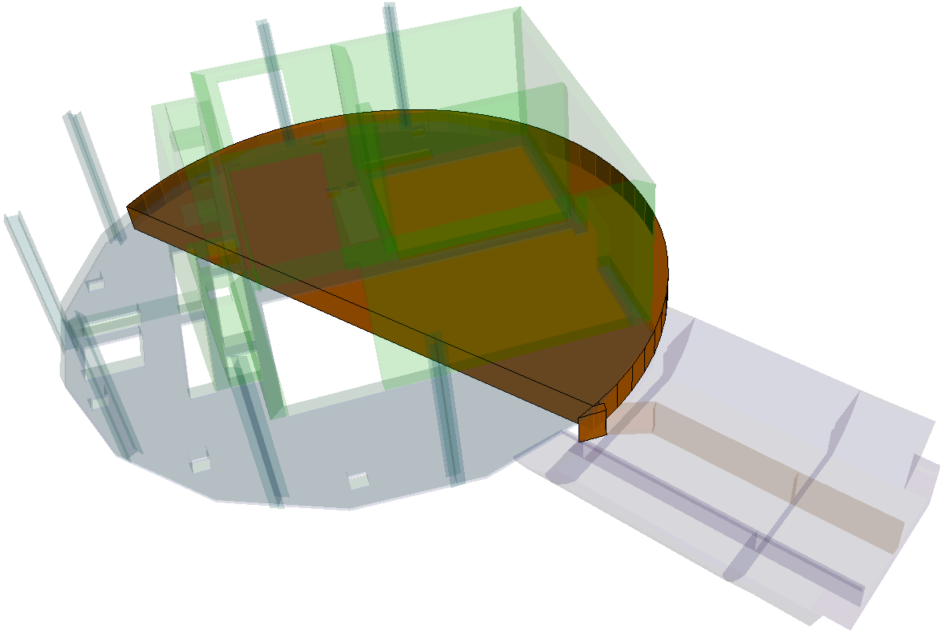


Figure 6.2: Deformed geometry from Mile End Shaft

et al. (2017) initially concluded that *IFC* was an inappropriate format for extracting geometric features from *MicroStation*.

When the problematic elements are isolated and inspected in *Solibiri Model Viewer* (Solibri 2017), a close examination of the geometry reveals polyhedral slivers extending out from the element surface (Figure 6.3). It is possible that these slivers arise from transforming CSG to B-Rep. In boolean subtraction, a *cutter* solid is used to subtract volume from a larger profile. If one face of the cutter is coincident with a face of the original profile, very small errors in the geometry (possibly due to floating-point calculations), can result in a polyhedral sliver that is unresolvable in *FME Workbench*.

No further investigation into this problem is considered necessary as later releases of *MicroStation* have overcome the problem. However, this problem is worthy of consideration for two reasons. Firstly, although a solution was able to be found for this issue, the time required to investigate the problem and the upgrade of software inevitably adds to the expense of establishing interoperability between systems. Secondly, if this corrupted *IFC* file had been relied upon to provide an archive format, rather than the original proprietary format, then there would have been a long-term loss of information.

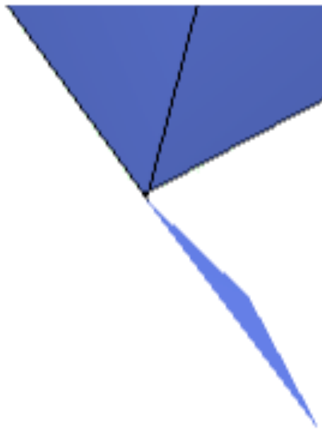


Figure 6.3: Polyhedral sliver created from IFC-sourced CSG

6.2.6 Extraction via SketchUp

CAD elements in the *MicroStation* models can also be exported and read into *FME Workbench* as *Trimble SketchUp* (*SKP*) files. Using this option, the geometry is converted to B-Rep within *MicroStation* and not in *FME Workbench*. As a consequence of using this route, curved surfaces are simplified, resulting in less representative geometry but lower data storage sizes.

Elements exported as *SKP* files are not attributed with any semantic information other than the name of original level in *MicroStation*. This information is hidden within *FME Workbench* as geometry *trait* information. Elements are not tagged with their *MicroStation Element IDs* when exported into an *SKP* file. This shortcoming can be resolved using one of two workarounds to reunite elements with their *MicroStation* ID.

The first workaround exploits the attribution of each geometric element in *SKP* with its level name. Before exporting the model from *MicroStation*, an *MVBA* script can be run that moves each element into a new level named after the *MicroStation Element ID*. Once read into *FME Workbench*, the *MicroStation* ID can in most cases be gleaned from the level name, however, the changing the level in *MicroStation* is only about 90 percent successful. The reason for this may be unclear, but it may be that some elements are locked into their levels by the BIM extension.

In these cases, a second workaround can be relied upon that attempts to marry up geometry with their *Element IDs* based on the mid-point of the element. However, the mid-point of the element is not always calculated identically due to distortions in geometry caused in the transformation of geometry into B-Rep (Section 6.3).

Just as when exporting to the *DWG* format, multi-part non-contiguous elements are split up while being read into *FME*. Furthermore, some geometries with *tunnels* passing through them result in the component being split up into parts.

There is, however, a cause for concern that *MicroStation* loses some elements when exporting via *SketchUp*. A process is therefore required to identify these missing elements and replace them with geometry from an alternative source such as *AutoCAD* or *IFC*.

6.3 Geometry Transformation

One of the criteria identified in Section 6.2.1 is that the transformed representation should accurately reflect the original *DGN* geometry taking into account that the representation will be voxelated at a resolution of 0.25 m (Section 8.3.3.3).

Each of the three geometry export methods, *DWG*, *IFC* and *SKP*, resolves the *MicroStation* geometry into B-Rep using a different interface. The *DWG* format maintains geometry in a CAD-based format and geometry is converted to high-resolution B-Rep on import to *FME Workbench*. However, it should be noted that *MicroStation* failed to export files to *DWG* that contain extensive curved geometry, such as tunnel shield walls.

The *IFC* format is capable of representing simple shapes as CSG and two-dimensional (2D) extrusions; however, more complex shapes are converted to B-Rep within the *AECOsim Building Designer* version of *MicroStation*. The more simple shapes are converted to B-Rep within *FME*.

SketchUp SKP is the only format that is wholly converted to B-Rep within the main *MicroStation* application. The user can set the resolution in the export settings, and it is possible to export in low-resolution with less demand on computational resources, i.e. storage volume and processing time and with less chance of geometry failure.

Figure Figure 6.4 shows a concrete slab at the bottom of the *Mile End Shaft*. The blue lines represent the high-resolution B-Rep converted from *DWG*, the red and green lines show low-resolution B-Rep from *IFC* and *SKP*. The high-resolution B-Rep converted from *DWG* conforms to the original *DGN* model at the sub-millimetre level. The figure shows the concave walls of the shaft are represented by all three formats within a tolerance of 94 mm. This variation is comfortably less than the voxel resolution size of 0.25 m (Section 8.3.3.3) and the transformation of curved surfaces to B-Rep at this resolution is suitable for voxelisation. However, it may not be suitable for other purposes such as enclosing a watertight space.

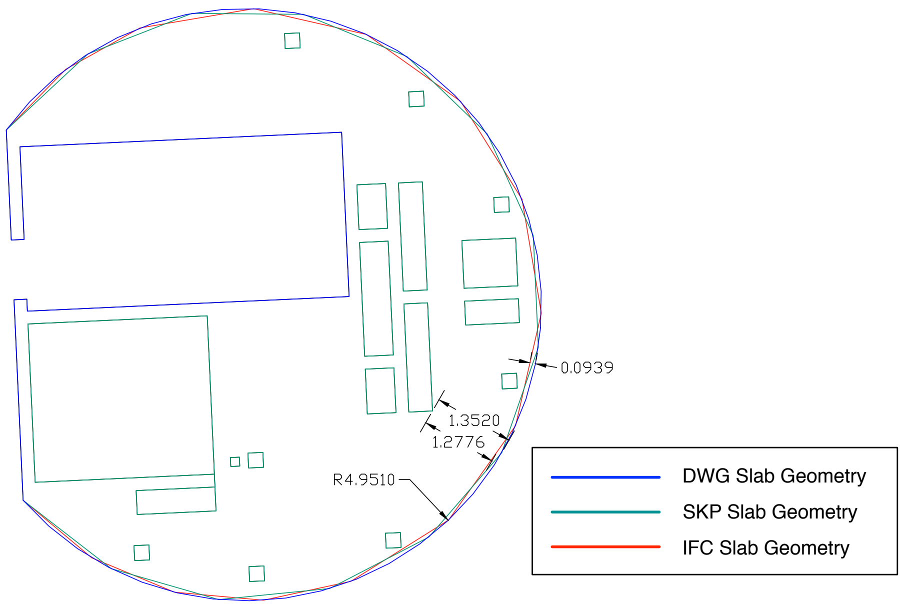


Figure 6.4: Dimensional comparison of B-Rep geometry

6.4 Assessment of Extraction and Transformation Methods

The findings in this section are summarised in Table 6.1 using columns that correspond to the criteria identified above. The first column lists the different export methods investigated and whether they can be read in *FME* is tabled in the second column. Two rows are included in the table for reading features from *IFC* with and without geometry. The next two columns cover the quality and reliability of the geometry exported. The column headed *Element ID* reports whether features are tagged with the *Element ID* used in *MicroStation*. The final column includes whether any additional information is attached to each feature corresponding to the fourth criterion. The last criterion regarding the ability to use the format in a batch process is not presented in the table as each format can be handled in a batch process by both *MicroStation* and *FME Workbench*.

Table 6.1: Comparison of MicroStation extraction methods

	Readable in FME	Geometry	Reliability	USID	Info
DGN	No	N/A	N/A	N/A	N/A
MVBA (CSV)	Yes	Bounding Box	Yes	Yes	Level & Const. Class
IFC	Yes	Bounding Box Low res B-Rep CSG Extruded 2D	Some geometry incorrect leading to incorrect bounding box	Yes	Semantic class BIM properties
DWG	Yes	High res B-Rep (split into parts)	Some geometry not watertight	Yes	No
SKP	Yes	Low res B-Rep (split into parts)	Some features missing	Level-renaming workaround Midpt-matching workaround	

From this analysis, it is evident that the above methods are not perfect. For this research, *IFC* is the most suitable medium for extracting information because it does not break the link between geometric and semantic information. However, the export does suffer from occasionally incorrect geometric representations, and some elements are lost altogether. On these occasions, it is possible to substitute the geometry from the *DWG* and *SKP*.

The final step of the ETL operation is to load the geometry to an *Oracle Spatial* database in the *SDO_GEOMETRY* format. The choice of destination is somewhat incidental to the ETL operation. As an alternative to using *Oracle Spatial*, the feature could just as easily be written as a *MultiPatch* object for use in an *Esri* application.

Although *Oracle Spatial* is a proprietary platform, the structure of the *SDO_GEOMETRY* format is well documented and openly accessible (Kothuri, et al. 2007). It is straightforward to write to and from *FME Workbench* and capable of being read from and written to directly using a *Python* script. *Oracle Spatial* also provides a toolbox of 3D spatial query tools such as *SDO_ANYINTERACT* or *SDO_INSIDE*.

6.5 Implementing the Extraction Workflow

Having investigated the various methods for exporting geometric and semantic information from the *MicroStation* platform, this section will describe a practical way for extracting elements, transforming them into a GIS-compliant geometry format and uploading the elements to a 3D GIS. The workflow described here will be used to prepare elements that will go on to be used for generating explicit 3D space geometry and for performing spatial joins in Chapter 8 and Chapter 9.

The workflow will read in CAD files created using the BIM extensions in *Bentley MicroStation* V8 Select Series 2. It will also read in a list of elements, identified using an *MVBA* script as described in Section 6.2.3, to act as a benchmark control list.

The workflow will use *IFC* as its principal source of geometric and semantic information. Where it is not possible to extract and transform an element in the benchmark control list using the *IFC* source, the workflow shall use the *DWG* export file as the first alternative, and *SKP* export file as the second alternative. Semantic information from *IFC* will be joined with the *DWG* or *SKP* geometry.

In the event that using *DWG* or *SKP* leads to elements being split into parts, *Part IDs* will be assigned and handled consistently (i.e. the *Part ID* refer to the same part regardless of source). When divided into parts, the complete set of elements shall instead be referred to as the set of features.

Elements/features are to be uploaded with valid solid geometry. If a valid solid cannot be transformed from any of the sources, a valid surface shall be uploaded in its place to a separate table. A full feature set of valid surfaces can be obtained from reading both the solid table and the surface table.

The workflow should ideally be automatic and batch-executable. Where this is not feasible, the workflow should handle a specific list of exceptional elements that have been manually identified for special treatment. With such a list, subsequent batch executions can be repeated automatically.

The workflow will be executed in two parts. In the first part, a batch control process (attached at Appendix G.1) will be run on a list of *DGN* model files within in *MicroStation*. This batch control process carries out a list of tasks described as follows. After opening a *DGN* file, it exports the model as an *IFC* file and a *DWG* file. It will then run an *MVBA* script (attached at Appendix G.2) which exports a benchmark control list to *CSV* (while also exporting information on the *DGN* level). The *MVBA* script also moves each element to a new level named in accordance with the *Element ID*. Having run the *MVBA* scripts, the batch control process will then export the model as a *SKP* file and then saves the model with renamed levels as a *DGN* file. This *DGN* with renamed levels is used for finding elements using *Element ID*.

For the second part of the workflow, *FME Workbench* will be used to merge information from all four sources using a fully developed workspace. An *FME* workspace is run for each *DGN* model file, reading the *CSV*, *IFC*, *DWG* and *SKP* files created earlier and uploading the element features to *Oracle Spatial*.

As well as uploading the element features, the *FME* workspace counts the elements and features being processed at each stage of the workspace and exports the count statistics as a *CSV* file. These counts enable the workflow to be quantitatively assessed by accounting for every element in the benchmark control list.

The location of these counting points in the workspace is illustrated in the flow diagrams that follow using an encircled letter and a list of letters used, along with an alias and short description is provided at the end of this chapter in Table 6.3. The results collected at each of these counting points is tabulated in Chapter 7.

An overview of the data flow in the *FME* workspace is illustrated as a flow diagram in Figure 6.5. In this flow diagram, the paths are colour-coded according to the information source. In Figure 6.6 to Figure 6.10, the flow diagram has been broken up into five parts to enable each part to be described more fully in the subsections that follow.

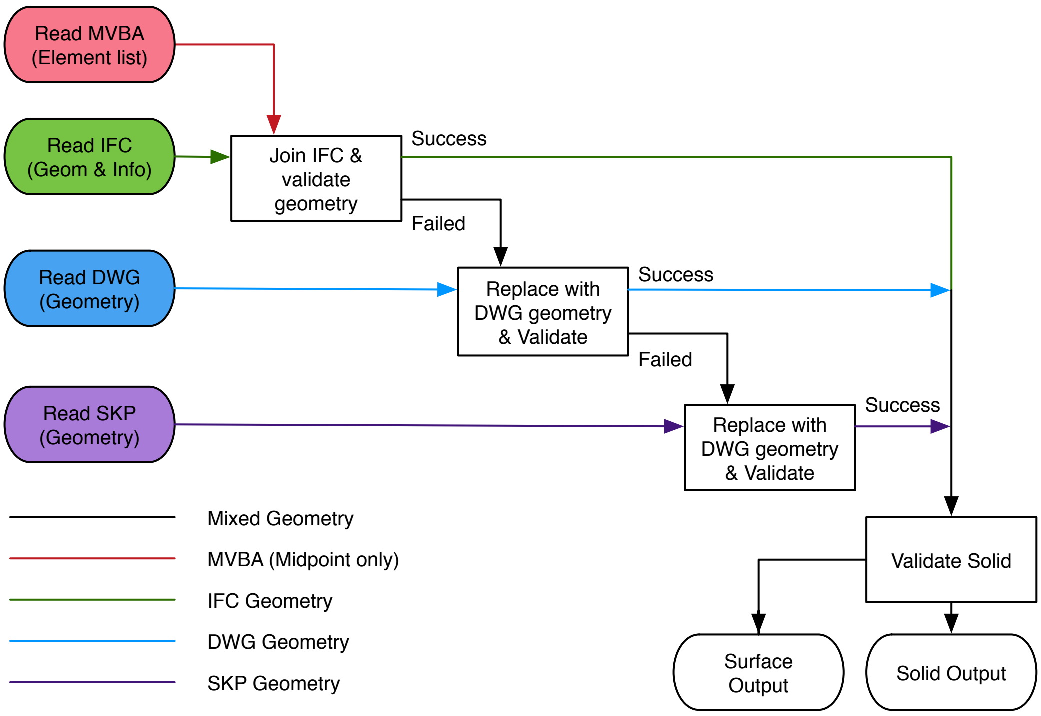


Figure 6.5: Skeleton workflow for merging element geometry

6.5.1 Reading MVBA Element List

Following the diagram in Figure 6.6, the workflow starts by reading a complete list of the elements that are contained in the *MicroStation DGN* model file. This list was generated using the *MVBA* script described in Section 6.2.3 and saved to a *CSV* file. The letter *A* is used to refer to the number of elements in this list.

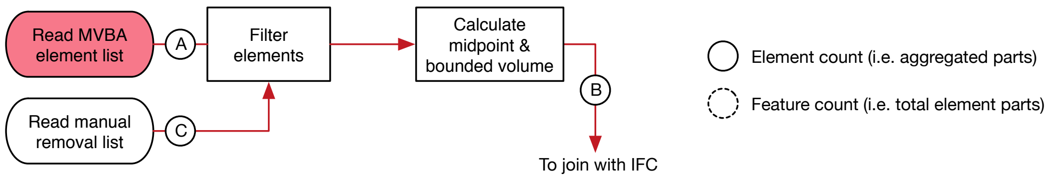


Figure 6.6: Workflow for merging element geometry - Part 1 - Reading MVBA element list

This complete list needs to be pruned down by filtering out construction elements that do not belong to the model. Construction elements should be tagged as such; however, if they are wrongly classified as primary elements, then certain assumptions can be used to filter them out, such as 2D elements located in the zero elevation plane and 1D elements such as construction lines. The information harvested using the *MVBA* script can therefore be used to filter out construction elements using the criteria in Table 6.2.

Depending on the quality of the model, there may still be some errant construction elements. These must be identified manually and removed by supplying a list of elements for removal in a *CSV* file. The letter *C* is used to refer to the number of elements manually designated for removal.

Table 6.2: ETL workflow filter criteria

Criteria	Condition
Construction class	0
Max Elevation	0m
Element Type	Surface & Solid
Manual Identification	False

The midpoint and bounding box volume of each element is calculated from the bounding box coordinates identified using the *MVBA* script. The midpoint can be used as a means for matching *SKP* geometry without an *Element ID*, and the bounding box volume will later be used as an integrity check.

The letter *B* represents the list of elements that constitute the model after removal of construction elements. This list shall act as the benchmark against which the list of outputted elements will be assessed.

6.5.2 Reading IFC

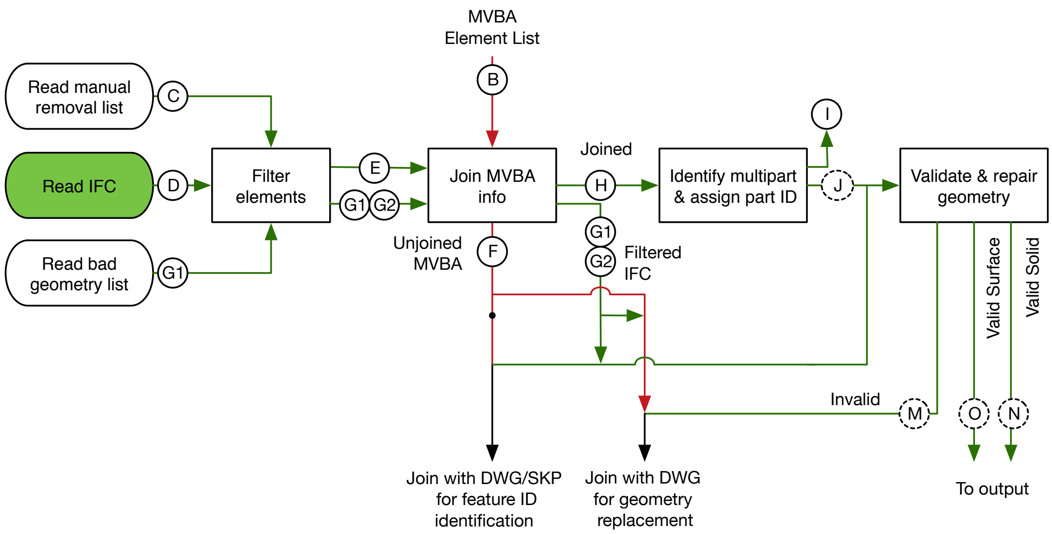


Figure 6.7: Workflow for merging element geometry - Part 2 - Reading IFC and joining MVBA

In Figure 6.7, the workflow continues to read the geometry and attributes of the collection elements exported from *MicroStation* as *IFC* files (*D*). Although each *IFC* element has its own IFC Globally Unique Identifier (GUID), it is necessary for consistency to identify each element according to its *MicroStation Element ID*, generally referred to here as just the *Element ID*. The *Element ID* of each *IFC* element can be extracted using a regular expression to parse the *Description* attribute.

The imported *IFC* elements (*D*) are filtered using the same criteria as Section 6.5.1, including the use of a manual list of construction elements (*C*). Upon inspecting the final model, it may come to the user's attention that some elements may have corrupted geometry, such as in Figure 6.2. These elements can be added to a *CSV* file (*G1*) so that the element is tagged with an instruction on subsequent runs for the element to acquire its geometry from an alternative source. In the same way, any element without geometry will be tagged to take up is geometry from an alternative source (*G1*).

Each element in the filtered *MVBA* list (*B*) is joined with its corresponding *IFC* element (*E+G1+G2*) using the *Element ID* as the primary key. The letter *H* is used to refer to these joined elements. If there any redundant elements that have erroneously made their way into the *IFC* file they will be dropped at this point. All elements from the *MVBA* not joined with elements from the *IFC* file (*F*) remain in the workflow and join (*G1, G2*) to acquire their geometry from alternative sources.

Element geometry occasionally contains multiple parts. Elements with multiple-parts (*J*) are de-aggregated to maintain consistency with *DWG* and *SKP* features. A unique primary key is created for each feature made up from the *Element ID* and a *Part ID* unique to that element-part. The letter *I* refers to the total number of features in the workflow. After splitting up elements into features, the midpoint and bounding box volume of each feature is calculated for matching with *DWG* and *SKP* sourced features.

The features based on the *MVBA* element list and merged with geometry and semantic information from *IFC* are tested using the *FME* Geometry Validator. The features are checked for a range of issues, including non-intersection and correct face orientation. Features that pass or can be repaired are processed for output either as valid solids (*O*) or surfaces (*N*). Features that fail (*M*), together with any pre-identified elements (*G1+G2*), are queued to have their geometry replaced, first from the *DWG* source, and then the *SKP* source.

6.5.3 Reading DWG

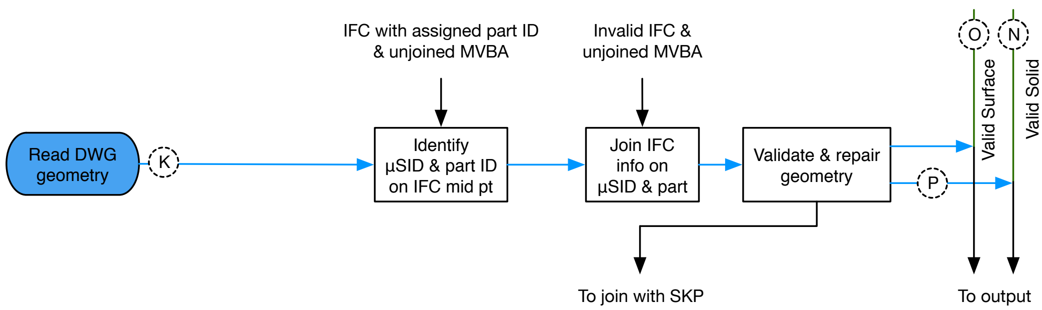


Figure 6.8: Workflow for merging element geometry - Part 3 - Reading and merging DWG

The next step, illustrated in Figure 6.8 is to read the features from the *DWG* file. As referred to above, the export of elements from *MicroStation* to *DWG* occasionally splits up elements into parts, and unique geometries read from *DWG* shall be referred to as features. Furthermore, the *DWG* may also contain errant construction features, and so these are filtered out, keeping only surface geometry and geometry with a maximum elevation greater than zero. The number of features read from *DWG* after filtration, is referred to using the letter *K*.

The *Element ID* can be extracted from the *DWG* feature using the method described in Section 6.2.4. This *Element ID* is the primary means for joining *DWG* features with the *IFC* and *MVBA* information. However, using the *Element ID* alone prevents the *DWG* feature being matched correctly with the de-aggregated *IFC* part. To achieve this, the midpoints and bounding volumes[2] are calculated for marrying up.

The features that have been selected from the *DWG* to provide alternative geometry are passed to the *FME GeometryValidator* transformer for validation. Validated and repaired solid features (*P*) and surface features are queued for output. The number of non-solid surfaces has not been collected as this exercise is primarily interested in solid features. Those features that fail are passed for matching with *SKP* sourced geometry.

6.5.4 Reading SKP

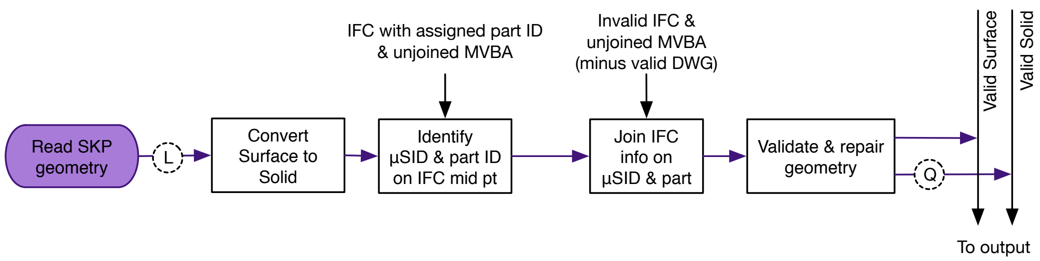


Figure 6.9: Workflow for merging element geometry - Part 4 - Reading and merging SKP

In the next part, as illustrated in Figure 6.9 features are read from the model exported via *Trimble SketchUp* (*SKP*). Like the elements exported via *DWG*, any multiple-part elements will be de-aggregated into constituent parts. It is, therefore, necessary to match up not just on *Element ID* but also on *Part ID*. Also, as with the *DWG* import, any elements that can be assumed to be construction elements based on geometry type and elevation can be removed. Once this is done, the total number of *SKP* features is referred to by the letter *L*

[2] If parts are concentrically located, the midpoint alone may be insufficient for matching.

Unlike *DWG*, the *SKP* format does not store the original *Element ID* when exported from *MicroStation*. The *Element ID* can, however, be maintained using the level-renaming workaround as described in Section 6.2.3. It should be noted that this workaround cannot always be relied upon so matching on midpoint and volume may need to be called upon.

It should be noted that there may be an occasion where the geometry for a multi-part element cannot be sourced from either *IFC* or *DWG*. In these cases, the *Part ID* of the *IFC* features will need to be resolved manually. The need to resolve *Element* and *Part IDs* manually has not arisen as part of this exercise.

As with the *DWG* features, once the features have been matched and used to replace geometry missing from the *IFC*/*MVBA* lists, the geometry can be validated as a solid (*Q*) and queued for output, or if not valid, queued for output as a surface.

6.5.5 Loading to Database

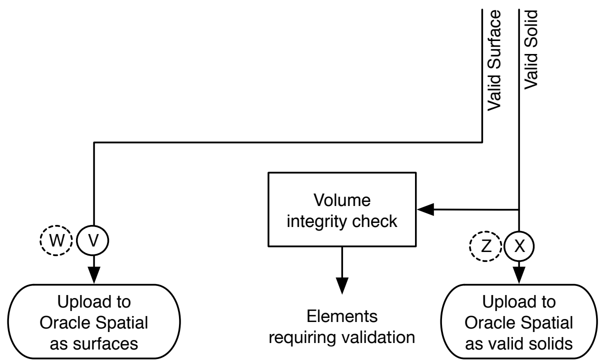


Figure 6.10: Workflow for merging element geometry - Part 5 - Loading

Once the geometry has been validated, repaired or replaced, a final integrity check is performed by comparing the bounding box volume of the current geometry with the bounding box volume calculated from the coordinates written by the *MVBA* script. If the volume of the final element differs from the volume calculated from the *MVBA* bounding box by more than 10 percent, then it is highlighted for manual validation.

Each feature is written to an *Oracle Spatial* database with its *Element ID* and its Part ID and the filename of the *DGN* model file. Together, these three fields act as a unique primary key for future identification. As well as these three fields, any other information such as class, attributes, properties and element relationships can be uploaded to the database.

The geometry of the feature is written as an *SDO_GEOMETRY* object which is stored in the geometry column of the database. The colour of the CAD element used in the original *MicroStation* file is also written to the database using the RGB format. This RGB data can be read and applied to the geometry in applications that subsequently use the features.

The final tallies of surface elements (V), surface features (W), solid elements(X) and solid features (Z) are counted prior to upload. These figures shall be used in the next Chapter 7 to assess the quality of the ETL operation for a selection of model files.

Table 6.3: Letters used in workflow diagrams

Letter	Alias	Description
A	MVBA Elements	Elements read from MVBA CSV file
B		MVBA Elements after filtering
C		Elements designated for manual removal
D	IFC Elements	Elements read from IFC file
E		Elements from IFC file after filtering
F	Unjoined Elements	Elements from MVBA Element list not joined with elements from IFC file
G1	Candidate Elements	Elements from IFC file that have been manually removed for matching with DWG/SKP features
G2	Candidate Elements	Elements from IFC file that have been automatically removed for matching with DWG/SKP features
E+G1+G2	Filtered IFC Elements	Elements from IFC file plus Candidate Elements
H		Elements from MVBA list that have been joined with IFC elements
H+G1+G2	Joined Elements	Elements from MVBA list that have been joined with IFC elements plus Candidate Elements
I	Joined Features	Features deaggregated from Joined Elements
J	Multi-part Elements	Joined Elements made up from multiple parts
K	DWG Features	Features read from DWG file
L	SKP Features	Features read from SKP file
M	Candidate Features	Joined Features identified as invalid
N	Valid Solid Features	Joined Features identified as valid Solid
O	Valid Surface Features	Joined Features identified as valid Surface
P	Selected DWG Features	Selected DWG Features identified as valid Solid
Q	Selected SKP Features	Selected SKP Features identified as valid Solid
R		Not used
S		Not used
T		Not used
U		Not used
V	Output Surface Elements	Total Elements identified as valid Surface
W	Output Surface Features	Total Features identified as valid Surface
X	Output Solid Elements	Total Elements identified as valid Solid
Y		Not used
Z	Output Solid Features	Total Features identified as valid Solid

7 Assessment of Extraction Transformation Loading Workflow

In the previous Chapter 6, the various methods for extracting semantic, geometric and attribute information of *MicroStation* Computer Aided Design (CAD) elements authored using the *Bentley MicroStation* Building Information Modelling (BIM) extensions were considered, and an Extract-Transform-Load (ETL) workflow covering the whole process from CAD to a Geographic Information System (GI System) was proposed.

In this chapter, the ETL workflow will be tested on nine *DGN* model files selected from the Crossrail federated CAD model. The selection, listed in Table 7.1 and illustrated in Appendix A, includes a range of structural, architectural and electrical model files authored by different contractors. These files will subsequently be used to provide the building elements for generating spaces and for providing Mechanical, Electrical and Plumbing (MEP) assets with which to perform spatial joins.

The first set includes three structural model files that form the *Broadgate Ticket Hall* complex at Liverpool Street Station. This set also contains two MEP model files forming part of the electrical installation in the ticket hall. The second set includes two structural model files that form part of the *Mile End Shaft* and two architectural models that contain doors and partition walls. The location of the *Broadgate Ticket Hall (BTH)* and *Mile End Shaft (MES)* are indicated on a schematic of the *Elizabeth Line* in Figure 1.1.

All the model files were read, transformed and uploaded as a single two-part batch. The first part of the batch was performed in *MicroStation*, the second part of the batch was performed in *FME Workbench* managed by a *Python* script. Running the whole process as a batch ensures all results have been consistently collected and proves that the workflow can be up-scaled. It should be noted that the process is not fully automatic as corrupted geometries must be manually identified and inputted to the workflow via a *CSV* file.

The batch process in *MicroStation* exports the MicroStation Drawing File (DGN) model files into three different formats: Industry Foundation Classes (IFC). MicroStation Drawing File (DWG) and SketchUp Model File (SKP). The number of elements in each model together with the file size of each format of each file in Table 7.1 is tabulated in Table 7.2.

Due to vagaries[1] in how *FME Workbench* handles the reading of different file types, the *Python* script must first copy the files associated with a single *DGN* model file into a single directory and rename each file to a consistent filename for the *FME* workspace to read. Once the files to be read have been collated, the *Python* script calls the *FME Workbench* to be run.

The *Python* script then reads the counts of the elements and features and collates the results into *LaTeX* tables. These results are tabled and discussed in Section 7.1 that follows.

Table 7.1: List of files tested by ETL operation

File	Type	Description
LPL-C-1-41051	Structural	Broadgate ticket hall ceiling
LPL-C-2-41052	Structural	Broadgate ticket hall and escalator descent
LPL-E-1-42201	Electrical	Broadgate ticket hall electrical fittings
LPL-E-2-42205	Electrical	Broadgate ticket hall electrical fittings
LPL-C4-00301	Structural	Liverpool Street station platform and access tunnels
MES-S-00004	Structural	Mile End Shaft bottom level
MES-S-00007	Structural	Mile End Shaft mid level
MES-A-2-00001	Architectural	Mile End Shaft bottom level non-structural walls and doors
MES-A-Z-31749	Architectural	Mile End Shaft stairway

[1] *FME* workspaces should in theory be able to run in batches by passing source filenames to a *workspace runner*, however this method did not work for all readers.

Table 7.2: File sizes of files tested by ETL operation

File	No. of Elements	DGN (kB)	IFC (kB)	DWG (kB)	SKP (kB)
LPL-C-1-41051	133	4700	966	426	2500
LPL-C-2-41052	199	12700	789	618	1800
LPL-C4-00301	642	48100	18300	-	27600
LPL-E-1-42201	372	2100	1100	670	3600
LPL-E-2-42205	477	4100	1500	604	5500
MES-S-00004	31	812	185	347	642
MES-S-00007	24	694	295	126	618
MES-A-2-00001	18	925	77	99	123
MES-A-Z-31749	92	2300	902	279	7000

7.1 ETL Operation Results

7.1.1 Reading MVBA Element List

The first part of the workflow, described in Section 6.5.1, reads in the list of elements generated using the *MicroStation Visual Basic Application (MVBA)* script and filters out construction elements using criteria and a manual list. The number of elements counted read in (*A*) and remaining after filtering (*B*) are presented in Table 7.3 and Table 7.4. Also included are numbers on how the elements were filtered out. The percentages, calculated from the top row (*A*), show that each file generally contains more elements than just those that make up the final model.

It would appear that the criteria in Table 6.2 are sufficient for filtering the list of elements. However, in the case of one model, *LPL-C4-00301*, there are 13 supernumerary elements scattered across the model, as illustrated in Figure 7.1. These elements had to be identified by visual inspection, and their *Element IDs* added to a *CSV* file for automatic removal during subsequent runs.

The bottom row (*B*) of the tables shows the number of elements in each file that will be used as a benchmark to provide an objective assessment of the ETL operation.

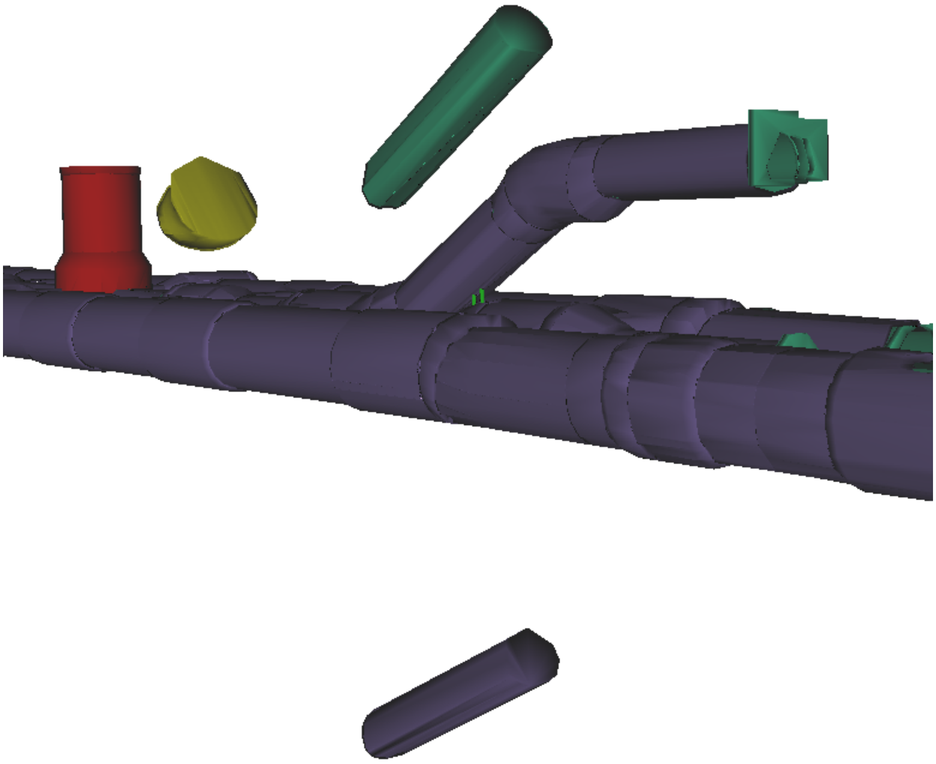


Figure 7.1: Supernumerary elements present in LPL-C4-00301

Table 7.3: Count of filtered MVBA elements (Broadgate)

	LPL-C-1-41052	LPL-C-2-41051	LPL-C4-00301	LPL-E-1-42201	LPL-E-2-42205
Elements in MVBA file (A)	199 (100%)	133 (100%)	642 (100%)	372 (100%)	477 (100%)
Elements not surface/solid	17 (8%)	5 (3%)	256 (39%)	2 (0%)	8 (1%)
Elements with construction tag	1 (0%)	1 (0%)	37 (5%)	0 (0%)	0 (0%)
Elements with zero elevation	21 (10%)	9 (6%)	0 (0%)	0 (0%)	0 (0%)
Elements manually removed (C)	0 (0%)	0 (0%)	13 (2%)	0 (0%)	0 (0%)
Filtered MVBA elements (B)	160 (80%)	118 (88%)	336 (52%)	370 (99%)	469 (98%)

Table 7.4: Count of filtered MVBA elements (Mile End Shaft)

	MES-S-00004	MES-S-00007	MES-A-2-00001	MES-A-Z-31749
Elements in MVBA file (A)	31 (100%)	24 (100%)	18 (100%)	92 (100%)
Elements not surface/solid	0 (0%)	1 (4%)	1 (5%)	16 (17%)
Elements with construction tag	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Elements with zero elevation	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Elements manually removed (C)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Filtered MVBA elements (B)	31 (100%)	23 (95%)	17 (94%)	76 (82%)

7.1.2 Reading IFC Elements

The next part of the workflow, described in Section 6.5.2, reads in the elements from ETL files (*B*) and filters out construction elements, leaving usable model elements (*E*) and elements with corrupted (*G1*) or missing geometry (*G2*). The numbers of elements for each file are shown in Table 7.5 and Table 7.6. The percentages are calculated from the top row (*D*).

On reading the ETL files, two elements overall were found to have null geometry (*G2*), and one element is manually identified as having deformed geometry (*G1*). The geometry of these elements was subsequently substituted using geometry from alternative sources.

It should be noted that just as 13 elements were removed from the list of *MVBA* elements, the same 13 elements are removed from the list of ETL elements (noting that two elements have already been filtered by other criteria).

The bottom row of the tables (*E+G1+G2*) presents the number of filtered IFC elements that will be used as a benchmarking for assessing the quality of the ETL operation.

Table 7.5: Count of filtered IFC elements (Broadgate)

	LPL-C-1-41052	LPL-C-2-41051	LPL-C4-00301	LPL-E-1-42201	LPL-E-2-42205
Elements in IFC file (D)	162 (100%)	119 (100%)	403 (100%)	370 (100%)	475 (100%)
Elements not surface/solid/null	1 (0%)	0 (0%)	8 (1%)	0 (0%)	6 (1%)
Elements with zero elevation	0 (0%)	0 (0%)	2 (0%)	0 (0%)	0 (0%)
Elements manually removed (C)	0 (0%)	0 (0%)	11 (2%)	0 (0%)	0 (0%)
Feat. manually identified (G1)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Features with null geom. (G2)	0 (0%)	1 (0%)	0 (0%)	0 (0%)	0 (0%)
Filtered IFC elem. (E+G1+G2)	161 (99%)	119 (100%)	382 (94%)	370 (100%)	469 (98%)

Table 7.6: Count of filtered IFC elements (Mile End Shaft)

	MES-S-00004	MES-S-00007	MES-A-2-00001	MES-A-Z-31749
Elements in IFC file (D)	31 (100%)	23 (100%)	17 (100%)	73 (100%)
Elements not surface/solid/null	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Elements with zero elevation	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Elements manually removed (C)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Feat. manually identified (G1)	1 (3%)	0 (0%)	0 (0%)	0 (0%)
Features with null geom. (G2)	0 (0%)	1 (4%)	0 (0%)	0 (0%)
Filtered IFC elem. (E+G1+G2)	31 (100%)	23 (100%)	17 (100%)	73 (100%)

7.1.3 Merging with IFC

Having loaded the *MVBA* list of elements and the ETL elements, the two sets of elements are joined using their *Element ID*, as also described in Section 6.5.2. The numbers counted in this part, together with percentages calculated from the top row (*B*), are presented in Table 7.7 and Table 7.8. Elements with deformed or null geometry (*G1+G2*) are included in this table of elements.

Within the *Broadgate Ticket Hall* model files, all of the *MVBA* elements are present in the ETL files, but there are 48 elements in the ETL files that are not in the *MVBA* list. However, within the *Mile End Shaft* model files, three elements in *MES-A-Z-31749* are present in the *MVBA* list but not in the ETL file. Manual inspection of the model reveals that these are elements should have been tagged as construction elements in *MicroStation* but were not.

Table 7.7: Count of merged elements (Broadgate)

	LPL-C-1-41052	LPL-C-2-41051	LPL-C4-00301	LPL-E-1-42201	LPL-E-2-42205
Filtered MVBA elements (B)	160 (100%)	118 (100%)	336 (100%)	370 (100%)	469 (100%)
Filtered IFC elem. (E+G1+G2)	161 (100%)	119 (100%)	382 (100%)	370 (100%)	469 (100%)
Joined elements (H+G1+G2)	160 (100%)	118 (100%)	336 (100%)	370 (100%)	469 (100%)
MVBA elements not joined (F)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
IFC elements not joined	1 (0%)	1 (0%)	46 (13%)	0 (0%)	0 (0%)

Table 7.8: Count of merged elements (Mile End Shaft)

	MES-S-00004	MES-S-00007	MES-A-2-00001	MES-A-Z-31749
Filtered MVBA elements (B)	31 (100%)	23 (100%)	17 (100%)	76 (100%)
Filtered IFC elem. (E+G1+G2)	31 (100%)	23 (100%)	17 (100%)	73 (100%)
Joined elements (H+G1+G2)	31 (100%)	23 (100%)	17 (100%)	73 (96%)
MVBA elements not joined (F)	0 (0%)	0 (0%)	0 (0%)	3 (3%)
IFC elements not joined	0 (0%)	0 (0%)	0 (0%)	0 (0%)

7.1.4 Splitting Multi-part Elements into Single-part Features

The final part of the step described in Section 6.5.2 is to split multi-part elements (*J*) into parts. The total number of features (*I*) is expressed as a percentage of the total number of elements (*H*) as presented in Table 7.9 and Table 7.10. These tables do not include elements with deformed or null geometry (*G1+G2*) as there is no way of determining whether these elements have multiple parts.

On the whole, it can be observed that only a small number of elements in structural and architectural model files contain multiple parts. However, one MEP model file, *LPL-E-2-42205*, contains 21 elements that are made up of 119 parts.

Table 7.9: Count of merged features (Broadgate)

	LPL-C-1-41052	LPL-C-2-41051	LPL-C4-00301	LPL-E-1-42201	LPL-E-2-42205
Joined elements (H)	160 (100%)	117 (100%)	336 (100%)	370 (100%)	469 (100%)
Multi-part elements (J)	3 (1%)	1 (0%)	0 (0%)	0 (0%)	21 (4%)
Additional parts	3 (1%)	1 (0%)	0 (0%)	0 (0%)	98 (20%)
Joined features (I)	163 (101%)	118 (100%)	336 (100%)	370 (100%)	567 (120%)

Table 7.10: Count of merged features (Mile End Shaft)

	MES-S-00004	MES-S-00007	MES-A-2-00001	MES-A-Z-31749
Joined elements (H)	30 (100%)	22 (100%)	17 (100%)	73 (100%)
Multi-part elements (J)	1 (3%)	0 (0%)	7 (41%)	0 (0%)
Additional parts	1 (3%)	0 (0%)	7 (41%)	0 (0%)
Joined features (I)	31 (103%)	22 (100%)	24 (141%)	73 (100%)

7.1.5 Geometry Validation and Replacement

The element features, read in Section 6.5.2, are tested using the *FME GeometryValidator* transformer. The number of features (*I*) counted in the validation process are presented in Table 7.11 and Table 7.12 with the percentages being calculated from the top row (*I*).

The tables show the features outputted as solids (*N*), those features that fail validation (*M*), of which some (*O*) can also be outputted as surfaces. Where a feature fails validation as a watertight solid, an attempt will be made to replace it with features from *DWG* (*P*) and *SKP* (*Q*).

While it is desirable to upload all features as watertight solids, it should be noted that if this condition is not enforced, then 100% of all features joined between the *MVBA* list and the ETL model (*I*) are capable of being uploaded to the spatial database (*W*). With some features, it has been necessary to replace geometry from alternative sources; for example, the deformed geometry in *MES-S-00004*, as shown in Figure 6.2, has been replaced by geometry sourced from the *DWG* model; for example, the element with null geometry (which should have been a multi-part element) in *MES-S-00007*, has been reinstated with two features from the *DWG* model.

Focussing on valid solid features, the 19 features failing validation as a solid in *MES-A-Z-31749*, have been replaced by ten features from *DWG* and nine features from *SKP*. When working on the curved tunnel surfaces in *LPL-C4-00301*, 27 out of 51 failed solid features have been reinstated with the result that 91 percent of all original elements can be uploaded to the spatial database as valid solids. However, the success rate when the workflow is applied to MEP entities, such as *LPL-E-1-42201* and *LPL-E-2-42205*, is much lower, with approximately only half of all features being uploaded as valid solids.

Table 7.11: Count of validated and replaced features (Broadgate)

	LPL-C-1-41052	LPL-C-2-41051	LPL-C4-00301	LPL-E-1-42201	LPL-E-2-42205
Joined features (I)	163 (100%)	118 (100%)	336 (100%)	370 (100%)	567 (100%)
Joined bad features (G1+G2)	0 (0%)	1 (0%)	0 (0%)	0 (0%)	0 (0%)
Features valid/repaired (N)	155 (95%)	117 (99%)	281 (83%)	130 (35%)	433 (76%)
Features failing validation (M)	8 (4%)	1 (0%)	51 (15%)	131 (35%)	4 (0%)
Surface Features (O)	0 (0%)	0 (0%)	4 (1%)	109 (29%)	130 (22%)
Features replaced by DWG (P)	2 (1%)	1 (0%)	0 (0%)	0 (0%)	0 (0%)
Features replaced by SKP (Q)	0 (0%)	1 (0%)	27 (8%)	0 (0%)	4 (0%)
Surface/Solid Features out (W)	163 (100%)	119 (100%)	336 (100%)	370 (100%)	567 (100%)
Solid Features out (Z)	157 (96%)	119 (100%)	308 (91%)	130 (35%)	437 (77%)

Table 7.12: Count of validated and replaced features (Mile End Shaft)

	MES-S-00004	MES-S-00007	MES-A-2-00001	MES-A-Z-31749
Joined features (I)	31 (100%)	22 (100%)	24 (100%)	73 (100%)
Joined bad features (G1+G2)	1 (3%)	1 (4%)	0 (0%)	0 (0%)
Features valid/repaired (N)	31 (100%)	22 (100%)	24 (100%)	54 (73%)
Features failing validation (M)	0 (0%)	0 (0%)	0 (0%)	19 (26%)
Surface Features (O)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Features replaced by DWG (P)	1 (3%)	2 (9%)	0 (0%)	10 (13%)
Features replaced by SKP (Q)	0 (0%)	0 (0%)	0 (0%)	9 (12%)
Surface/Solid Features out (W)	32 (103%)	24 (109%)	24 (100%)	73 (100%)
Solid Features out (Z)	32 (103%)	24 (109%)	24 (100%)	73 (100%)

The final two tables, Table 7.13 and Table 7.14 show the same results as the previous two tables, but the figures have been populated in terms of multi-part elements instead of features.

Table 7.13: Count of uploaded elements (Broadgate)

	LPL-C-1-41052	LPL-C-2-41051	LPL-C4-00301	LPL-E-1-42201	LPL-E-2-42205
Filtered MVBA elements (B)	160 (100%)	118 (100%)	336 (100%)	370 (100%)	469 (100%)
Joined elements (H+G1+G2)	160 (100%)	118 (100%)	336 (100%)	370 (100%)	469 (100%)
Surface/Solid elements out (V)	160 (100%)	118 (100%)	336 (100%)	370 (100%)	469 (100%)
Solid elements out (X)	156 (97%)	118 (100%)	308 (91%)	130 (35%)	339 (72%)

Table 7.14: Count of uploaded elements (Mile End Shaft)

	MES-S-00004	MES-S-00007	MES-A-2-00001	MES-A-Z-31749
Filtered MVBA elements (B)	31 (100%)	23 (100%)	17 (100%)	76 (100%)
Joined elements (H+G1+G2)	31 (100%)	23 (100%)	17 (100%)	73 (96%)
Surface/Solid elements out (V)	31 (100%)	23 (100%)	17 (100%)	73 (96%)
Solid elements out (X)	31 (100%)	23 (100%)	17 (100%)	73 (96%)

7.1.6 Summary

Overall the results of the ETL operation show it is possible to count a full set of elements contained within a *MicroStation DGN* model file, format all those elements into Boundary Representation (B-Rep) surfaces, and upload all those elements (broken down into features) into a *Geospatial Information Platform*, in this case, *Oracle*. The exercise has shown that the *MicroStation* export functions are not 100 percent reliable; furthermore, in three out of the nine model files, it has been necessary to replace geometry from an alternative source. These geometry export issues raise an important question as to how trustworthy is data that has been sourced from a different format. The importance of quality control measures in safety-critical use cases cannot be overstated.

The more challenging requirement has been to upload geometry as watertight solids. To meet this requirement, it has been necessary for seven out of nine model files to look to the *DWG* or *SKP* model files to replace geometry. While the number of watertight features found in the structural and architectural files is generally above 90 percent, the figure falls to 35 percent in the case one of the MEP model files.

The purpose of this chapter has been two-fold. Firstly, The ETL operation completes the first prerequisite step (PRS 1) in preparing the CAD elements for the asset linking method proposed in Chapter 2. These elements extracted and uploaded in this chapter will now be used to create interior spaces (PRS 2) in Chapter 8 and perform spatial joins (PRS 3) in Chapter 9. At the same time, the work carried out has been a valuable case study into the overcoming the challenges that stand in the way of seamless integration of BIM and Geographic Information Systems (GIS); more on this will be discussed in Chapter 10.

8 Creating Spaces for Asset Management

The workflow for linking Computer Aided Design (CAD) elements with AIMS assets, as initially described in Section 2.2, identified the need for three prerequisite steps; firstly the extraction and transformation of CAD elements into a Three-dimensional (3D) Geographic Information System (GI System) (PRS 1), secondly the creation of watertight spaces (PRS 2) and then the spatial join of those created spaces with the extracted CAD elements (PRS 3). Following the work described in Chapter 6 and Chapter 7, the Extract-Transform-Load (ETL) of building elements and Mechanical, Electrical and Plumbing (MEP) assets is now complete. The next stage in this workflow is to create watertight spaces fit for performing spatial queries.

In Section 3.10, six different methods for generating explicit space geometry were identified from the literature, these being:

a. Floor plan extrusion

b. Boolean difference

c. Topological reconstruction

d. Surface pairing

e. Cell and Portal Graph (CPG) analysis

f. Watershed segmentation*

In this chapter, the practical application of creating spaces using three of these methods - *floor plan extrusion*, *boolean difference* and *watershed segmentation* - will be investigated to assess their suitability for performing spatial queries. *CPG analysis* will not be considered as it is the manual precursor to the *watershed segmentation* method, nor will *topological reconstruction* and *surface pairing* be considered as these methods require spaces to be fully enclosed by building elements (i.e. walls, floors and ceilings) with all openings between spaces being filled with doors and windows. During the period of this investigation, the Crossrail CAD model files were not consistently available to this standard of design.

The methodology in Section 8.1, will consider each of the three methods and then propose how they might be implemented. The three methods will be used to create spaces, and the effectiveness of each technique will be commented on in Section 8.3. These spaces will then be selected for testing in Chapter 9 to consider their suitability for performing spatial joins.

8.1 Space Generation Methodology

8.1.1 Floor Plan Extrusion

The most straightforward approach involves the extrusion of two-dimensional (2D) floor plans to form 3D spaces. The method relies on the creation of floor plans by architectural design contractors as required under the Crossrail contract. These floor plans were delivered as non-geospatial 2D CAD files, and the Crossrail Geographic Information Systems (GIS) team converted the line drawings into spatially referenced features. As part of their production, each space was attributed with a floor elevation and ceiling height generically based on its level.

It is proposed to use *FME Workbench* to extrude the 2D polygons into 3D spaces according to their ceiling heights. The spaces will then be uploaded into *Oracle Spatial* where they are ready to be assessed on their suitability for performing spatial joins.

8.1.2 Boolean Difference

The second approach uses *boolean difference* to create spaces from their bounding elements. For this method, the building elements that enclose the space(s) are subtracted from a larger solid template profile. If all the spaces are perfectly enclosed by watertight elements, then the operation will output a collection of spaces that are located within the solid template profile. However, this method relies on the elements extracted in Chapter 7 to be uploaded as solid features.

It is proposed to create a solid template profile that will enclose the spaces being created. An *FME* workspace will read in the solid template profile and the building elements and use the *FME Clipper* transformer to remove segments from the profile to reveal the enclosed spaces. The resulting collection will be de-aggregated to form individual spaces and each space will be tested using the *FME GeometryValidator* transformer before being uploaded to *Oracle Spatial* to be used for spatial joins.

If the spaces are not fully enclosed, the *boolean difference* approach provides no benefit when applied to a generic template as the resulting output will be a large contiguous space that is not split up into individual spaces. The implicit spaces in the model files are not fully enclosed, as can be seen from a plan of the *Broadgate Ticket Hall* spaces in Figure 8.17; there are small openings in the walls and floors and large portals that link the spaces. It may, however, be possible to overcome this by using a combined approach that will be described in Section 8.1.4.

8.1.3 Watershed Segmentation

The final approach uses the *watershed transform* to segment the wider space into individual spaces. This technique is particularly useful for closing off the openings and portals that exist between spaces as depicted in Figure 8.17.

Haumont, et al. (2003) has demonstrated that *watershed segmentation* can be used to identify spaces in an architectural scene to construct CPGs. In this chapter, it is proposed to investigate whether it is possible to adapt the *watershed transform* as a means of creating spaces to be used in asset management, in particular as a means of creating watertight spaces for performing spatial joins.

While Haumont, et al. (2003) has demonstrated the ability of to create portals between spaces, it is not clear from the paper whether their Graphics Processing Unit (GPU) assisted method is capable of creating watertight spaces. Therefore, despite the advantages in faster computational time claimed by the method advocated by Haumont, et al. (2003), it is proposed to use the more straightforward mathematical morphological method for the work in this chapter as the voxel-based output is guaranteed to be watertight. An open-source implementation of the *watershed transform* is distributed as a function within the *Scikit-Image Python* package (see Section 8.2 for a list of *Python* packages implemented) and it is proposed to use this code, as it is readily available and works in conjunction with the *Python numpy* package.

A program will be written in *Python* to implement the *Scikit-Image* watershed function as described in Section 8.2. Once written, this program will be used to create spaces which will be compared with the spaces created using the *floor plan extrusion* and *boolean difference* methods.

8.1.4 Combined Approach

In addition to the above methods, it is also proposed to perform a three-step operation involving a combination of the above methods. The first step involves performing a *watershed segmentation* output, as described in Section 8.1.3; the second step dilates the output by one-voxel (Section 8.2.11); the third step cuts the dilated output using the geometric features as described in Section 8.1.2. This combined method will use one approach to segment the volume into individual spaces and the other approach to produce a high-quality representation of the space.

8.2 Python Implementation of Watershed Transform

In order to use the *watershed transform* to create spaces, a *Python* program will be written that will read in building elements from *Oracle Spatial* and prepare them following the flow diagram in Figure 8.1. It will then process the output of the *watershed transform* and upload the spaces back to the database as watertight B-Rep solid geometry objects.

The *Python* program will take advantage of the *Python* packages listed in Section 8.2, specifically implementing the *watershed transform* function distributed as part of the *Scikit-Image* package. This function takes three inputs, a list of source markers that will be used to seed the algorithm, a voxel array of inverted distance values and a mask array of voxels that do not belong to the spaces (i.e. the building elements and exterior space). The output of the function is a voxel array of values that correspond to the seeds from which the segmented space is grown.

The remainder of this section will describe how the distance field is generated and how the locations of the seed voxels are identified before being inputted to the distance transform function. It will also explain how the output array is handled to generate a B-Rep mesh of the interior spaces.

Table 8.1: Python packages used to perform watershed segmentation

Package	Version	License	Provides	Citation
Python	3.6.7	PSF		Python Software Foundation (2020)
Cython	0.29.10	Apache 2.0	Compiles typed Python code	Cython Community (2020)
NumPy	1.16.2	BSD	Data structures for voxel arrays	Harris, et al. (2020)
SciPy	1.3.0	BSD	Euclidean distance tool	Virtanen, et al. (2020)
scikit-image	0.15.0	BSD	Watershed transformation	van der Walt, et al. (2014)
pythonOCC	0.18.2	LGPL 3.0	Surface and solid object classes Interactive visualisation	Paviot (2019)
Qt	5.6	LGPL 3.0	GUI for pythonocc	Qt Company (2020)
PyQt	5.6.2	GPL 3.0	Bindings for Qt	Riverbank Computing (2020)
trimesh	2.38.42	MIT	Tools for voxelising meshes	Dawson-Haggerty (2020)
NetworkX	2.3.0	BSD	Graph tools	NetworkX Developers (2020)
python-igraph	0.7.1	GPL 2.0	Graph tools	igraph Core Team (2020)
Rtree	0.8.3	MIT	Spatial indexing	Gillies (2020)
Shapely	1.6.4	BSD	Manipulation of 2D features	Gillies (2020a)
cx_Oracle	7.1.3	BSD	Read/Write Access to Oracle RDBMS	Oracle Corporation (2020)

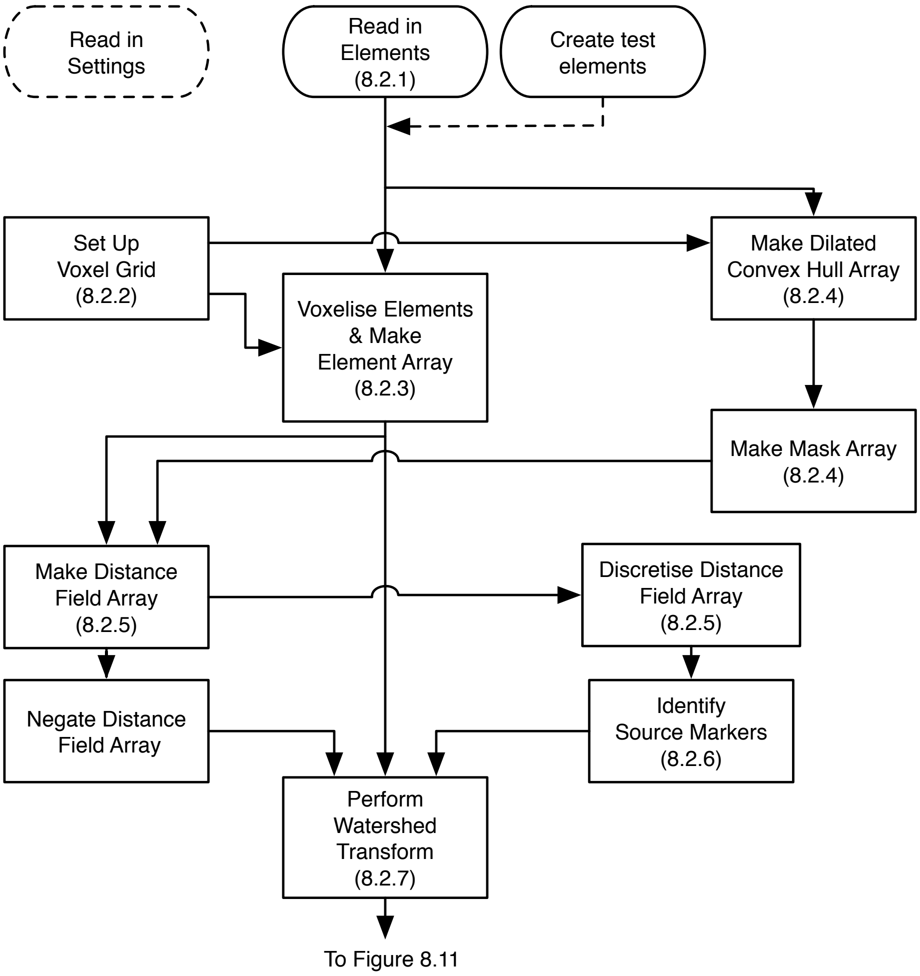


Figure 8.1: Watershed segmentation workflow - Part 1

The workflow will be explained here with the aid of a simple three room model, as illustrated in Figure 8.2. The model consists of four exterior walls, three interior walls, a floor and a ceiling, all enclosing three interior spaces. Each space is connected to the other spaces by an opening in the interior walls. The ceiling has been removed in Figure 8.2 to aid visibility.

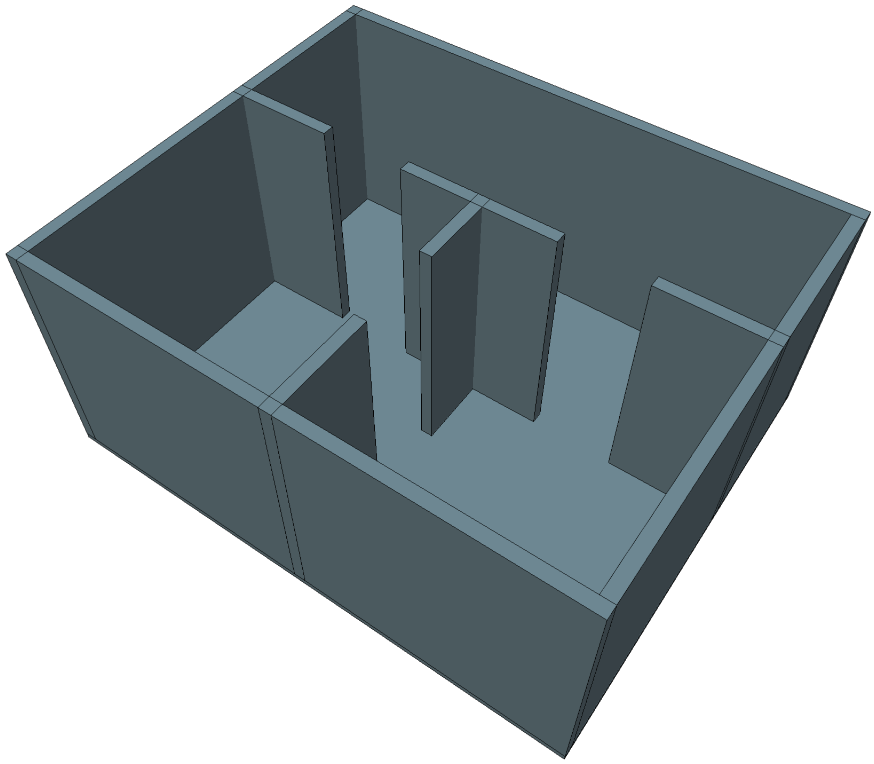


Figure 8.2: Three room example model

8.2.1 Reading in Building Elements

The building elements that enclose the spaces are stored in an *Oracle Spatial* database. The elements are read from the database using the *cx_Oracle* package. A script has been written to read the *Element Information Array* and *Ordinate Array* of the *SDO_GEOMETRY* format and convert them into a planar meshes which will be stored as a *TopoDS_Shell* object in *OpenCascade* (using *pythonocc* as a wrapper). Each *TopoDS_Shell* object is stored in a *Python* dictionary referenced using the object hash as a key. Additional information such as the type of element and its colour are stored alongside the *TopoDS_Shell* object.

8.2.2 Setting Up the Voxel Grid

An array of voxels is set up that extends over the minimum bounding volume of the building elements with a specified margin. The voxel axes are by default aligned with the Cartesian axes of the geometry being read. Alternatively, the axes orientation can be specified to be in general alignment with the building elements.

A transformation matrix is calculated and stored as a *gp_Trsf* object in *OpenCascade* . This transformation will be used to transform the *TopoDS_Shell* elements into a bespoke Coordinate Reference System (CRS) that is based on the origin and axes of the voxel array.

For the simple model in Figure 8.2, a small 3D voxel array has been created that encloses the features surrounded by a two voxel margin.

8.2.3 Voxelisation of Elements

This implementation will use the *trimesh* package to voxelise the geometry of the building elements. This package uses the *subdivision* method (Fei, et al. 2012) to convert a mesh of triangular faces into voxels.

This *subdivision* method differs from the classic method used in computational geometry (Cohen-Or and Kaufman 1995). The classic method takes each face of the mesh in turn and identifies the axial plane with which each face is most closely aligned. The face is then pixelated in this 2D plane and the coordinate on the orthogonal axis is calculated using the planar equation (Cohen-Or and Kaufman 1995). However, the voxels generated using the classic method do not fully cover the surface mesh.

The *subdivision* method also works through each face of the triangular mesh. Each face is iteratively subdivided until the longest edge of the face is less than a specified maximum length, as illustrated in Figure 8.3. The voxel locations are generated from the set of rounded vertex coordinates taken from the subdivided faces.

In Figure 8.3, a single 2D triangular face is converted into unit sized voxels. In this example which uses an edge factor of 2.0, the algorithm iteratively subdivides the triangular face until the longest edge is less than a maximum edge length of 0.5 (i.e. the inverse of edge factor). In Figure 8.3a only three voxels have been identified from the vertices of the triangular face prior to subdivision. At this stage, the longest edge is 7.13 units.

In Figure 8.3b, the face is subdivided into four new triangles, thus identifying three more voxels and reducing the longest edge to 3.56 units. Repeating again in Figure 8.3c reduces the edge length to 1.78 units.

In Figure 8.3d, all the edges are now less than 1.0 units and the voxelisation of the face has now filled out although there are parts of the triangle that are not represented by a voxel. By the fourth subdivision in Figure 8.3e, the longest edge has been reduced to less than 0.5, leading to better voxel coverage. However, looking closely at Figure 8.3f, although 99.96 percent of the face is represented by a voxel, there are still portions that have not been voxelised. Reducing the edge factor improves the voxel coverage at the expense of computational time. From Figure 8.3e shows that an edge factor of 2.0 will delivers satisfactory results.

The *subdivision* method only identifies voxels from the surface mesh and does not work for voxels located inside building elements. The interior voxels can be identified from the convex hull of the building element, although this method is not perfect as it can generate false positives if there are interior holes or other complex intrusions. If this is important, further research into voxelisation methods relevant to solid geometries will need to be made.

The voxelised elements are added in turn to create a single boolean element array representing the building elements that enclose the interior spaces within them.

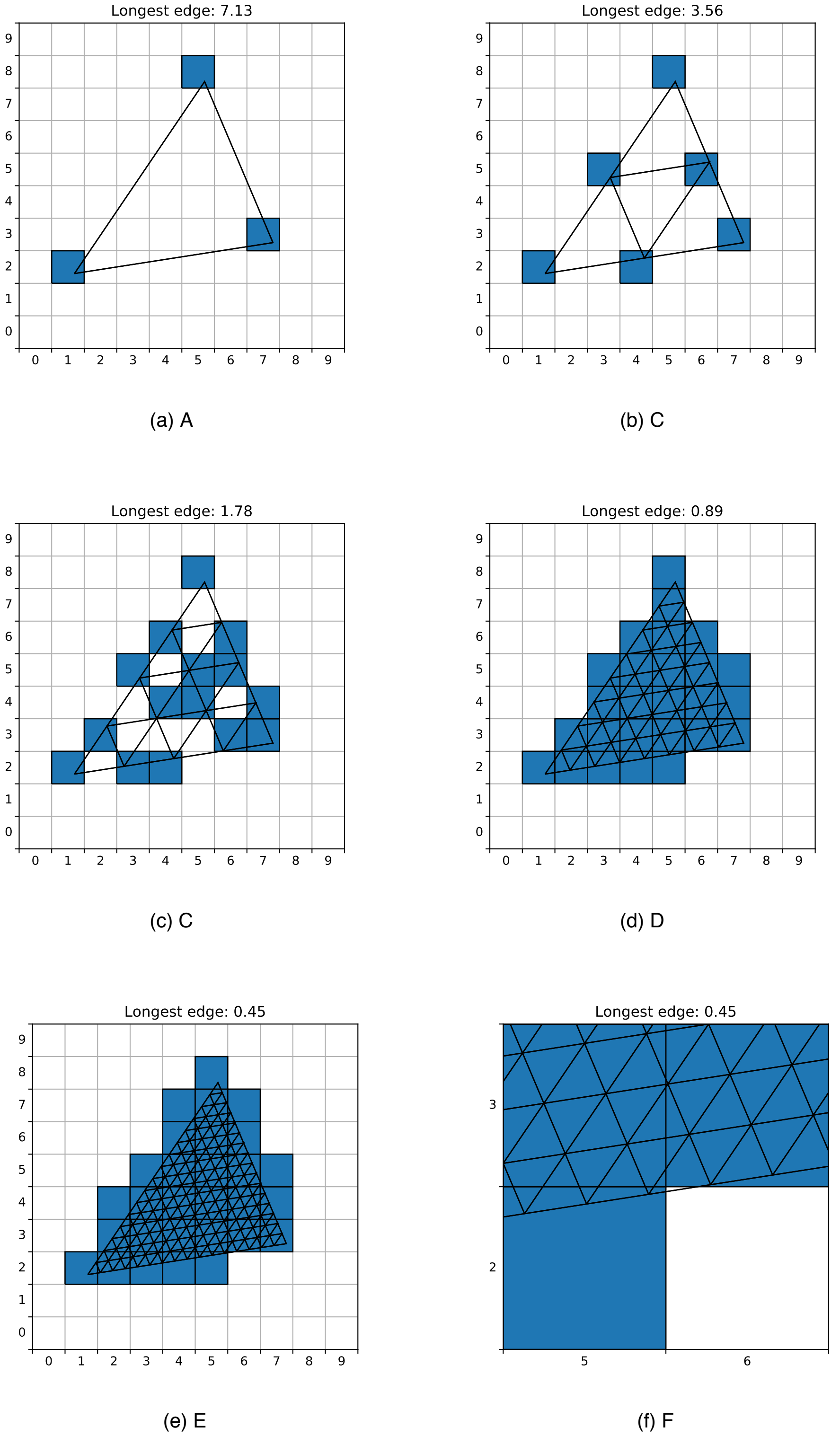


Figure 8.3: Voxelisation by subdivision

8.2.4 Making a Dilated Convex Hull Mask

The building element vertices are converted to a convex hull using the hull function of the *scipy* package. This surface mesh is voxelised using the same method as Section 8.2.3, with all interior voxels filled in. The hull voxel array is then dilated by a specified number of voxels. A mask array is created from all voxels in the array that do not belong to the dilated hull. The buffer region is important for the *watershed transform* to work on openings near the boundary of the convex hull.

Figure 8.4 is a plan view of the simple model illustrated in Figure 8.2. It shows a 2D plan of the voxel array, the building elements in black and their voxel representations in light blue. The mask array, created from a hull array dilated by one voxel, is shown as the hatched area along the top edge of the array.

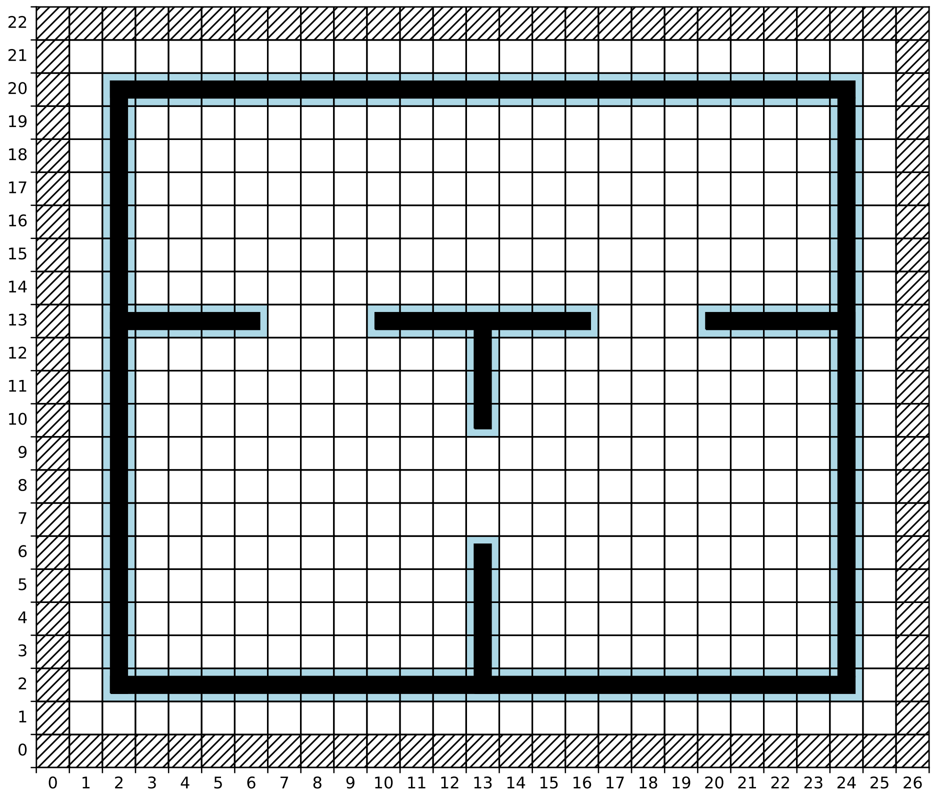


Figure 8.4: Voxel slice (light-blue) through three-room model (black)

8.2.5 Making Distance Field Array

The next step is to generate a distance field array derived from the building elements and the mask array. The mask array will be treated as a solid object that influences the values in the distance field. This distance field array will be calculated using the *scipy* Euclidean distance transform tool (SciPy Community 2019) which implements a function based on the *Voronoi* algorithm developed by Maurer, et al. (2003).

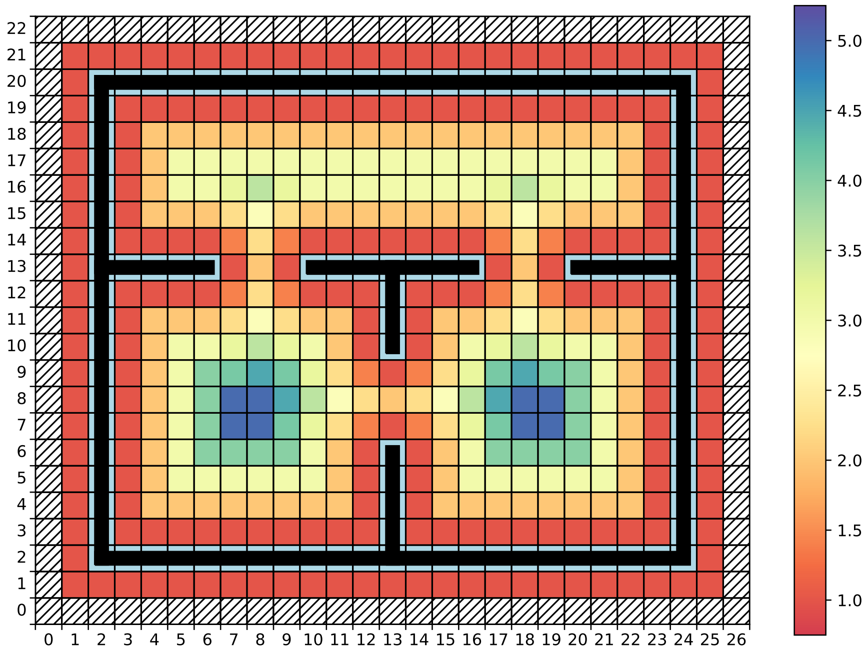


Figure 8.5: Voxel slice through three-room model - distance field

The output is an array of floating-point values measured in voxel units. The distances returned are measured between voxel centres. Because the *watershed transform* function requires an integer array for its input, this floating-point array will be multiplied by an arbitrarily large number (1000 will be used here) and then be converted to integer values.

Figure 8.5 shows a 2D plan of the features of the simple model with a slice of floating-point distance field array at the seventh voxel up from the base. In this diagram, the dark orange voxels alongside the features voxelised features have a value of 1.0, the orange voxels in the corners (i.e. at (1,1,6)) have a value of 1.414, and the indigo voxels in the centre of the spaces have a value of 5.091.

8.2.6 Identify Source Markers

The *watershed transform* starts its transformation by flooding up the distance field array. In order to maintain the flooding analogy, it is necessary to use a negated distance field, so that the voxels furthest from a building element are minima points. The algorithm uses these minima points as markers representing the sources of entry into the catchment areas of the array.

The algorithm can work without specifying these minima points, and the algorithm will work through the minima points as it finds them. The trouble with this is that a catchment area may have more than one cell holding the same minima value, which results in the catchment area being influenced by more than one source. An alternative is to use a gradient function on the distance field; however, the gradient function is susceptible to noise in the distance field arising from discretisation rounding. Beucher and Meyer (1993) recognised that over-segmentation is caused by the indiscriminate use of source markers to seed the algorithm. Beucher and Meyer (1993) recommended the use of homotopic modification to pre-select the source markers with which to seed the algorithm.

It is proposed to use the method described in the rest of this section for pre-selecting source markers in this research. This method will attempt to avoid over-segmentation by applying a discretisation factor to the floating-point values in the distance field and rounding up to the nearest integer. This discretisation will merge portions of the catchment area, forcing there to be only one source marker within one portion.

The adjacency of voxels in an array is described in terms of how many voxels are touched by the central voxel in a $3 \times 3 \times 3$ array. Voxels that touch face-to-face as illustrated in Figure 8.6a are referred to *6-adjacent*; voxels that touch edge-to-edge (Figure 8.6b) are referred to as *12-adjacent*; voxels that touch corner-to-corner (Figure 8.6c) are referred to as *8-adjacent*. Adjacency in all three dimensions is referred to as *26-adjacent*.

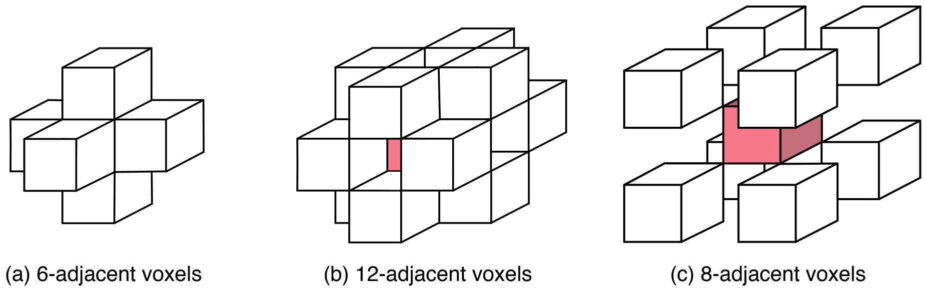


Figure 8.6: Voxel adjacency

The method will also use *26-adjacency* to merge regions of the catchment area, i.e. voxels that are *26-adjacent* are included when grouping regions together. When the lowest value in the discretised distance field is reached, the method switches to using *6-adjacency*; this ensures that catchment areas are not connected up through thin building elements that may only be *12-adjacent* or *8-adjacent*.

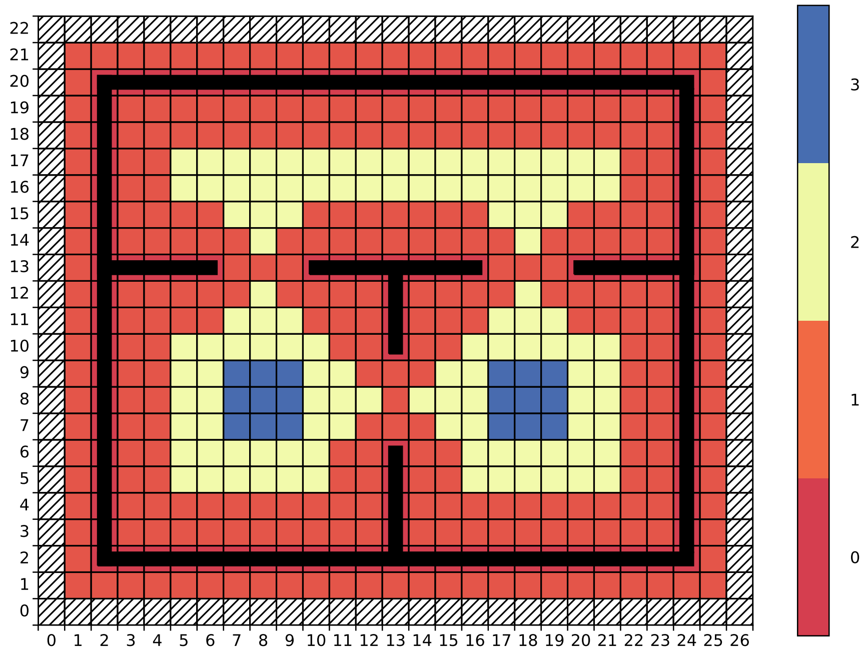


Figure 8.7: Voxel slice through discretised distance field (Discretion Factor(DF)=2.0)

Figure 8.7 shows a 2D plan of the features of the simple model with a slice of distance field array at the 7th voxel up from the base following discretisation using a factor of 2.0 . The values in the discretised distance field array are half of the values in the previous distance field array because each value has been divided by a Discretisation Factor (DF) of 2.0 before being rounded up to the nearest ceiling value (noting that all distances have not yet been negated). The effect of the DF is to create isodistant contour lines around the source locations and by choosing a DF of 2.0 is used, the isodistant contour lines are separated by at least one voxel. The effect of varying the DF will be investigated in Section 8.3.3.7.

Preparatory Step – A graph network is used to model adjacency with every voxel in the array with a discretised distance value higher than zero being represented as a node in the graph. Edges will be created between nodes representing adjacent voxels as the connecting up algorithm progresses.

Edges are added to the graph by traversing every voxel in the voxel array. At each voxel, the algorithm looks up the value of the voxels adjacent to that voxel. For reasons of computational efficiency, only half of the adjacent relationships need to be examined as the other half will be dealt with when the array traversal reaches the other half of the adjacent voxels.

If a *26-adjacent* voxel has the same value as the voxel being examined, then an edge representing the adjacency is added to the graph. The edge is attributed with two attributes, *start* and *end*, which are assigned the value of the two adjacent voxels the attributes.

If a *26-adjacent* voxel has a value greater or less than the value of the voxel, then an edge is added representing the adjacency and is the lower value is assigned to the *start* attribute and the higher value to *end*. I.e., if the first voxel has a value of 2 and the adjacent voxel has a value of 3, add an edge with *start = 2* and *end = 3*.

First iteration - Step 1 – The graph is now analysed to identify isolated sources and their connected relationships. A sub-graph containing those edges where the *start* attribute is equal to the highest value in the discretised distance array (i.e. 3) is extracted from the original graph structure. This sub-graph is then split into separate sub-graphs that are mutually connected. Figure 8.8 shows two separate sub-graphs containing edges that represent adjacent voxels that have a discrete value of 3.

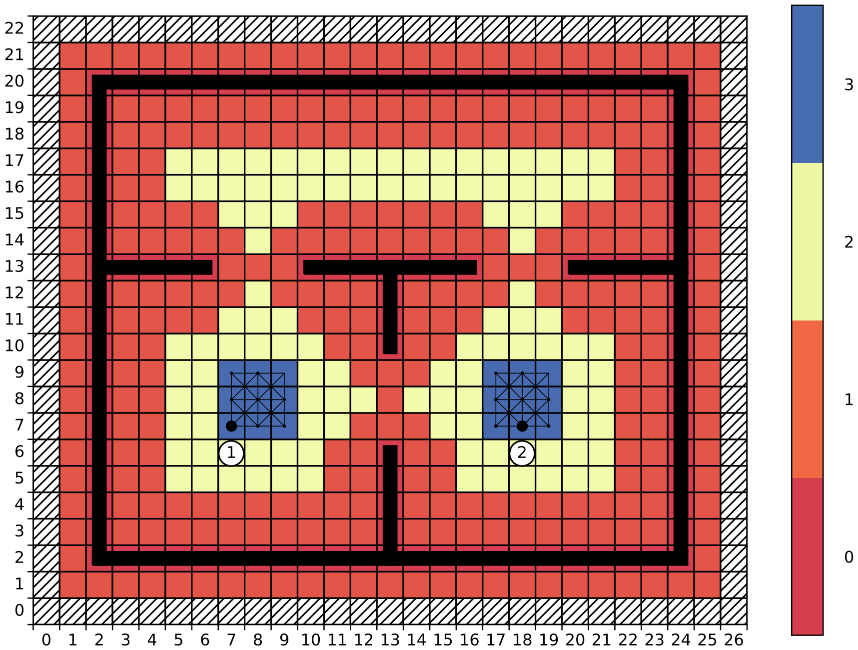


Figure 8.8: Building graph and identification of source markers (Value = 3)

First iteration - Step 2 – The next step is to determine the source of the two sub-graphs. During the first iteration, no sources will have as-yet been determined, and a node from each of the sub-graphs is chosen to be the source location. One of the nodes that corresponds to the voxel with the highest value in the floating-point distance field array will be tagged as a source node. These sources nodes are added to a dictionary of sources for future reference. In Figure 8.8, two voxels have been designated as source nodes, and these are identified by a number contained within a small black circle.

Second iteration - Step 1 – The next step is to repeat Step 3 for the second-highest value in the discretised distance array (i.e. 2). This time, the extracted sub-graph shall contain those edges where the *start* attribute is greater than or equal to the second-highest value in the discretised distance array (i.e. 2 and 3). Again the sub-graph is separated into mutually connected sub-graphs.

Figure 8.9 shows three unconnected graphs consisting of the edges connecting voxels with a discrete distance value of 2 or 3.

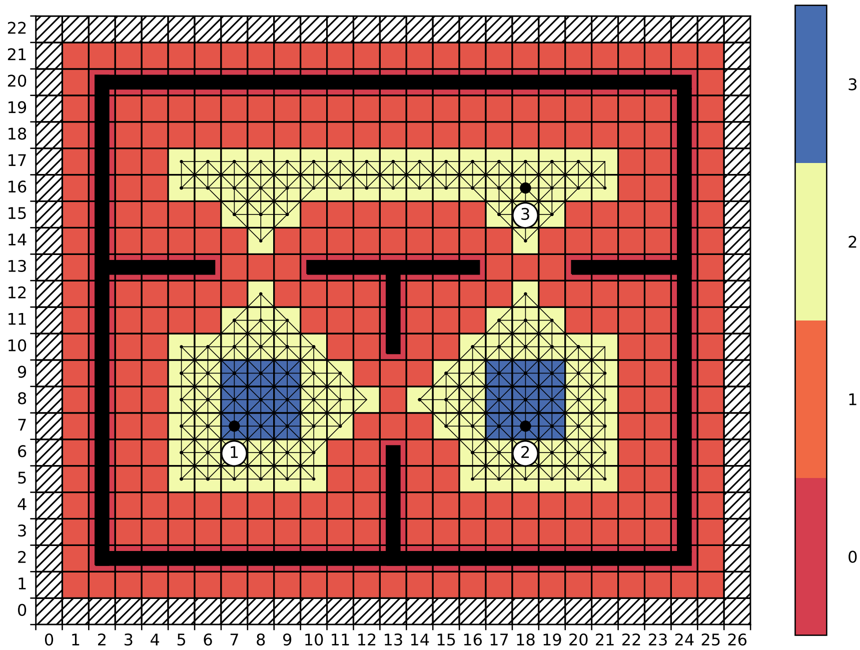


Figure 8.9: Building graph and identification of source markers (Value = 2)

Second iteration - Step 2 – Step 2 is repeated to determine the sources of each of the mutually connected sub-graphs. The bottom-left sub-graph in Figure 8.8 already contains Source 1 that was identified in the previous iteration. Likewise, the bottom right subgraph already contains Source 2 that was also identified in the previous iteration. However, the top sub-graph in Figure 8.8 does not contain a source, and so a node is chosen to be the source location. This node is identified as Source 3 in Figure 8.10.

Third iteration - Step 1 – Step 1 is repeated for the next highest value in the discretised distance array (i.e. 1) extracting a sub-graph containing those edges where the *start* attribute is greater than or equal to the value under consideration (i.e. 1, 2 and 3). Figure 8.10 shows two sub-graphs: a single interior sub-graph connecting all the voxels inside the features; and an exterior sub-graph connecting the voxels outside the features. Note that the graph edges between nodes of value 1 are connected if voxels are 6-adjacent.

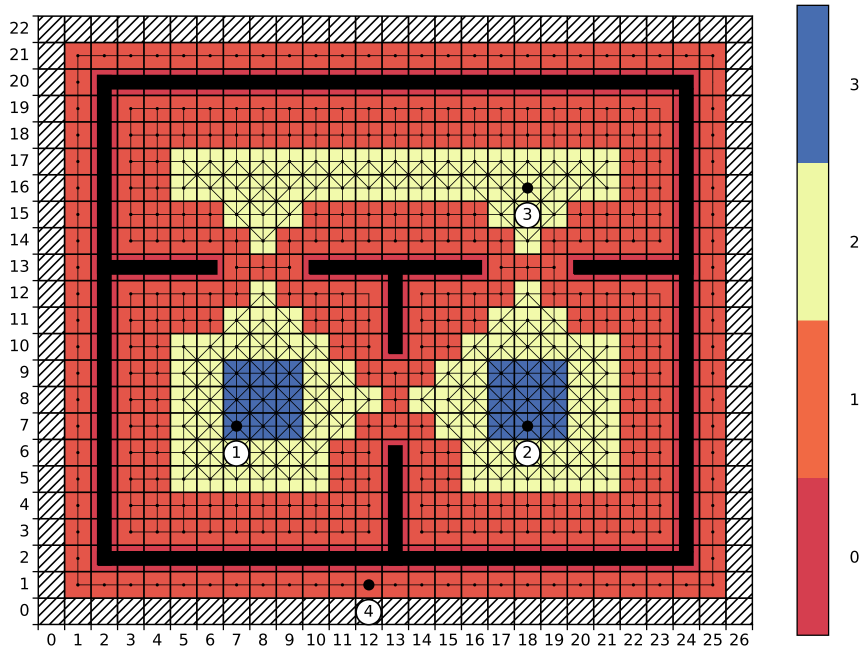


Figure 8.10: Building graph and identification of source markers (Value = 1)

Third iteration - Step 2 – This iteration has identified a fourth source in the new exterior sub-graph. The interior sub-graph already contains identified sources, therefore no further sources need to be added.

8.2.7 Perform Watershed Transform

Having iterated through the values in the discrete distance field array, the sources that have been identified are used as markers to seed the *watershed transform* algorithm. The *scikit-image watershed* function takes three arguments; the array that the algorithm will be performed on, an array containing the source markers used to seed the algorithm, and a mask array made from the element array and the mask array.

The result will be a voxel array representing the space around the features segmented into contiguous regions. These contiguous regions exist inside and outside the features, with the chance that some of the regions will be partially inside and partially outside. Figure 8.11 shows how four regions have been created by seeding the *watershed transform* with the four markers previously identified. The three internal spaces are displayed in perspective in Figure 8.12.

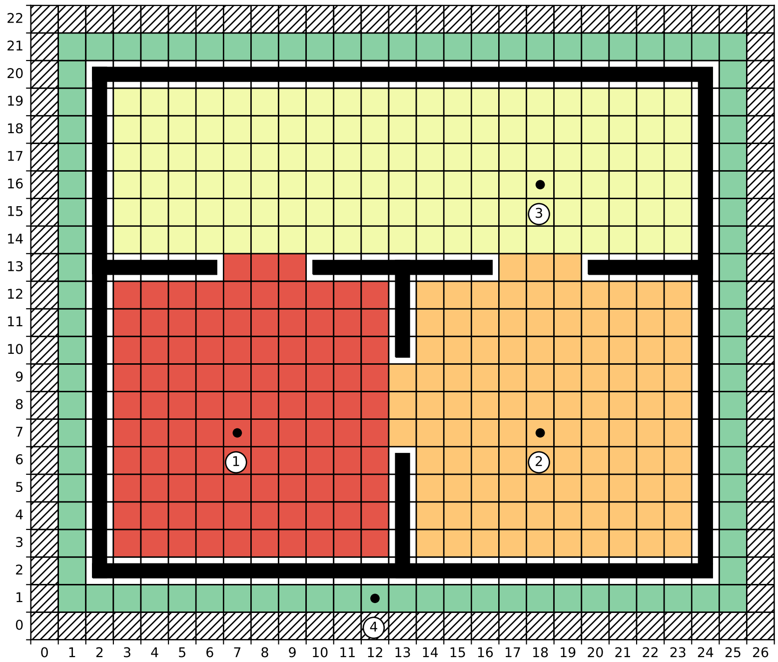


Figure 8.11: Spaces segmented by watershed transform - plan

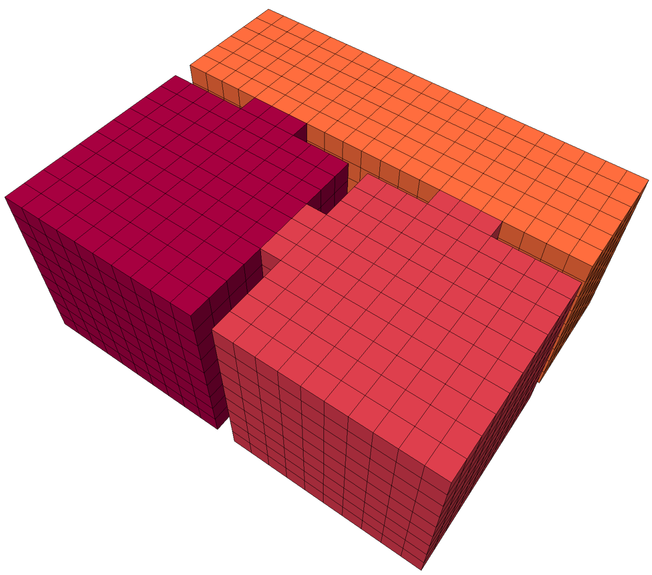


Figure 8.12: Spaces segmented by watershed transform - perspective view

8.2.8 Identify Internal Spaces

The spaces in the voxel array having been segmented, the next step will be to post-process the result and convert the spaces into a form that can be used for performing spatial joins. It is proposed to use the workflow in Figure 8.13 to achieve this, as described in the following subsections.

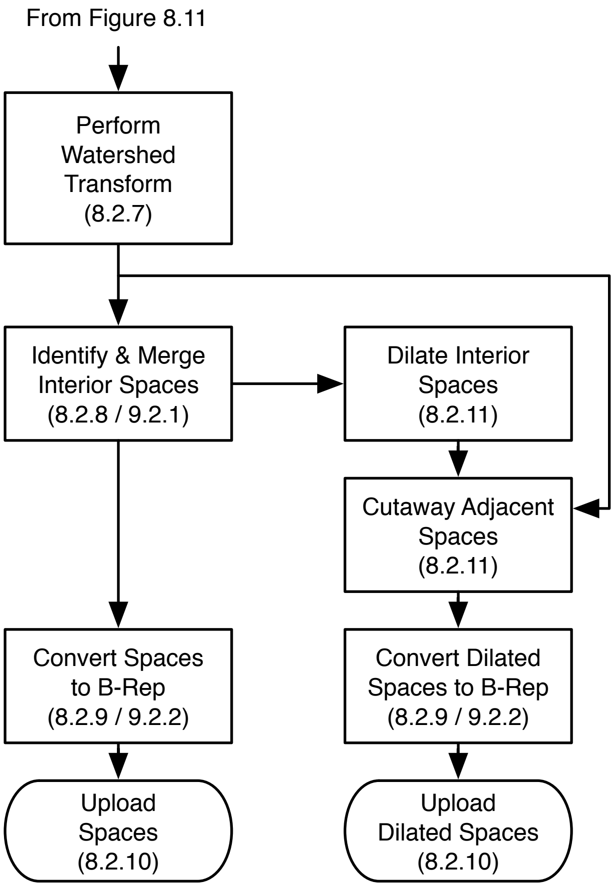


Figure 8.13: Watershed segmentation workflow - Part 2

A simple heuristic can be used to eliminate those regions that have a proportion of their surface area touching the convex hull. However, the quality of this heuristic has not been investigated. In this research, it is only used to pre-classify the results and allow manual identification.

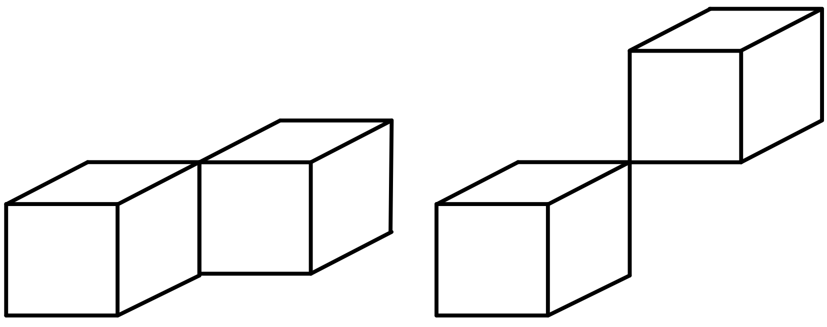
A 3D visualisation of the spaces is presented to the user in a window with a legend of all the spaces displayed alongside the visualisation. The heuristic algorithm has pre-determined which spaces are visible and invisible. Spaces are manually added and removed to correct the heuristic suggestions, as will be illustrated in Figure 8.16.

There is a possibility that the watershed transform algorithm will over-segment the spaces. In the event that this happens, the opportunity will also be given for spaces to be merged.

8.2.9 Convert Spaces to B-Rep

Each interior space is represented by a set of contiguous voxels that hold the value of that particular space. These voxels can be converted to a Boundary Representation (B-Rep) surface by traversing through the array looking for voxels with that value. If the adjacent voxel does not hold the same value, then a square face (or two triangular faces) is created between the two voxels orientated so that the surface normal is directed outwards.

This simple method is not guaranteed to create a manifold surface suitable for representing a watertight solid; however, the method is useful for obtaining a preliminary visualisation of the array. If just two voxels in array are configured such that only their corners or edges touch as illustrated in Figure 8.14 then the method described here will not create a manifold surface. A better algorithm suitable for creating a manifold B-Rep surface will be described in Section 9.2.2.



(a) (b)

Figure 8.14: Non-manifold voxel meshes

8.2.10 Upload to Oracle Spatial

Having converted the interior spaces from a voxel representation to B-Rep, the resulting watertight mesh can be written out in the *SDO_GEOMETRY* format and uploaded to *Oracle Spatial*.

8.2.11 Upload Dilated Interior Spaces

The voxel array spatial enumeration snugly fits inside the space it represents. This is a consequence of using the subdivision method as illustrated in Figure 8.3, although there may be occasional exceptions when the situation in Figure 8.3f occurs. If the voxel array in dilated by one voxel, it is expected that the actual enclosed space will be mostly contained by spatially enumerated array. It is therefore intended to dilate the merged interior spaces by one voxel[1] and upload an additional watertight mesh created from this dilated array.

8.3 Results and Observations

The three approaches described in Section 8.1 were used to create the spaces that make up the *Broadgate Ticket Hall* at Liverpool Street Station and a collection of spaces within the *Mile End Shaft*. The spaces are enclosed by building elements extracted from the two following construction files listed in Table 7.1 and illustrated in Appendix A.

The space complex that makes up the *Broadgate Ticket Hall* was chosen as it represents a challenging set of geometries with which to test the *watershed transform* algorithm; these geometries are challenging because they contain irregular shaped spaces that span multiple storeys. The challenge was made more difficult due to the architectural model files not being available for the ticket hall complex. The *Mile End Shaft* spaces were chosen as a second test case as they provide a contrast to the *Broadgate Ticket Hall* . Instead of the large irregular spaces found in the *Broadgate Ticket Hall*, these spaces are smaller and are partitioned with architectural walls and doors. Creating suitable spaces is

[1] It should be noted that this dilation is separate from, and in addition to, the dilation used in the method described in Section 9.2.2.

still a challenge because there are many openings passing between the spaces. The model files used to perform the *watershed segmentation* for the *Broadgate Ticket Hall* are listed in Table 8.3.

Table 8.2: Model files used to enclose BTH spaces

Model File	Type	Description	Comment
MES-S-00009	Structural	Mile End Shaft mid level	Similar to MES-S-00007
MES-S-000010	Structural	Mile End Shaft mid level	Similar to MES-S-00007
MES-S-000011	Structural	Mile End Shaft mid level	Similar to MES-S-00007
MES-S-000011	Structural	Mile End Shaft mid level	Duplication of MES-S-000011 then offset by storey height
MES-A-2-00001	Architectural	Mile End Shaft walls and doors	Duplicated and offset for each storey
MES-A-Z-31749	Architectural	Mile End Shaft staircase	-
Bespoke Model	Structural	Extruded cylindrical shell	Bespoke model created in lieu of the outer casing model

The *floor plan extrusion* and *watershed segmentation* methods were also used to create spaces that make up the *Mile End Shaft* . These spaces are enclosed by building elements extracted from three similar structural model files (similar to *MES-S-00007* used in Chapter 7), one architectural model file (similar to MES-A-2-00001) and the staircase architectural model file (*MES-A-2-Z-31749*). The model was capped with offset duplicates of the slab elements from *MES-S-00011* and enclosed by an extruded cylindrical shell; the structural model file for the shaft casing was not available so a cylindrical shell was substituted in its place. The model files used to perform the *watershed segmentation* for the *Mile End Shaft* are listed in Table 8.3.

Table 8.3: Model files used to enclose MES spaces

Model File	Type	Description	Comment
MES-S-00009	Structural	Mile End Shaft mid level	Similar to MES-S-00007
MES-S-000010	Structural	Mile End Shaft mid level	Similar to MES-S-00007
MES-S-000011	Structural	Mile End Shaft mid level	Similar to MES-S-00007
MES-S-000011	Structural	Mile End Shaft mid level	Duplication of MES-S-000011 then offset by storey height
MES-A-2-00001	Architectural	Mile End Shaft walls and doors	Duplicated and offset for each storey
MES-A-Z-31749	Architectural	Mile End Shaft staircase	-
Bespoke Model	Structural	Extruded cylindrical shell	Bespoke model created in lieu of the outer casing model

8.3.1 Floor Plan Extrusion

The Crossrail floor plans for the *Broadgate Ticket Hall* were passed through *FME Workbench* using the method proposed in Section 8.1.1. The spaces created were converted using the *GeometryCoercer* transformer into a B-Rep solid mesh and uploaded back to *Oracle Spatial* to be used for spatial query operations. An illustration of these spaces created is shown in Appendix C.1.

8.3.2 Boolean Difference

A rectangular 2D area was drawn up in CAD around the building elements of *Broadgate Ticket Hall* that were taken from the two CAD model files. This rectangular profile was extruded in *FME Workbench* to form a solid template profile that spanned the volume of the spaces to be created.

An *FME* workspace was written to read in the solid building element features created in Chapter 7 and to use with the *FME Clipper* transformer and the extruded solid profile. The imported features were tested using the *FME GeometryValidator* transformer, and it was found that a proportion of the features required fixing to repair their geometry before subsequent use. It would appear that some features, which had been validated as watertight solids in Chapter 7, have become corrupted at some stage during the export to *Oracle Spatial* and being read back into *FME Workbench*.

The spaces created during early attempts to use the workflow suffered from a multitude of slivers that remained in between the building elements that had been extracted. These slivers could be avoided by buffering the building elements by 1 mm before they were used for clipping; however, this operation comes at the expense of forming a space that is fractionally smaller than it should be. The effect can be offset by buffering the result by the same amount.

The *Clipper* transformer was then used in the *FME* workspace to produce a *boolean difference* of the solid form and the enclosing elements. The resulting output was then checked using the *GeometryValidator* transformer, and the output was found to fail validation and could not be repaired.

The following strategies were attempted to achieve a valid watertight result:

a. A large offset was applied to the input features to reduce the magnitude of the coordinate digits in the hope of reducing any floating-point errors. Applying this offset had a marginal improvement on the performance of the difference operation, but not significant enough to provide a reliable solution.

b. The faces of the resulting output were de-aggregated, repaired with the *GeometryValidator* and stitched back together, but this did not produce a valid result.

c. The geometry was exported into *Meshlab*, a third-party application, that provides tools for inspecting, cleaning and repairing mesh geometry, including merging close vertices and removing zero area faces (MeshLab 2020). Using *Meshlab* sometimes resulted in repaired geometry, but the process was haphazard and could not be relied upon.

Although using the *boolean difference* method with the *FME Clipper* transformer failed to create valid watertight solids, the process was successful at creating surface meshes that can be used to visualise spaces.

Figure 8.15 contains a visualisation of the result of using *boolean difference* on a single cuboid template profile. It can be seen that the result provides limited benefit either towards performing spatial queries or for visualisation.

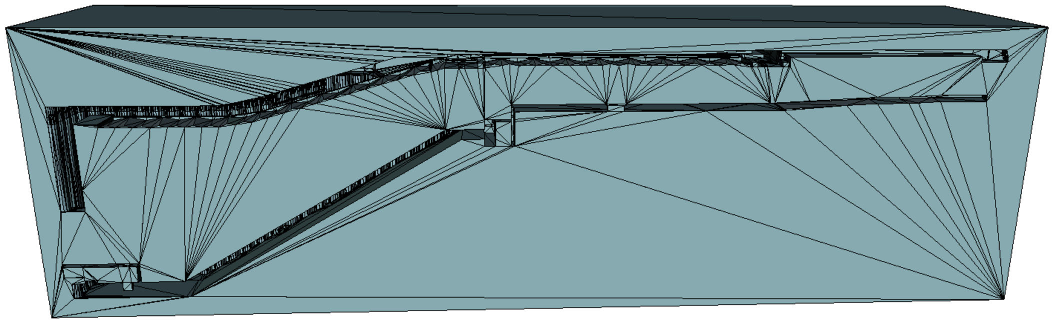


Figure 8.15: Cutaway of Broadgate Ticket Hall

8.3.3 Watershed Segmentation

The program and workflow described in Section 8.2 was written and developed in *Python* using 12,000 lines of *Python* code. The workflow of this program is illustrated in Figure 8.1 and Figure 8.13, and the full code is appended to this thesis (Appendix G.3).

8.3.3.1 Variation of Watershed Segmentation Parameters

The program has been written to read in certain parameters from a settings file. As well as controlling the flow of the program, the settings file contains the values of the parameters that can be altered. These parameters have been varied to investigate their effect on the performance of the algorithm. The effect of the following parameters will be discussed in the following sections.

a. Voxel grid orientation

b. Voxel size

c. Distance field discretisation factor

d. Size of convex hull dilation

8.3.3.2 Voxel grid orientation

The orientation of the voxel grid for the *Broadgate Ticket Hall* was manually set as (0.97, -0.24, 1) which rotated the North axis 13.9 degrees eastwards about the vertical axis. The effect of re-orientating the voxel grid is to bring the voxel faces in alignment with the majority of the building element features.

The orientation of the voxel grid for the *Mile End Shaft* was aligned with North as an exercise to test how well the algorithm works without aligning the axes.

8.3.3.3 Voxel grid cell size

The size of the voxel cell is inversely proportional to the resolution of the voxel grid. The *watershed segmentation* algorithm will be tested on voxel sizes of 1 m, 0.5 m and 0.25 m for the *Broadgate Ticket Hall* and 0.5 m, 0.25 m and 0.1 m for the *Mile End Shaft*.

8.3.3.4 Distance field discretisation factor

The size of the distance field discretisation factor influences the width of the regions in between contours in that field. Using a large factor will result in large bands, which will reduce the number of source markers leading to less segmentation. The *watershed segmentation* algorithm will be tested on discretisation factors of 2.0, 1.0, 0.5 and 0.25. The choice of these DF values, along with the voxel size will be discussed in Section 8.3.3.7.

8.3.3.5 Convex hull dilation

The size of the convex hull dilation affects the ability of the *watershed transform* to identify boundaries at the interface between interior and exterior spaces. With no dilation, analysis of the discretised distance field does not yield a source marker in the region of the exterior space. Without an exterior source marker, there will be no zone influence, and a Skeleton of Influence Zone (SKIZ) will not be created at the interface.

A convex hull dilation of two voxels has been used throughout these tests.

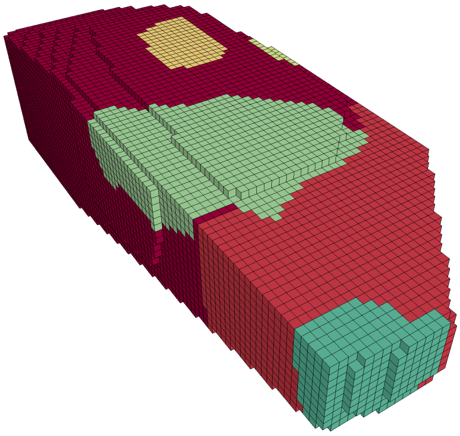
It should be noted that spaces created in the hull dilation are cropped from the final output array. When openings to the exterior space are coincident with the convex hull, the impression may be given that the *watershed transform* algorithm is performing the cut, whereas the operation is actually being performed by the hull dilated crop.

8.3.3.6 Manual Post Processing of Watershed Segmentation

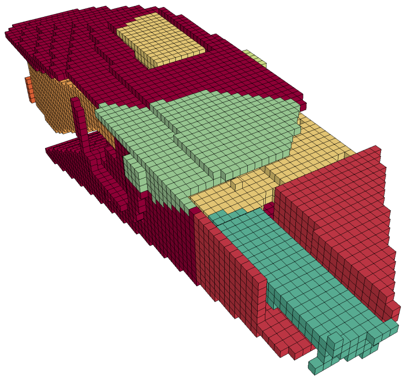
Identification of Interior Spaces – Running the *watershed segmentation* program on the *Broadgate Ticket Hall* model creates a collection of spaces as illustrated in Figure 8.16. Because the spaces in the model are not closed, visualisation of the interior spaces is occluded by the exterior spaces which form around the outside of the building but still within the limits of the convex hull.

A simple heuristic algorithm was used to pre-cull these exterior spaces, as shown in Figure 8.16; however, the workflow developed in this chapter relies on the interior spaces to be correctly identified by manual selection. The program was written with an interactive visualisation enabling these exterior spaces to be removed on screen.

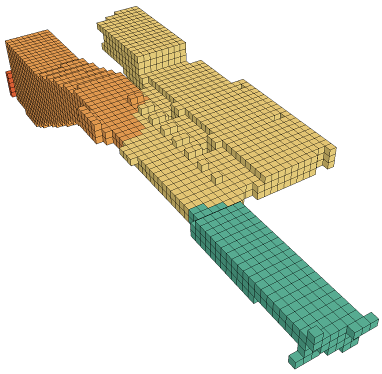
Taking the segmentation of 1 m voxels using a DF of 2.0 as an example, Figure 8.16a is a visualisation of the spaces as produced, Figure 8.16b shows the spaces with the hull dilation removed after being pre-culled and Figure 8.16c shows the interior spaces after manual removal.



(a) Exterior spaces



(b) Automatic removal of exterior spaces



(c) Manual removal of exterior spaces

Figure 8.16: Removal of exterior spaces to reveal interior spaces

Merging of Interior Spaces – From the results in Figure 8.18, it can be seen that the resulting output is over-segmented, especially when performed at the smaller voxel sizes and levels of discretisation that are required to create space boundaries that correspond with the real interior spaces. Further study is required to investigate why over-segmentation is occurring and whether there are any practical solutions, before or after performing the *watershed transform*.

For the immediate work of this thesis, there is a need to create spaces that are suitable for testing spatial join operations in Chapter 9, and so the over-segmented spaces will be manually selected and merged to create spaces that are approximately aligned with the floor plan (see Chapter 9).

8.3.3.7 Variation of Voxel Size and Discretisation Factor

The *watershed segmentation* method was performed on the *Broadgate Ticket Hall* using nine combinations of voxel cell size (1.0 m, 0.5 m, and 0.25 m) and discretisation factor (2.0, 1.0, and 0.5). The segmentations produced for each combination are presented in Figure 8.18 with larger illustrations, as well as plan and section views provided at Appendix B. Slightly different combinations of voxel size and DF were be used for investigating the *watershed segmentation* on the *Mile End Shaft* later in this subsection.

The larger voxel size was chosen arbitrarily at 1.0 m and then decreased in size. At some point, the minimum voxel size is prohibited by memory available to perform the algorithm. Although time is not necessarily a problem, decreasing the voxel size will increase computational time governed by a third order power law. The smaller of the three voxel sizes, 0.25 m, is expected to be a satisfactory compromise between representation and computational resources, as many assets are expected to be larger than this size.

The DFs were chosen to represent idempotent zones that would be approximately two voxels, one voxel wide and less than one voxel wide. Choosing a DF of 2.0 results in the creation of idempotent zones that are at least one voxel wide. However, when a large DF is used in conjunction with larger voxel sizes, the algorithm will fail to identify small spaces, as will be demonstrated by the results from the *Mile End Shaft* in Section 8.3.3.7. The program was also run with a DF of 0.25; at a voxel size of 1.0 m, the results were identical to the output produced using a DF of 0.5; however, at a voxel size of 0.25 m, the lower DF resulted in further over-segmentation.

From a manual inspection of the structural model files, five spaces can be identified; these are the north half of the ticket hall, the south half of the ticket hall, the escalator descent towards the platforms, the escalator ascent to the street, and a long corridor leading off to the east as illustrated in Figure 8.17. The use of the *watershed segmentation* algorithm is driven by the need to seal up openings, of which four basic types can be found among the five spaces of the Broadgate complex, as depicted in Figure 8.17. The results of varying voxel size and DF shall be assessed by considering how well each type of opening is closed.

Wall and floor openings – Three openings within the building element geometry connect the ticket hall with exterior spaces. The algorithm has segmented the ticket hall space from the exterior space within these openings in each of the tests.

The ticket hall subdivision – The ticket hall is split into two halves by the presence of two pillars. The tests performed using a voxel size of 1 m have failed to identify this boundary, although the test with a DF of 0.5 had partial success. The tests performed with a voxel size of 0.25 m detected a boundary in all three cases; however, the boundary is not a planar surface, and the result is obfuscated due to over-segmentation.

The internal portals – There are three interior portals that lead off the ticket hall to the escalators and the corridor. Only the test at 0.25 m with a DF of 0.5 was able to pick up these portals. Again the boundaries do not approximate a planar surface and over-segmentation prevents automatic detection of these boundaries.

The exterior portals – The ticket hall complex has three exterior portals. The first is a horizontal portal where the escalator shaft to the street reaches ground level; the next is at the bottom of the escalator shaft to the platforms, and the last is at the end of the corridor.

The success of whether the algorithm can form a boundary at these portals is undetermined as each of these portals is coincident with the surface of the convex hull. Removing the dilated buffer from the result of the *watershed transform* masks the result at lower resolutions, however, in the 0.25 m tests it is evident that the *watershed transform* has not positioned a space boundary in the appropriate location.

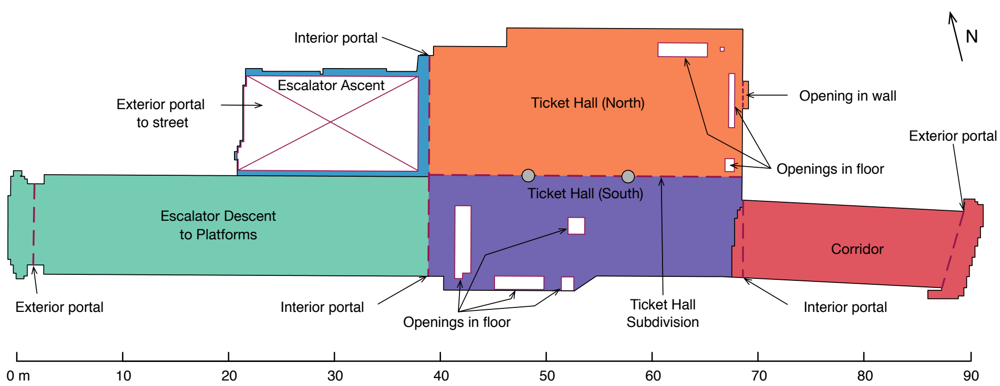
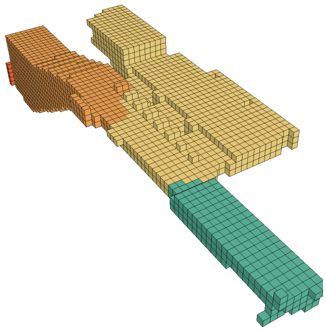


Figure 8.17: Spaces and openings within Broadgate Ticket Hall

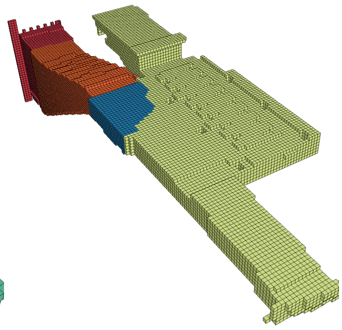
Mile End Shaft – The *watershed segmentation* algorithm was performed on the *Mile End Shaft* spaces with a variety of voxel sizes and discretisation factors. For this test case, the range of voxel sizes was adjusted down to 0.5 m, 0.25 m and 0.1 m to take into account the smaller spaces. The range of discretisation factors was also changed to 1.0, 0.5 and 0.25.

A qualitative inspection of the results shows that using 0.5 m sized voxels is too coarse. Moreover, when this voxel size is used with a discretisation of 1.0, the algorithm fails to detect the presence of the room. It would appear that using a voxel size of 0.25 m with the three different discretisation factors results in well formed spaces without over-segmentation. Furthermore, the *watershed segmentation* algorithm has been successful at closing off the various openings between spaces throughout the model, using parameters set at 0.25 m and a DF of 0.5.

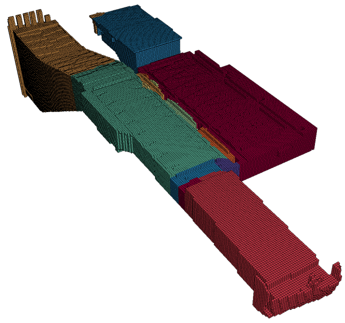
Because the *Mile End Shaft* model is smaller is size, it has been possible to investigate using the algorithm with 0.1 m voxels. This results in a higher quality representation, although there is some over-segmentation when a DF of 0.25 is used.



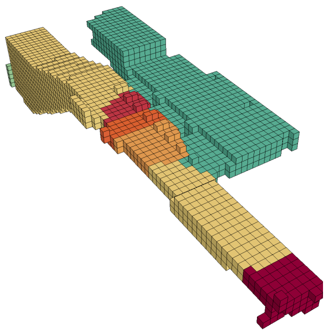
(F1) Size=1.0 m, DF=2.0



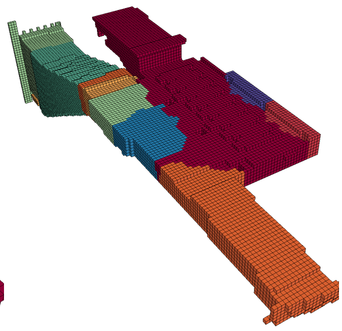
(F2) Size=0.5 m, DF=2.0



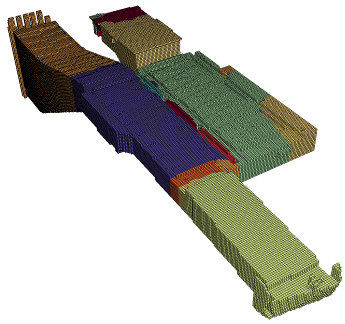
(F3) Size=0.25 m, DF=2.0



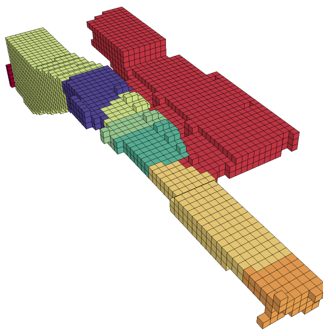
(F4) Size=1.0 m, DF=1.0



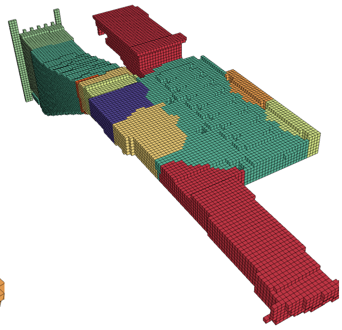
(F5) Size=0.5 m, DF=1.0



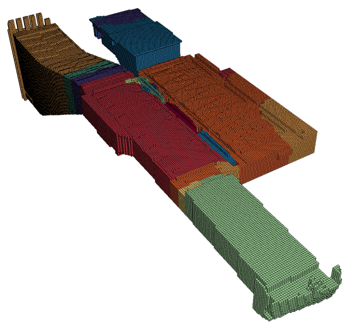
(F6) Size=0.25 m, DF=1.0



(F7) Size=1.0 m, DF=0.5

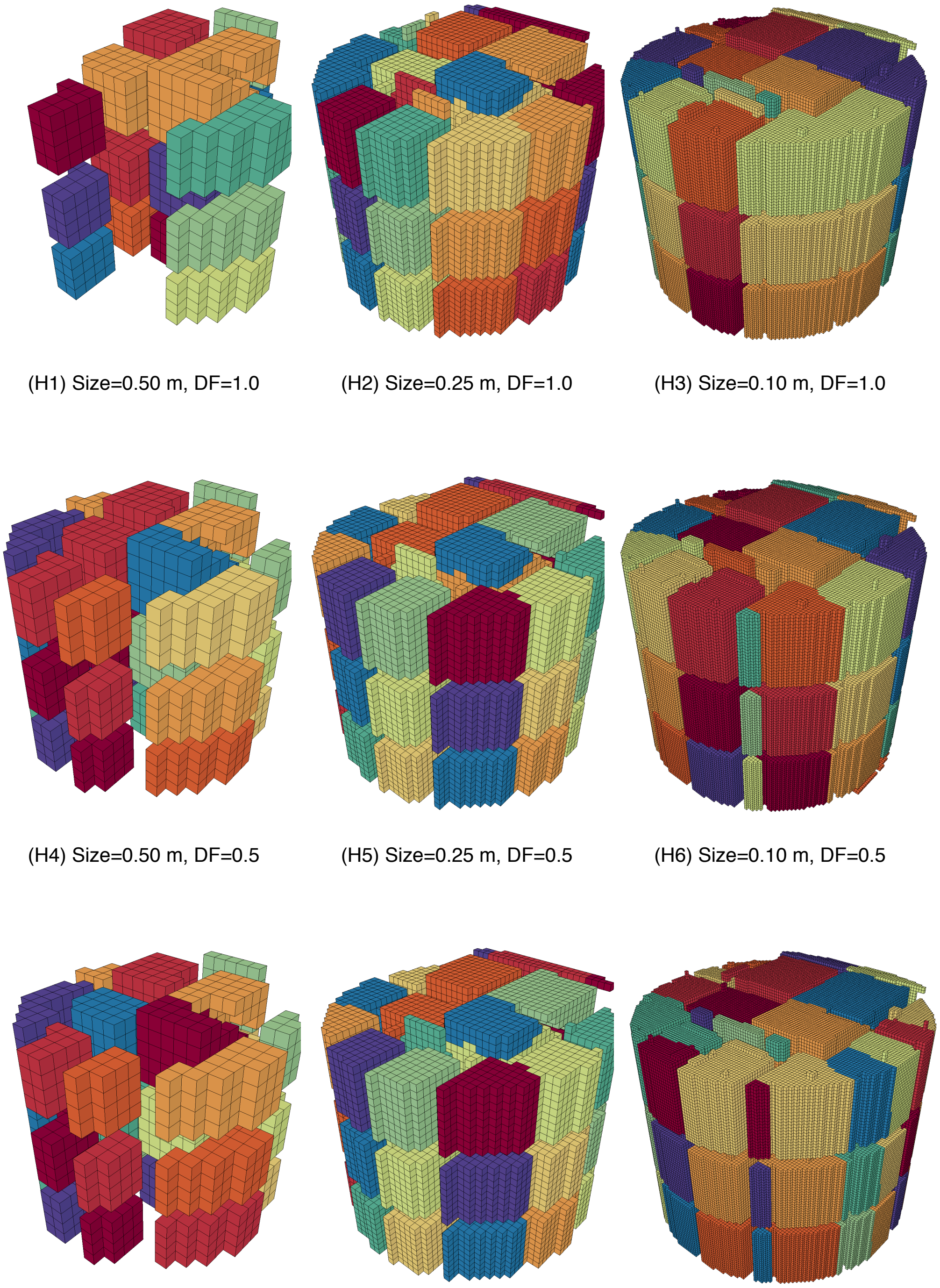


(F8) Size=0.5 m, DF=0.5



(F9) Size=0.25 m, DF=0.5

Figure 8.18: Performance of Watershed Segmentation varying voxel size and discretisation factor (Broadgate Ticket Hall)



(H1) Size=0.50 m, DF=1.0 (H2) Size=0.25 m, DF=1.0 (H3) Size=0.10 m, DF=1.0

(H4) Size=0.50 m, DF=0.5 (H5) Size=0.25 m, DF=0.5 (H6) Size=0.10 m, DF=0.5

(H7) Size=0.50 m, DF=0.25 (H8) Size=0.25 m, DF=0.25 (H9) Size=0.10 m, DF=0.25

Figure 8.19: Performance of Watershed Segmentation varying voxel size and discretisation factor (Mile End Shaft)

8.3.3.8 Computational time

The computational times taken to produce the results illustrated in Figure 8.18 are presented in Table 8.4. These timings were measured from a single execution performed on a *Macbook Pro* (see Appendix F for processor specifications).

Table 8.4: Watershed segmentation computation times

Run	Voxel Size	Discret. Factor	Mask (s)	Voxelise (s)	DFT (s)	Markers (s)	Watershed (s)	Total (min)
B1	1.0	2.0	3.9	20.5	0.2	5.5	0.1	0.5
B2	0.5	2.0	12.0	52.7	2.0	28.9	0.5	1.6
B3	0.25	2.0	52.1	219.3	19.9	873.8	5.2	19.5
B4	1.0	1.0	4.1	21.3	0.2	6.1	0.1	0.52
B5	0.5	1.0	11.8	56.7	2.0	53.2	0.5	2.1
B6	0.25	1.0	53.1	213.8	19.7	882.7	5.0	29.6
B7	1.0	0.5	4.0	19.9	0.2	4.3	0.1	0.5
B8	0.5	0.5	11.5	54.4	2.0	40.4	0.5	1.8
B9	0.25	0.5	50.6	213.0	19.6	1168.6	5.1	24.3

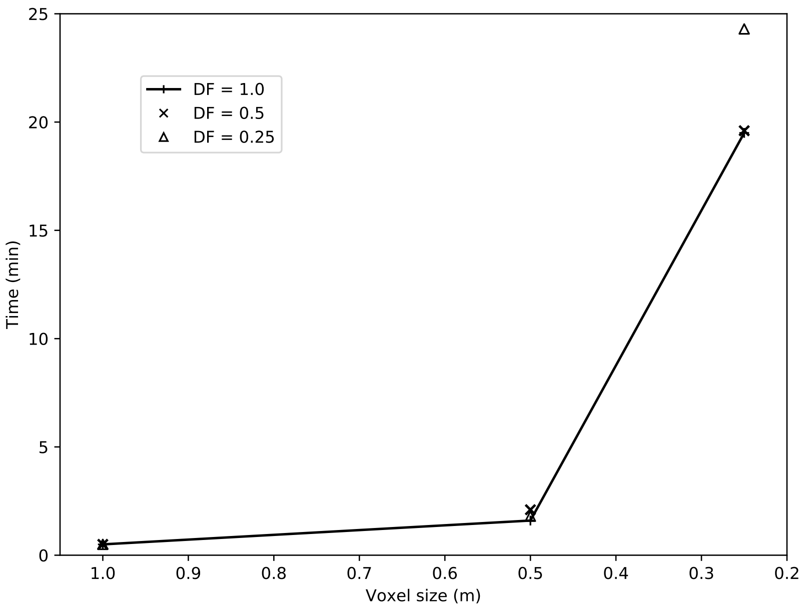


Figure 8.20: Watershed segmentation computation times

8.3.3.9 Combined Approach

If the voxel array output from using *watershed segmentation* is dilated by one voxel and the result passed to the *FME Workbench* for clipping using the same workspace as described in Section 8.3.2, then spaces, such as those illustrated in Figure 8.21, can be obtained. The geometry of spaces can also be manipulated to improve the visualisation experience. In Figure 8.22, the faces of the space boundary that face towards the viewer have been culled, creating an open aspect into the space.

Although it has not been possible in the course of this research to create watertight spaces due to problematic boolean operations, it should be possible to develop a suitable method on further investigation into suitable boolean operations (see FW4 in Chapter 11).

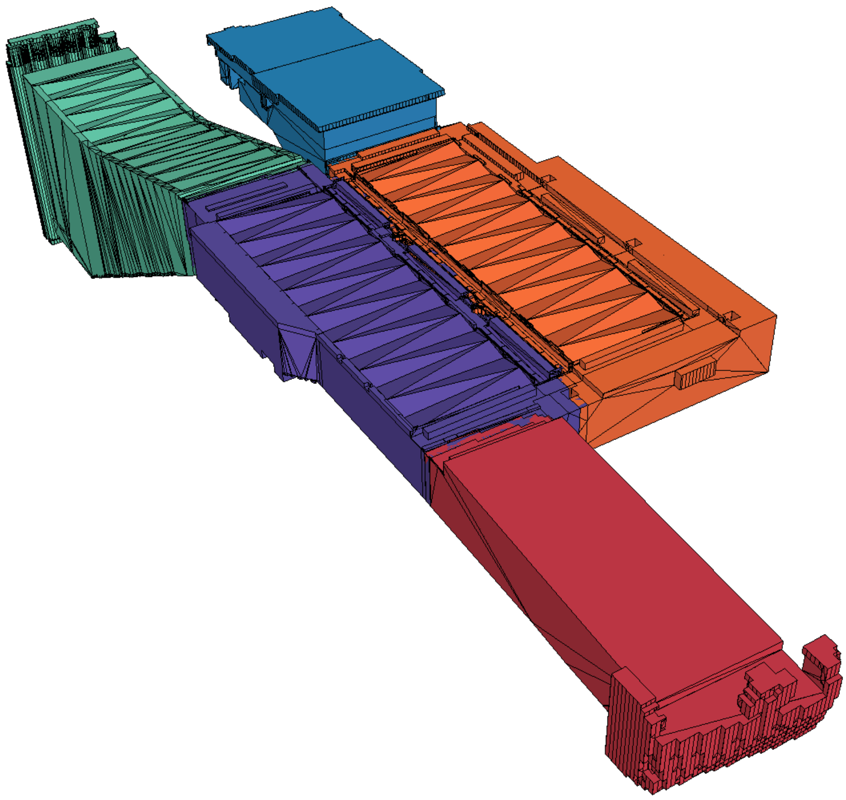


Figure 8.21: Spaces created using combined method

8.3.3.10 Summary

The tests performed in this subsection have demonstrated the basic capabilities and limitations of *watershed segmentation*. In summary, the algorithm is useful as it can be relied upon to create a watertight space, with robust geometry that is unlikely to deteriorate down the line. It is successful at closing off small to medium (relative to space width/height) openings in floors, walls and ceilings, using parameters set at 0.25 m and a DF of 0.5. The algorithm does have the ability to close off large portals, although the quality of the boundaries in these cases is not perfect.

There are several approaches that could be used to improve the quality of the output, and these will be discussed in Chapter 10. In the meantime, these spaces produced here will be tested in the next chapter on their ability to perform spatial joins.

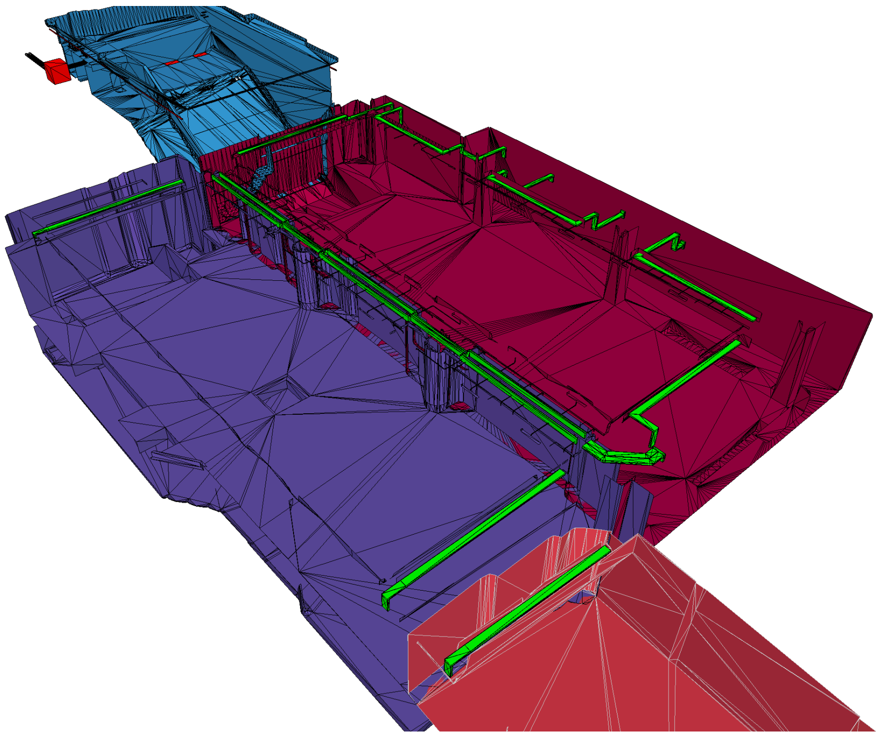


Figure 8.22: Boolean difference spaces with open aspect

9 Using Spaces for Asset Management

The previous chapters have described two of the three steps required in the workflow to link Computer Aided Design (CAD) elements with Asset Information Management System (AIMS) assets, as identified in Section 2.2. In this chapter, the CAD elements extracted in Chapter 6 and Chapter 7 and the spaces created in Chapter 8 are joined using a spatial query so that each CAD element will be linked to the space, or spaces, in which it is located (PRS 3); alternatively, the CAD element may not be located in any space at all.

9.1 Method

The first step in assessing the suitability of using spaces for Asset Management (AM) is to investigate the most appropriate platform on which to perform spatial joins. Up until now, *Oracle Spatial* has been used as storage medium for holding element features and space features. The *Oracle Spatial* platform was chosen as it is a robust medium via which features can be written to and from *FME Workbench* and *Python* and which has its own spatial join functionality through the *SDO_ANYINTERACT* function. The other candidate platform is *ArcGIS* accessed through the *ArcScene* application and using the *Inside3Dl* tool that is part of the *3D Analyst* toolbox.

Before performing the spatial operations on a Geographic Information Systems (GIS) platform, the *Broadgate Ticket Hall* spaces will be aggregated or merged and converted to a manifold form. Once this is done, spatial queries will be performed on the two sets of Mechanical, Electrical and Plumbing (MEP) features in Table 9.1 together with three sets of spaces taken from Table 9.2 using the most appropriate tool and platform. The

BTH-AGGREGATED set contains those spaces generated using *floor plan extrusion* (the *BTH-EXTRUDED* set) aggregated into five principal spaces; the *BTH-MANIFOLD* set is formed from merging the over-segmented spaces that were generated in Chapter 8 using a voxel size of 0.25 m with a Discretisation Factor (DF) of 0.5. The *BTH-DILATED* set is a single-voxel dilation of the spaces that were used to create the *BTH-MANIFOLD* set. A summary of these space sets can be found at Table 9.2. The output from these six spatial join operations will then be visualised to enable a qualitative assessment of the operation.

It is also proposed to investigate the performance of spatial join operations on the spaces produced from the *Mile End Shaft* model. For this investigation the MEP features will be extracted from two Mechanical MEP model files and two Electrical model files (in Table 9.1) and merged into a single featurset *MES-ME-4-00001*. The spatial operations will be performed between this featureset and three sets of spaces (Table 9.3).

The first set of spaces will be created using *floor plan extrusion*. Because *MES-EXTRUDED* consists of spaces on three identical levels (Levels 4-6), the spatial query will be simplified by reducing the set to include only the eleven spaces on Level 4 as listed in Table 9.5 and to be referred to as *MES-EXTRUDED-11*. The second set has been created by merging the output of the watershed segmentation using a voxel size of 0.25 m with a discretisation factor of 0.5 (the *MES-MANIFOLD* set), and a third set (the *MES-DILATED* set), similar to the *MES-MANIFOLD* set except that the spaces have been dilated all round by a single voxel. Again, these sets are simplified by extracting the eleven spaces on Level 4 as listed in Table 9.5 and will be referred to as *MES-MANIFOLD-11* and *MES-DILATED-11*.

Table 9.1: MEP feature sets (Broadgate Ticket Hall and Mile End Shaft)

MEP Feature Set	Description
LPL-E-1-42201	370 electrical asset features generally located on Level 1
LPL-E-2-42205	570 electrical asset features generally located on Level 2
MES-ME-4-00001	48 mechanical and electrical assets generally located on Level 4

Table 9.2: Broadgate Ticket Hall space sets

Appendix	Space Set	Description
C1	BTH-EXTRUDED	Extruded Floor Plan
C2	BTH-AGGREGATED	Extruded Floor Plan Minor spaces aggregated into 5 principal spaces
C3	BTH-MANIFOLD	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 5 principal spaces Converted to Manifold B-Rep
C4	BTH-DILATED	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 5 principal spaces Dilated by one voxel (0.25 m) Converted to Manifold B-Rep
C5	BTH-CLIPPED	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 5 principal spaces Dilated by one voxel (0.25 m) Converted to Manifold B-Rep Clipped with building element features Used for visualisation only

Table 9.3: Mile End Shaft space sets

Appendix	Space Set	Description
E1	MES-EXTRUDED	Extruded Floor Plan 38 spaces extruded
	MES-EXTRUDED-11	11 spaces on Level 4 selected for Spatial Join
E2	MES-MANIFOLD	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 25 principal spaces Converted to Manifold B-Rep
	MES-EXTRUDED-11	11 spaces on Level 4 selected for Spatial Join
E3	MES-DILATED	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 25 principal spaces Dilated by one voxel (0.25 m) Converted to Manifold B-Rep
	MES-DILATED-11	11 spaces on Level 4 selected for Spatial Join

9.2 Preparation of Spaces

9.2.1 Merging of Spaces

In Chapter 8, performing the *watershed transform* algorithm with a voxel size of 0.25 m and a DF of 0.5 produced the most suitable results for use in this chapter. These spaces are most suited because their boundaries align most closely with the boundaries found in the floor plan boundaries; however, using these spaces is problematic because they are over-segmented. It is intended, therefore, to merge these over-segmented spaces to create five principal spaces as listed in Table 3.2 and illustrated in Appendix C.3.

The extruded spaces generated from the *Broadgate Ticket Hall* floor plans (as illustrated in Appendix C.1) also contain five similar spaces, as well as collection of minor spaces which are bounded by non-structural walls. These minor spaces were not created by the *watershed segmentation* method as the model files containing the non-structural walls were not available. It is intended to aggregate these minor spaces into the five spaces that will be broadly aligned with the five principal spaces in Table 9.4. These spaces are illustrated in Appendix C.

These five spaces are simplified representations that have been created specifically to prove that the spatial join operations carried out in Section 9.3.2 are achievable in practice, and to identify any challenges arising. The names of these five spaces have been arbitrarily chosen for the purpose of assessing the suitability of spatial join operations.

Similar operations were performed on the *Mile End Shaft* sets of spaces, creating eleven principal spaces as list in Table 9.5. These spaces are illustrated in Appendix E.

Table 9.4: Broadgate Ticket Hall spaces

Space	Description	Colour
1	Escalator Descent to Platforms	Green
2	Escalator Ascent to Street	Blue
3	Ticket Hall (North)	Maroon
4	Ticket Hall (South)	Purple
5	Corridor	Red

Table 9.5: Mile End Shaft spaces

Space	Description	Level	Colour
1	Void Space	4-6	Maroon
2	Stairway	4-6	Camel Beige
3	Lift Shaft	4-6	Purple
4	Riser	4-6	Blue
5	Lift Lobby	4	Lemon Green
6	Corridor A	4	Maroon
7	Corridor B	4	Salmon
8	Room A	4	Salmon
9	Room B	4	Squash
10	Room C	4	Light Green
11	Room D	4	Camel Beige

9.2.2 Conversion to Manifold B-Rep

The spaces created using *watershed segmentation* in Chapter 8 are represented as a spatial enumeration in the form of a voxel array. Section 8.2.9 in the previous chapter described a simple method for converting a voxel array to Boundary Representation (B-Rep) that is suitable for visualising the space. However, this method cannot be relied on to describe a watertight manifold solid due to the reasons previously explained.

A novel approach has been developed in support of this research to create a manifold B-Rep surface from a voxel array. It is based on the observation that if each voxel in the array is subdivided into 64 smaller voxels and then dilated by a quarter-sized voxel, as illustrated in Figure 9.1, then the resulting voxel array can be used to construct a manifold B-Rep mesh. However, the resulting representation will suffer from having a larger volume than the original volume.

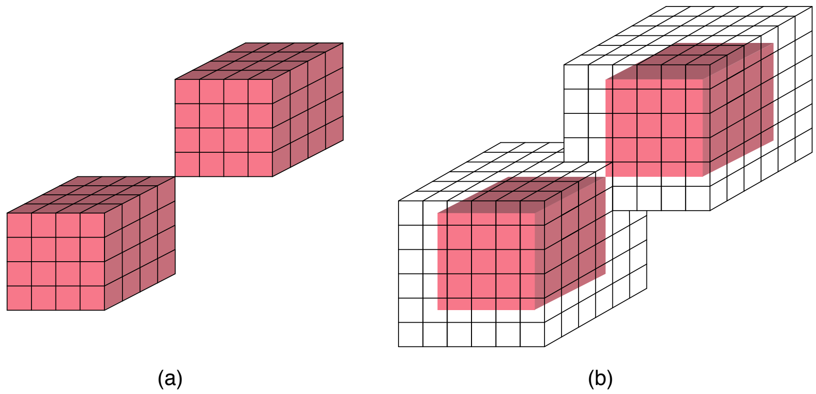
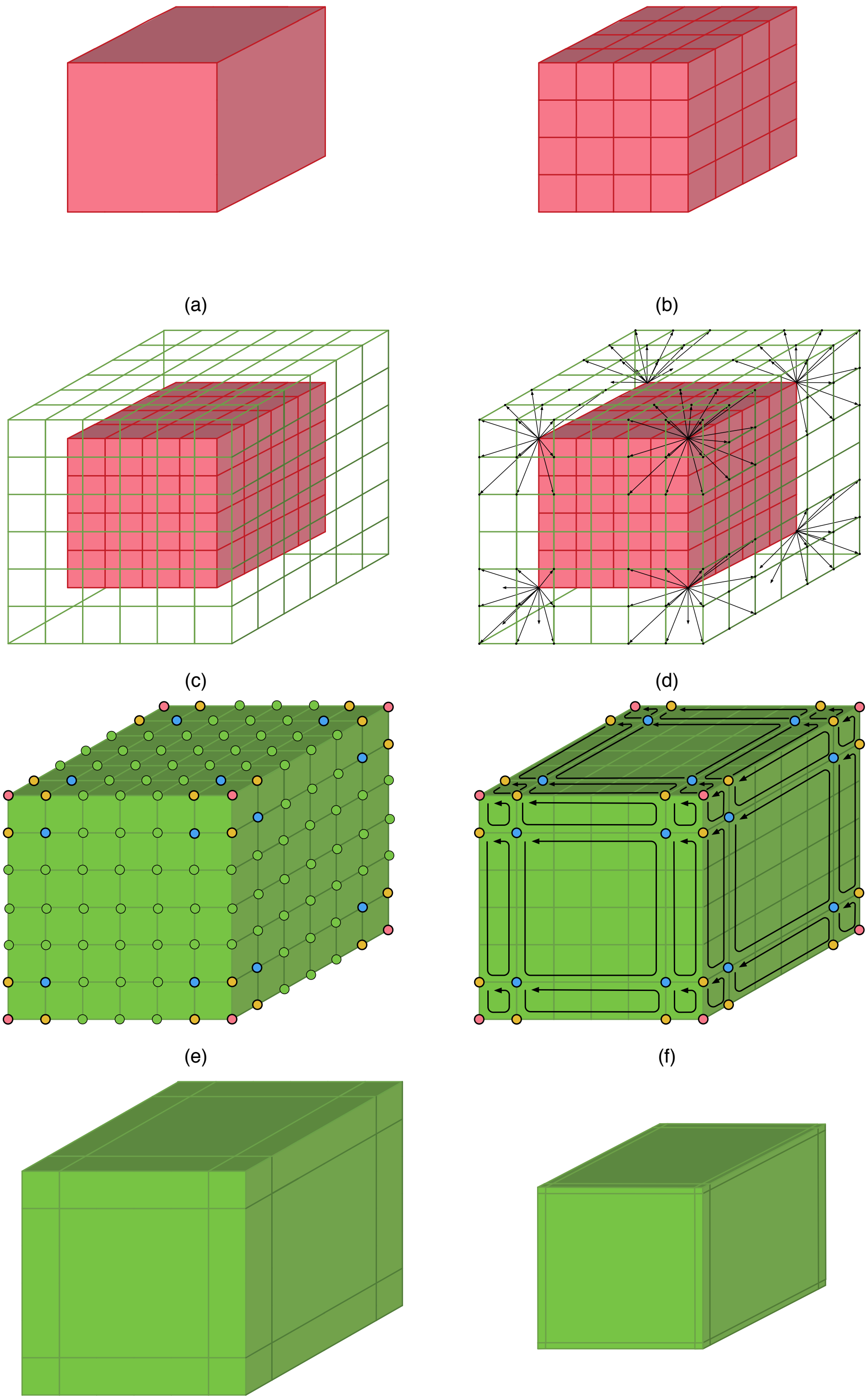


Figure 9.1: Subdivided and dilated non-manifold voxels

The effect of this enlargement can be minimised by amalgamating smaller faces into larger faces so that only the significant vertices found on the corners remain, as illustrated in Figure 9.2. Once the voxel array has been subdivided and dilated (Figure 9.2b) and converted to a mesh (Figure 9.2c), the significant vertices, coloured in red, blue and yellow in Figure 9.2e, are identified. The smaller faces bounded by non-significant vertices can then be replaced by larger faces (Figure 9.2g), by travelling around the significant vertices (Figure 9.2f). The position of every vertex in the mesh can now be expressed as a vector originating from a vertex on the original array (Figure 9.2d). If the distance of these vectors is scaled down using a minuscule factor, the volume of the mesh can be minimised while still maintaining the topology of the mesh.

This method is used to create watertight B-Rep from the voxel arrays generated in Chapter 8. The positions of the mesh vertices are calculated to be a distance from the original array that is 1 per cent of the size of the voxel (i.e. 2.5 mm).



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 9.2: Conversion to manifold B-Rep surface

9.3 Results

9.3.1 Spatial Join Confidence Check

Prior to carrying out a spatial join of MEP features and spaces, a confidence check was carried out to prove that *Oracle Spatial* is a suitable platform from which to perform spatial join operations.

For this confidence check, two tables were created one containing electrical assets extracted from *LPL-E-1-42201* (see Appendix A.3) and another containing simple three-dimensional (3D) spaces extruded from the *Broadgate Ticket Hall* floor plan (see Appendix C.1). A 3D spatial index was built for each table. A spatial query was then performed on a single feature using the SQL statement in Figure 9.3 and the operation was timed.

```
SELECT a.ELEMENT_ID, a.SUB_ID, b.SPACE_ID FROM
(SELECT * FROM MEP_FEATURES WHERE ELEMENT_ID='1') a,
EXTRUDED_SPACES b
WHERE SDO_ANYINTERACT (a.GEOM, b.GEOM) = 'TRUE'
```

Figure 9.3: SQL statement performing spatial join for single feature

The spatial query was performed on both a Desktop PC and on a virtual machine running within a *Macbook Pro* (see Appendix F for the specification of the different processors). Using *FME Workbench*, the same features and spaces were also loaded as *MultiPatch* features into *Esri ArcScene* running on the Desktop PC. The equivalent spatial query was performed using the *Inside3D* tool in the *3D Analyst* toolbox to compare the results, which are tabled in Section 9.3.1.

Table 9.6: Comparison of spatial query computational time

Element	Oracle (PC) (s)	Oracle (Mac) (s)	ArcScene (PC) (s)
Element 1	338.7	88.8	3.5
Element 2	283.8	30.6	3.6

From the results in Section 9.3.1, it can be seen that the *Inside3D* tool performed significantly faster than the *SDO_ANYINTERACT* tool by a factor between 10 and 100.

The difference in performance may be due to the different storage media used. The database files for the PC installation of Oracle are stored on a 1 Tb Hard Disk Drive (HDD) whereas the database files for the virtual machine installation on the *Macbook* are stored on a 300 Mb Solid State Drive (SSD). The superior read/write access times of the SSD provides a reasonable explanation as to why the *Macbook* was faster. Meanwhile, the geodatabase files used for the *ArcScene* installation on the PC are stored on 300 Mb SSD rather than the 1 Tb HDD. Furthermore it is expected that tools within *ArcGIS* are capable of caching data in Random Access Memory (RAM) providing further performance improvements. In light of the superior performance experienced using the *Inside3D* tool, the *Spatial Information Platform* was switched from *Oracle Spatial* to *Esri ArcScene* for the remainder of this investigation.

9.3.2 Spatial Join Operations

Once a suitable platform for performing spatial operations had been decided upon, the space sets and MEP feature sets were saved in an *Esri geodatabase* as *MultiPatch* features, because *Esri ArcGIS* is not capable of reading 3D objects directly from *Oracle Spatial*. Before proceeding, the space sets were tested using the *IsClosed* tool and their watertight status was confirmed. A spatial query using the *Inside3D* tool was performed for each configuration in Section 9.3.2 with the computation time recorded for each execution.

Table 9.7: Spatial join operations

Run	Element Set	Space Set	Time (s)
1	LPL-E-1-42201	BTH-EXTRUDED	55.0
2	LPL-E-2-42205	BTH-EXTRUDED	7.0
3	LPL-E-1-42201	BTH-MANIFOLD	167.0
4	LPL-E-2-42205	BTH-MANIFOLD	455.0
5	LPL-E-1-42201	BTH-DILATED	1250.0
6	LPL-E-2-42205	BTH-DILATED	575.0
7	MES-ME-4-00001	MES-EXTRUDED-11	0.6
8	MES-ME-4-00001	MES-MANIOLD-11	4.2
9	MES-ME-4-00001	MES-DILATED-11	4.1

The *Inside3D* tool outputs a table containing the *geodatabase featureset OBJECTID* for each feature that is either partially or fully inside a space together with the *featureset* primary key of the space. If a feature is inside more than one space, then a separate row is written for each space it is inside. An *FME* workspace was written to read the results, join the primary keys with the original feature sets to produce Section 9.3.2. This table lists the number of features identified within a single space according to that single space; it also lists the number of features identified as belonging to more than one space, and the number of features not identified inside any space. The features are coloured according to their identified space and visualised in conjunction with spaces used to perform the spatial join (Figure 9.4 to Figure 9.6); in these diagrams, features belonging to more than one space are coloured magenta, and features not belonging to any space are coloured black.

The results of the spatial join operation performed on the *Mile End Shaft* features and spaces was also processed using an *FME* workspace. The results are tabulated in Section 9.3.2 and illustrated in Figure 9.7 to Figure 9.9. Enlarged versions of these illustrations are reproduced at Appendix H.

Table 9.8: Spatial join results (Broadgate Ticket Hall)

Run	Element Set	Space Set	1	2	3	4	5	> 1	None
1	LPL-E-1-42201	BTH-EXTRUDED	5	0	177	90	3	24	71
2	LPL-E-2-42205	BTH-EXTRUDED	38	6	162	56	0	25	283
3	LPL-E-1-42201	BTH-MANIFOLD	0	24	75	13	0	0	258
4	LPL-E-2-42205	BTH-MANIFOLD	89	4	185	162	0	43	87
5	LPL-E-1-42201	BTH-DILATED	0	47	154	81	4	38	46
6	LPL-E-2-42205	BTH-DILATED	77	4	210	164	4	79	32

Table 9.9: Spatial join results (Mile End Shaft)

Run	Element Set	Space Set	Stair	Lobby	Rm A	Rm B	Rm C	Rm D	> 1	None
7	MES-ME-4-0001	MES-EXTRUDED-11	5	5	1	2	2	1	0	32
8	MES-ME-4-0001	MES-MANIFOLD-11	10	4	1	2	2	1	0	28
9	MES-ME-4-0001	MES-DILATED-11	0	5	1	2	2	0	1	37

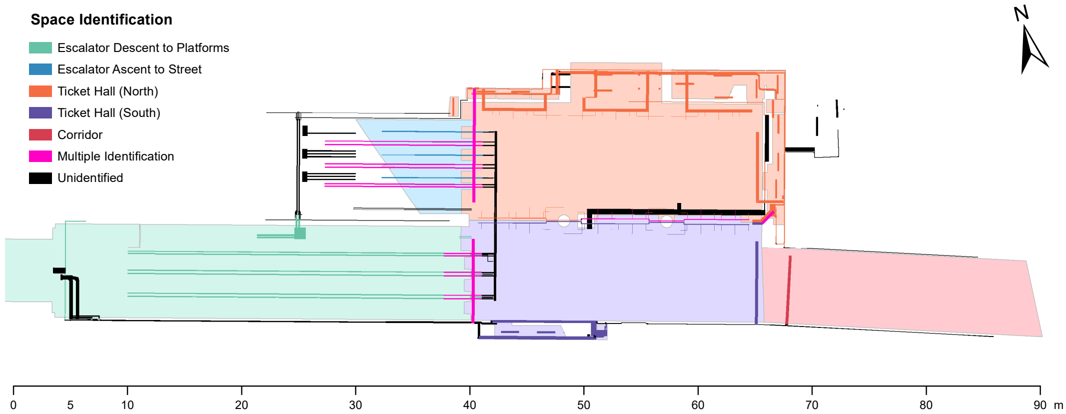


Figure 9.4: MEP features joined with BTH-EXTRUDED space set

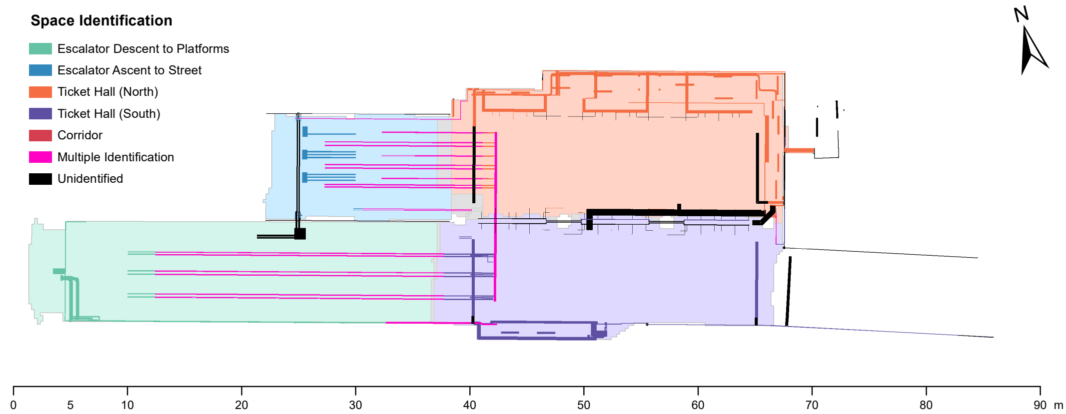


Figure 9.5: MEP features joined with BTH-MANIFOLD space set

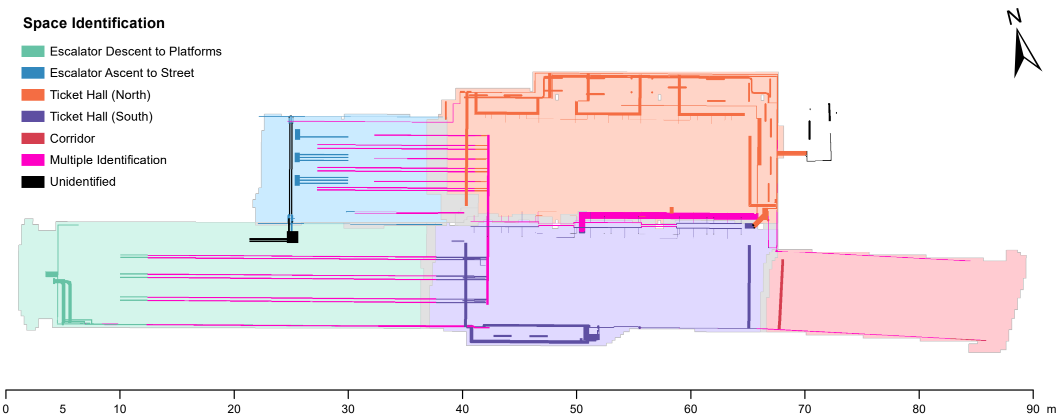


Figure 9.6: MEP features joined with BTH-DILATED space set

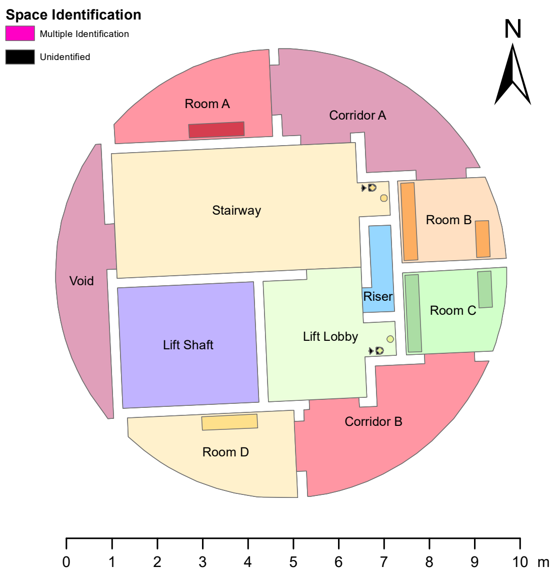


Figure 9.7: MEP features joined with MES-EXTRUDED-11 space set

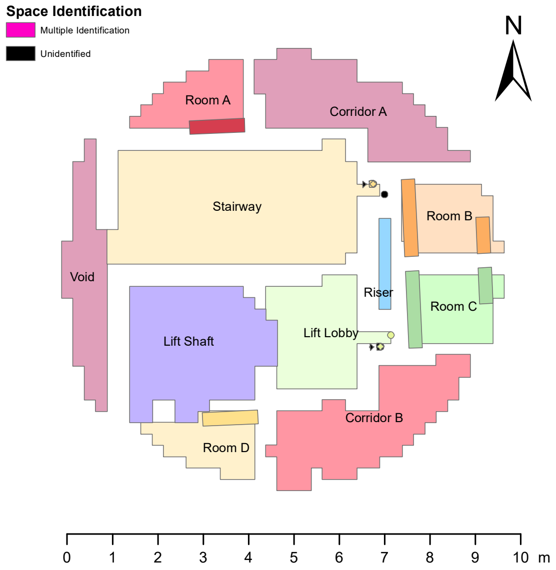


Figure 9.8: MEP features joined with MES-MANIFOLD-11 space set

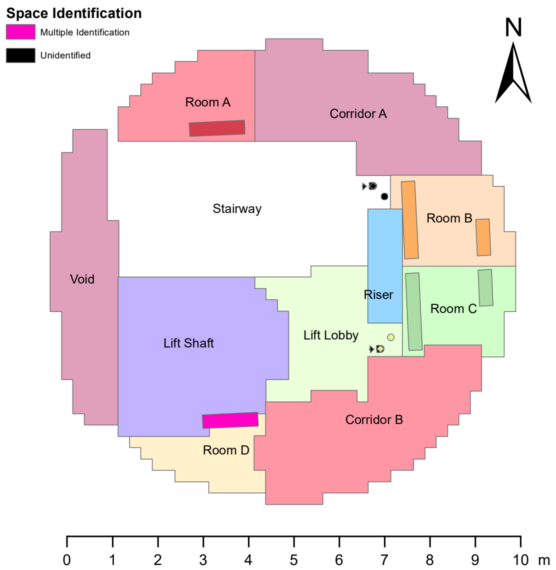


Figure 9.9: MEP features joined with MES-DILATED-11 space set

9.3.3 Observations

It was found that the *Inside3D* tool is not capable of handling *MultiPatch* objects that are not contiguous within a single feature. As a consequence, it was not possible to use the tool on the *BTH-AGGREGATED* space set. This issue was overcome by performing the spatial join on the *BTH-EXTRUDED* space set and aggregating the results afterwards.

When using the *BTH-EXTRUDED* space set, it should be noted that only half of *Space 2* (the *Escalator Ascent*) is represented in the floor plan. The space is divided in two by a zig-zagged line that loses one half of the space in the level above, the plan of which is not available. This depiction is the standard convention for representing staircases and elevators in floor plans. As a consequence, a significant number of features located in this space were not identified as such when using the *BTH-EXTRUDED* space set. This is a disadvantage of re-purposing information for use in situations that it is not originally intended.

During construction of the *BTH-MANIFOLD* space set, the algorithm used to create a manifold mesh from the voxel array failed while processing *Space 5* (the *Corridor*), the reasons for which have not yet been determined. Rather than reject the *BTH-MANIFOLD* space set, it has been decided to continue using the other valid spaces, as relatively few features are located within *Space 5*.

The two-dimensional (2D) visualisation of the MEP features coloured according to the space by which they have been identified (Figure 9.4 to Figure 9.6) is suitable for providing an overview, but the portrayal of 3D information in a 2D plan results in a loss of information. For a better understanding of spatial relationships it is necessary to consult 3D visualisation on-screen.

From Figure 9.4, it can be seen that the *BTH-EXTRUDED* space set has correctly identified the space of the majority of the features. As can be expected, features that cross over from one space to another are identified as belonging to more than one space.

There are a considerable number of features that are attached to walls but are not identified as being in the space because the floor plan does not cover the full extent of the space. There are also many features that are located above or below the space that have not been identified.

There are also features located adjacent to *Space 2* (the *Escalator Ascent*) that have been identified as belonging to *Space 1* (the *Escalator Descent*) as they fall within the 2D footprint of that space. Both of these cases provide a good example of the concerns originally illustrated in Figure 2.1.

Checking out Figure 9.5, it can be seen that more features have been identified as belonging to a space, although there are still some that have not been identified, in particular the features located on the ceiling of *Spaces 3* and *4* (the *Ticket Halls*). Moving on to Figure 9.6, it can be seen that even more of the features have been identified as belonging to a space.

One consequence of using the *BTH-DILATED* space set, is that more features are identified as belonging to more than one space. This is because the *BTH-DILATED* spaces overlap as a consequence of dilating the original *BTH-MANIFOLD* spaces. There is therefore ambiguity in the *BTH-DILATED* space set, in contrast to the *BTH-MANIFOLD* set which is mutually exclusive. Any assets finding themselves in this overlap zone will be counted as belonging to more than one space. This problem can be overcome by writing an algorithm to choose the best space to which overlapping voxels should be allocated and further work is required to implement this (See Further Work FW6 in Chapter 11).

From the output of the spatial queries performed in Section 9.3.2, it can be observed that the 3D spaces generated using *watershed segmentation* are marginally better for identifying the location of elements than the two-and-a-half-dimensional (2.5D) spaces created using *floor plan extrusion*. This superior detection comes at the expense of lengthy computational time. Although computational time is not necessarily an important factor, it may need to taken into consideration given the scale of the Crossrail CAD model and the many thousands of assets and spaces contained within it. As might be expected, performing a spatial query on simple 2.5D spaces with a low number of faces is faster than performing the same query on intricate 3D spaces that have a high face count.

It is interesting to note that there does not appear to be any pattern concerning the relative times of the two MEP feature sets. It is difficult to provide an explanation as to why *Run 1* (*LPL-E-1-42201* with *BTH-EXTRUDED*) take eight times longer than *Run 2* (*LPL-E-2-42205* with *BTH-EXTRUDED*). Likewise, given that *Runs 1, 3* and *5* have fewer features than *Runs 2, 4* and *6*, it is interesting to note that *Runs 1* and *5* are respectively slower than *Runs 2* and *6*, although *Run 3* is faster than *Run 4*. This may be due to the algorithm picking up "*easy win"* intersections that are not readily apparent in *Run 4*.

The spaces that belong to the *Mile End Shaft* have a different form of complexity to the *Broadgate Ticket Hall* spaces. As well as being smaller, they are simpler than the *Broadgate* spaces in that they only contain vertical sides and horizontal floors/ceilings. The spatial join operations are simpler because there are less MEP features to analyse, but analysis is more complicated due to there being three levels of spaces. This last aspect was overcome by selecting only a single level of spaces with which to perform the analysis.

From Section 9.3.2, it would appear that the *MES-MANIFOLD-11* set of spaces performed better that *MES-EXTRUDED-11* however it must be remembered that the *Stairway* space in *MES-MANIFOLD-11* extends over three levels while the same space in *MES-EXTRUDED-11* is limited to *Level 4*. As such, *MES-MANIFOLD-11* identifies features located on *Levels 5* and *6*.

The *MES-DILATED-11* set of spaces in missing the Stairway space (Figure 9.9) because this space was not successfully imported into *ArcScene* as a closed watertight space, and the results in Section 9.3.2 reflect the absence of this space. Because the spaces in *MES-MANIFOLD-11* do not cover the same floor area as *MES-EXTRUDED-11*, this set of spaces fails to identify the second MEP feature rising through the Stairway space, however, if this space had been successfully imported into the *MES-DILATED-11* set, the MEP feature would have been identified.

Overall, the *MES-EXTRUDED-11* spaces created from the floor plans have been more successful at correctly identifying the location of MEP features within the *Mile End Shaft* spaces. Furthermore, the *MES-EXTRUDED-11* spatial join operations performed faster than the more complex voxel-based spaces. It can therefore be concluded that the *MES-EXTRUDED* spaces provide preferential results.

9.4 Summary

In summary, this chapter has shown that it is possible to use the spaces created using *floor plan extrusion* and *watershed segmentation* methods in Chapter 8 for performing spatial operations on the MEP features extracted from *MicroStation* in Chapter 7. Using these spatial operations, it is possible to join each MEP feature with the Identifier (ID) of the space in which it is located. When performed on a complex set of spaces such as the *Broadgate Ticket Hall*, it would appear that the 3D spaces created from a single-voxel dilation of a segmented and merged voxel array are more successful at identifying location of assets; however, this superior identification comes at the expense of longer computational times. This advantage is, however, lost when performing spatial join operation on simpler space sets such as those found in the *Mile End Shaft*.

10 Discussion

10.1 Addressing the Research Questions

This research began with a presumption that Building Information Modelling (BIM)/Geographic Information Systems (GIS) integration challenges are holding back the full potential of BIM with respect to its ability to increase efficiency, reduce expenditure and cut carbon expenditure. One such challenge identified for consideration in this thesis is the management of asset information on completion of construction and in expectation of handing over to the owner/operator.

In Chapter 2, this thesis proposed to use the Technical Information Systems at Crossrail as a case study into the challenges experienced by a major infrastructure project and ask:

> **RQ - Can a better understanding of the conceptual and technical challenges to the integration of BIM and GIS provide improved support for the management of asset information in the context of a major infrastructure project?**

This first half of this chapter discusses each of the supporting questions in turn before discussing and answering the principal research question. A general discussion of other observations and associated matters of interest shall then follow in the second half. Forward references are made in this chapter to the recommendations, contributions and further work outlined in Chapter 11.

10.1.1 Supporting Question 1

In Chapter 2, the first supporting question was asked:

> **SQ 1 - Can a novel Spatial Information System Framework be developed to identify and classify interoperability issues that currently hinder the management of asset information in the context of a major infrastructure project?**

The requirement to develop a novel integrated Spatial Information System Framework, such as the one proposed in Chapter 4, came about from a realisation that both GIS and BIM are socio-technical frameworks (Section 2.1.1). In 2015, as this research project was getting underway, a concerted effort was being made within the UK BIM community to educate the Architectural, Engineering and Construction (AEC) sector on the importance of *Building Information Management* (Section 4.4.6.1) and standardisation of the practice through the UK 1192 suite of specifications, as listed in Table 3.1. The separation of BIM into the practice of *Building Information Modelling* and *Building Information Management* (Section 4.2) prompted questions to be asked as to whether GIS was separated in similar ways, and how was geospatial information managed when working under the mandate of the UK 1192 suite.

This line of enquiry led to a review of general information system frameworks and the Levels of Conceptual Interoperability Model (LCIM) advocated by Tolk, et al. (2007) in Section 4.1. The decision to include a review of general information systems theory was taken so as to maintain a neutral position with regard to BIM and GIS.

When these frameworks were observed alongside other BIM-related and GIS-related frameworks in Figure 4.1, a hierarchical structure became apparent starting from the computational platform at the bottom to the disciplinary field at the top. From this hierarchy, a framework was developed in Section 4.3 that can be applied to both BIM-based systems and GIS-based systems. The harmonised nomenclature enables hierarchical components in each system to be compared like-for-like. This ability to compare systems is further enhanced by the ability to visualise the framework as a graphical representation (Figure 4.15).

Amirebrahimi, et al. (2015) and Kang and Hong (2015), among others, had already begun creating a taxonomy of the various approaches that have been taken in achieving BIM/GIS integration within academic literature. It was found that it is possible to adapt the diagrams in Figure 4.15 so as to illustrate how these various approaches to BIM/GIS integration are implemented. This is a significant benefit to the field of BIM/GIS integration as it enables individuals to visualise problems and propose solutions through the use of diagrams, not only at a technical level but also across the full socio-technical spectrum; for example, the diagrams can be used to map the activities to tools, models and platforms within a management protocol.

The framework was successfully applied to Technical Information Systems in use within Crossrail in Chapter 5. Each system was broken down and described at each level starting with the Spatial Information Environment in which the information is held, and working up to the Spatial Information Management protocols used to control the information, and beyond to the Spatial Information Disciplinary Field of the people who use those protocols. Using this approach, it was then possible to describe the heterogeneities at each level in Section 5.2 and thus identify the interoperability issues at play. This exercise has been a valuable and this thesis recommends its widespread adoption as a means of understanding how spatial information systems interact (Recommendation R1).

Strengths, Weaknesses, Opportunities and Threats (SWOT) analysis is a well-established management tool for analysing the marketability of a product (Zack 1999). It provides convenient headings with which to summarise the strengths, weaknesses, opportunities and weaknesses of the framework (Figure 10.1).

The most critical strength of the framework is that it has been developed from existing Information System frameworks, such as the Reference Model - Open Distributed Processing (RM-ODP), LCIM, and the *semiotic ladder* as already stated in this section. The framework has been qualitatively assessed by applying it to the integration approaches found in literature and the Crossrail Technical Information Systems. A further strength of the framework is that it lends itself well to graphical representation.

One of the biggest weaknesses is that there are nuanced distinctions between some of the levels in the framework, in particular between the *Environment* and *Platform* levels and between the *Activities* and *Management* levels. It is also felt that the alignment between the *Disciplinary Field* and *Conceptual* and *Dynamic* interoperability levels in the LCIM is not as strong as lower technical interoperability levels.

It is felt that the framework can be applied to any spatial information system to analyse the interoperability of that system with another spatial information system. Indeed, following the application of the Framework to Crossrail Asset Information Management System (AIMS), there may be other opportunities to apply the framework to quasi-spatial and non-spatial information systems. The framework provides opportunities for analysts to assess the system heterogeneities and subsequent interoperability challenges at each of the seven levels. Furthermore, the hierarchical nature of the framework encourages analysts to consider how conceptual differences have an effect on interoperability challenges.

One of the biggest barriers to using the framework is that it requires analysts to have an in-depth understanding of each level due to some of the nuanced distinctions between the levels, especially as some systems may be more difficult to decompose into levels than others.

Further work is required to obtain an independent assessment of the Framework by requesting feedback from industry and academic subject matter experts in the field of BIM and GIS. This feedback would provide material with which to extend the SWOT analysis in Figure 10.1 (Further Work FW1)

10.1.1.1 Application of Framework to AIMS

One of the first benefits of using the Spatial Information System Framework in this research was extending its application to AIMS in Section 5.1. Although the framework was not initially developed with AIMS in mind, this system could be broken down into the same framework levels as BIM and GIS.

If BIM-based *MicroStation* and AIMS are both considered as Spatial Information Systems, then it quickly becomes apparent that each employs different strategies for storing location information. In the former, Computer Aided Design (CAD)-based elements are fully

Strengths	Weaknesses
Framework developed from existing Information System frameworks e.g. *RM-ODP, LCIM, Semiotic ladder* Graphical representation for clear visualisation Applied to interoperability cases taken from literature Applied to Crossrail Technical Information Systems	Nuanced distinction between levels i.e *Environment* vs *Platform* i.e. *Activities* vs *Management* Weak alignment between *Disciplinary Field* level with *Conceptual* and *Dynamic* interoperability levels
Opportunities	**Threats**
Framework applicable to any spatial information system Possibility to apply framework to quasi-spatial and non-spatial information systems Framework encourages analysts to consider interoperability at each level Framework encourages analysts to consider how conceptual differences affect interoperability challenges	Training/experience required for analysts to have a comprehensive understanding of the each level in order to use framework Some systems may be more difficult to decompose into levels

Figure 10.1: SWOT analysis on Spatial Information System Framework

represented in Euclidean space as *SmartSolid* objects; on the other hand, in AIMS, the general position is only located using the name of the space in which they are contained. This difference in specifying location results in the information sets being non-interoperable and thus it is a challenge preventing integration.

By laying out the interplay of the various systems in a graphical form, it is possible to see patterns that might have otherwise remained hidden. For example, Figure 5.7 shows that spatial information with the same semantics and schematics, but not necessarily the same syntax, can be held on two difference platforms. Figure 5.7 also shows that the tools in the GIS platform cannot work with information in AIMS due to differences in the way location is represented. However, the information on all three platforms is managed by the same BIM Execution Plan at the *Spatial Information Management* level.

10.1.1.2 Understanding Conceptual Heterogeneity

A second benefit of using the Framework, is that it forces practitioners to think about higher-level conceptual differences as well as lower-level technical differences. These conceptual differences exist due to differences in the purpose that the information is intended to be used for and the cultural differences of the people who use the information. It has already been established in Section 3.3 that conceptual and semantic differences at the higher-levels have a cascading effect on system heterogeneity at the lower levels.

Spatial Information Activities conducted during design and construction are very different from the *Activities* conducted after handover. As the information is required to support different enterprise activities, the pragmatic differences at this level then have a cascading effect on the breakdown structures and schemes of classification at the technical levels. The publication of *Spatial Information Management* standards (i.e. the UK 1192 suite and ISO 19650 suite) that cover both the delivery and operational phases is a significant contribution to avoiding interoperability issues and working towards better integration.

Before information from one system can be used in a second system, a question must always be asked as to whether that information is still relevant when it is interpreted in that second system. Information must be recast down to the "*lowest common denominator"* before it can be mutually recognised by another system. Previous case studies have recommended that Asset/Facilities Management (AM/FM) practitioners set out their information requirements at the start of the design stage so that design contractors can provide information that will be fit-for-purpose when adopted by AM/FM systems (Thabet, et al. 2016; Lavy and Jawadekar 2014).

Crossrail had the foresight to embed Asset Management (AM) teams and implement its AM strategy from the start of the design phase in 2008 (Taylor 2017). It also made a decision at that time to not embed asset information into the Project Information Model (PIM)/Asset Information Model (AIM) CAD models. This decision was made to keep the model file size down and ensure a degree of interoperability between the *DGN* and *DWG* formats (Taylor 2017). It would appear that this decision did not foresee the loss of the relationship between information in the CAD models and AIMS.

Another example of how information requirements affect the make-up of spatial information is the explicit/implicit space paradigm described in Section 3.9. Spaces are used less in the delivery phase because the information requirements are focussed on supporting construction activities. The contractor, tasked with designing the construction, might not understand the importance of explicitly representing the space and, therefore, might not invest in the creation of quality spaces (Recommendation R3).

10.1.1.3 Role of the Spatial Information Disciplinary Fields

The final consideration to be discussed in this subsection is the role of the *Spatial Information Disciplinary Field* (Section 4.4.7) on the interoperability of information. The lengthy process of joining the professional body that makes up a disciplinary field, imparts a certain amount of shared knowledge to a practitioner of that field (Obermeyer 1994). Taking into account that higher-level conceptual differences have a cascading effect on lower-level interoperability, when information is communicated between people of the same disciplinary field, there might be an implicit meaning shared between them that does not need to be communicated. When communicated to someone outside the field, this implicit meaning is not shared; if the information is interpreted differently then this will be an interoperability issue.

In the graphical representation of the Spatial Information System framework in Figure 4.15, the *Spatial Information Disciplinary Field* has been drawn up surrounding the *Spatial Information Management* and *Spatial Information Activities* level. This gives the impression that the activities at these levels are carried out by a homogeneous discipline of professionals. In reality, these levels consist of multi-disciplinary teams, as depicted in Figure 5.7, who will interpret information differently depending on their background.

For the reasons discussed above, the development of a Spatial Information System Framework as asked by Supporting Question 1 has been a great help in gaining a better understanding of how interoperability issues arise between different systems. In the context of asset information management within Crossrail, it has proved its worth by providing a way to visualise how the three different information systems are interwoven. The Spatial Information System Framework, including its graphical representation, has the potential to be a helpful tool in describing the composition of and relationships between BIM and GIS systems and it is considered to be a significant contribution of this thesis (Contribution C1).

10.1.2 Supporting Question 2

As well as developing a framework to understand the various interoperability issues at play, this thesis also presented an investigation into the interoperability challenges at Crossrail. One particular challenge chosen as a case study is the lack of explicit relationships linking CAD-based elements and AIMS-based assets (Section 2.1.2). The proposed method for building links requires both elements and assets to be identified according to the name of the space in which they are contained.

Before embarking on this endeavour, it was recognised that there are three prerequisite steps that must first be undertaken (Figure 2.3). The first of these steps to be discussed in this subsection (PRS 1), is to perform an Extract-Transform-Load (ETL) operation on the CAD-based elements; the second (PRS 2) and third (PRS 3) steps will be discussed in Section 10.1.3.

This research benefits from having to take these steps because they each have their own interoperability challenges that require investigation. With regard to the first step, a questions is asked as to:

> **SQ 2 - What challenges are frustrating reliable ETL (extract, transform, load) operations between CAD-based design models and a GIS-based spatial data warehouse? How can these be overcome?**

Based on the work conducted in this research, Table 10.1 provides a list of the challenges frustrating ETL operations between *MicroStation* to a GIS-based format. The table includes a description of how the challenge was overcome in this investigation together with a potential long-term solution that may fix the source of the problem.

Table 10.1: Challenges frustrating ETL

Challenge	Short-term solution	Long-term solution
Closed access	Use third-party formats for exchange	ETL software to include more formats
Occasional element loss	Account elements with MVBA script	Report issues to software provider Authors to provide element catalogue
Inclusion of construction elements	Account elements with MVBA script	Report issues to software provider Authors to provide element catalogue
Occasional geometry deformation	Compare geometry across formats	Report issues to software provider
Multi-part elements	Handle parts as features	Handle parts as features Aggregate parts into elements
Element identification	Expose DWG Element ID Use level-renaming workaround	N/A
Fragility of solid geometry	Switch to alternative export format	Determine need for solid geometry
Degradation of geometric representation	N/A	Determine need for solid geometry
Endemic hindrance	Perseverence, time and money	Perseverence, time and money

10.1.2.1 Closed Access

The first challenge that frustrates ETL operations is the use of proprietary software, in this case *Bentley MicroStation*. Indeed, if *MicroStation* had been more accessible, through well documented high-level Application Programming Interfaces (APIs), it might have been possible to perform more of the operations described in this thesis within the *MicroStation* Platform; as it is, it has been necessary to extract elements for analysis in other platforms.

The closed nature of the *MicroStation* proprietary format also means that the ETL software application, *FME Workbench*, is unable to read three-dimensional (3D) *SmartSolid* elements directly from the *MicroStation DGN* format (Section 6.2.2). It is, therefore, necessary to extract elements using a third-party format supported by *MicroStation* and the BIM authoring extension as described in Section 6.5.

10.1.2.2 Occasional Element Loss

The next challenge concerns the occasional loss of elements when exported via third party formats, such as Industry Foundation Classes (IFC) and *Trimble SKP*, but not *AutoCAD DWG* (Section 6.2.5 and Section 6.2.6). No software can be expected to be totally perfect, and all reputable software publishers will have a quality assured process for resolving issues. It is important to document instances where elements are lost and provide feedback to the software provider.

Because engineering systems, such as underground railway stations, support safety-critical activities, there must be a quality assurance process to ensure that the elements extracted from the CAD environment can be relied upon to make safety-critical decisions. In this case study, the *MicroStation Visual Basic Application (MVBA)* script was used to identify every element in the model file and account for each element after extraction (Contribution C2).

A study of the *Bentley* community support forum has found several reports of missing elements when exporting to STL file format (STL) (Bell 2015) and STEP file format (STEP) (Zigelski 2015). These are relevant as STEP is the foundation format for IFCand STL is another mesh format. The *Bentley* response to the STL report was a recommendation to use an alternative to the *Surface* geometry type, which is an unhelpful recommendation when applied to *AECOsim* as the BIM extensions have no option for changing geometry type. The response to the STEP report was a warning to be mindful of geometry units and the distance from the origin. This response infers that they may be a problem involving floating-point geometry errors when using large valued coordinates to represent small faced elements.

10.1.2.3 Inclusion of Construction Elements

A significant number of construction elements were incorrectly extracted from the CAD model files by the IFC export tool, despite the element being tagged as a construction element in *MicroStation* (Section 6.2.5). These correctly tagged construction elements were also joined by construction elements that had been inadvertently tagged to be part of the model by the author.

In a similar way to the previous challenge, the mistaken inclusion of these elements the extraction may lead to erroneous decision-making concerning safety-critical activities. It is therefore essential that all construction elements are tagged as such and a quality assurance method, such as the *MVBA* script, be used to prevent these elements being included in the extraction.

Design contractors should be made aware that CAD model files will be used for ETL operations in the Employer's Information Requirement (EIR) and consideration should be given to making contractors check the model once it has been extracted and transformed into an alternative format (Recommendation R2). It will be a great benefit if the model designer produces an authorised catalogue of the elements that contribute to the model. Construction elements should ideally be deleted if they are not required; if it is necessary to keep them in the file, for whatever reason, then they should be identified as such and not included in the catalogue of elements.

10.1.2.4 Occasional Geometry Deformation

After running the automated ETL operation, as described in Chapter 6, it was discovered that the geometry of a single element had become significantly deformed when exported via IFC (Section 6.2.5). Instances like this, similar to the missing elements and mistakenly included elements in the previous challenges, need to be reported back to the software provider as part of their quality control system.

This deformation was discovered though a manual inspection of the output, and thus might not be detected in a large batch operation. In the workflow followed in Chapter 6, geometry was exported using four different methods. As well as providing a source of geometry for replacing features, exporting the models in *DWG* and *SKP* formats provides a method for detecting deformed geometry such as the element in Figure 6.2. It should be possible to devise an algorithm that compares the geometrical output of each of the four exports by comparing a property such as minimum bounding volume or surface area. If the geometric property of the element exceeds a tolerance, then the element should be flagged for manual inspection and for a solution to be put in place to overcome the deformed geometry (Further Work FW2).

A study of the *Bentley* community support forum has found a report of phantom elements appearing in MicroStation (Wallace 2013). The response from *Bentley* explains that this is due to the imperfect execution of a cutting operation, causing some geometry to be left hanging; in most situations, the geometry is cleaned up, but occasionally these hanging elements are not correctly resolved. The geometry under scrutiny at the bottom of the Mile End Shaft floor contains a circular element that has been forced into a planar mesh during the IFC export. It is hypothesised that an imperfect cut followed by conversion to a planar mesh has not been successful in this operation.

10.1.2.5 Multi-part Elements

On initial investigation, one particular challenge is related to how some CAD elements are split into multiple parts when exported in the *AutoCAD DWG* or *Trimble SKP* formats (Section 6.2.4 and Section 6.2.6). Once this phenomenon was understood, these multiple parts and the remaining single-part elements were handled as accordingly using the term *feature*, with each feature being identified by its *Element ID* and a *Part ID*. Practitioners should be aware of this and handle the newly created parts as features or reunite the parts as an aggregated element.

10.1.2.6 Element Identification

Every element in a *MicroStation* CAD file is represented by an *Element ID* unique to that file. When this *Element ID* is combined with the model file name, every element in the project can be uniquely identified and traced back to its source. When the model file is exported in the IFC format, the *Element ID* and file name are used to populate the *Description* attribute (Section 6.2.5).

Identifying elements after they have been exported via the *AutoCAD DWG* and *Trimble SKP* formats is not as straightforward as exporting via IFC. Elements that have been exported to the *DWG* format still keep their original *Element ID* from *MicroStation*, but it is hidden within the *AutoCAD Block Entity* and must be exposed (Section 6.2.4). If practitioners are not aware of this undocumented procedure, a challenge to integration arises.

On the other hand, elements that are exported to the *Trimble SKP* formats are not attributed with the *MicroStation Element ID*. They are, however, attributed with the name of the level (or *layer* using *AutoCAD* terminology) to which they belong. Passing the *Element ID* can be overcome using a workaround, whereby the element is moved to a new level in *MicroStation* which remains attached to the element in *SKP* (Section 6.2.6). The procedure, however, is not perfectly reliable and so an alternative method using geometric properties can also be used to link elements with geometry exported via other formats.

10.1.2.7 Fragility of Solid Geometry

All of the elements created in *MicroStation*, with limited exceptions, are solid objects. In order to represent a valid solid, Boundary Representation (B-Rep) geometry must satisfy a prescribed list of criteria, such as having planar faces that do not intersect and having every edge being shared by only two faces, among other criteria (Section 3.5.1).

After these elements had been extracted from *MicroStation* and read into *FME Workbench*, it was found that many elements no longer satisfied the criteria that are required for the geometry to represent a solid (Section 7.1.5). Indeed, some elements exported via the IFC format did not even satisfy the criteria for a valid surface. In these instances, it was possible in nearly all cases to swap invalid solids or surfaces from an alternative geometry source.

A basic inspection of features revealed that the most common cause of invalid solid geometry was due to triangles only having two distinct vertices. It is suspected this fault is caused when polygon faces with discretised curves are triangulated using the *fan* method resulting in long and extremely thin triangles. It is then suspected that floating-point numerical errors arise during geometry operations that result in an invalid planar surface. Attempts to repair these faulty surfaces using the *Geometry Validator* transformer in *FME* result in a valid surface but not a valid solid. The problems associated with *fan triangulation* could be avoided if *Delaunay triangulation* methods are used to convert polygons into triangular meshes (Shewchuk 2002).

The requirement to extract and transform elements as watertight solids was identified in Section 6.1 to support topological queries relating to intersection and containment. If maintaining the geometry of an element as a valid solid is troublesome, the question must be asked as to whether it is essential to maintain its solidity. If an object is valid solid then it is possible to perform certain functions such as volume calculation, and hence calculate the mass of an element. Valid solids are also important for performing boolean geometry, such as was used to create a clipped space from a dilated space created using watershed segmentation (Section 8.1.4). However, if the geometry is only being used for visualisation, or for voxelisation, it may not be important to maintain an element's solidity.

10.1.2.8 Geometry Degradation

There is a loss of geometric fidelity when exporting from *MicroStation* to a GIS-based format as described in Section 6.3. The fidelity of the representation after transformation varies according to the method used to export the geometry, with elements exported via *DWG* remain almost identical to the original *MicroStation* geometry. The loss of geometric fidelity was not an issue in this case study; however, there may be other cases where using reinterpreted geometry might cause problems. Take, as an example, a convex cylindrical surface that makes a perfect fit with a concave cylindrical surface in the original model; after reinterpretation, the two surfaces may no longer fit together.

10.1.2.9 Automation of ETL workflow

The ETL workflow described in Chapter 6 was specifically developed with automation in mind. A batch processing script was used to automate the export of model information from Bentley MicroStation, and workspace running was used to automate the transformation in *FME Workbench*. The MicroStation export functions, however, do not yet command sufficient trust for a user to stand back, although using three different export routes does provide the means for each geometry to be compared with two other geometries, thereby providing opportunities for manual intervention in the automated process. It is, however, frustrating that a routine export operation still requires manual oversight.

10.1.2.10 Endemic Hindrance

It is evident from the list of challenges documented above that extracting CAD elements from *MicroStation* is not a simple operation. Many hours of investigative trial and error were performed facing these challenges and identifying a workable solution. Initial investigations were frustrated by a bug in the IFC export function that practically closed off this route.

One of the challenges hindering integration is best described by the phrase *endemic hindrance*. The number of issues that must be resolved to achieve successful extraction, together with the associated time and expense, creates a barrier that is not commercially viable to cross. Although a bug only causes temporary interoperability, as it is likely to be resolved in the next update, this is not helpful in the present. With limited budgets and pressing time constraints, practitioners may reach the conclusion that there must be a cheaper solution and switch their efforts elsewhere.

A commercial project must recognise that extracting information from one environment for use in another carries considerable risk to the project. It may, therefore, be that the best commercial solution is to keep information within the same environment and accept the inevitability of vendor lock-in.

10.1.2.11 Longer-term solutions

In this thesis, a requirement has been identified to interchange geometric information from a BIM environment into a GIS environment to aid the acquisition and usability of AM information. There are, however, many other use cases that stand to gain from improved BIM/GIS integration, namely planning approval (Noardo, et al. 2019a), city modelling (Biljecki, et al. 2015), flood damage modelling (Amirebrahimi, et al. 2015) and development of digital twins (Whyte, et al. 2019). However, the BIM/GIS interoperability is still a significant challenge, with others having reported on the difficulties experienced extracting information sourced from *MicroStation* alone (Whyte, et al. 2019; Floros, et al. 2020).

Despite many initiatives to promote the use of IFC, many projects only consider the interchange of IFC as a desirable rather than an essential requirement. Because IFC is not core to a project, there is less endeavour to insist on reliable and trustworthy export of information via this format. Indeed, there is a vicious circle whereby IFC is not specified due to its risky nature, which leads to a reduced demand to invest in reliable interoperability. The long term solution must be to write detailed requirements into project specifications for interoperability of information from proprietary CAD to open source interoperability formats (Recommendation R2). BIM designers must then test all their models to prove that all elements can be exported via IFC into a GIS environment. This way, not only will more bugs and shortcomings in the export tools be identified and reported, but the project can escalate support and demand critical software updates.

10.1.2.12 Addressing Supporting Question 2

The workflow described in this subsection has achieved a complete extraction of the elements identified using the *MVBA* script, uploading them as non-watertight surface elements (Contribution C2). Furthermore, the workflow has had a high success at uploading watertight solid features from structural and architectural models to a spatial database. The uploading of electrical elements from Mechanical, Electrical and Plumbing (MEP) models has not been as satisfactory, but the need to extract watertight solid features from the MEP models is less important than with the structural and architectural models. The challenges hindering efficient ETL operations are summarised in Table 10.1 together with a short description of the solution implemented in this investigation and a recommended long-term solution.

10.1.3 Supporting Question 3

This subsection shall discuss the challenges relating to the second (PRS 2) and third (PRS 3) prerequisite steps and address the last supporting question:

SQ 3 - What methods exist for modelling complex spaces to locate assets using 3D spatial analysis? Can these be implemented?

Section 3.10 in the Literature Review described six methods for creating explicitly represented 3D spaces from building models. Three of these methods were selected in Chapter 8 and used to create practical spaces from the Crossrail CAD model elements. The outputs of two of these methods were then used in Chapter 9 for performing a spatial query with asset features to identify the name of the space in which those assets are located.

The *BTH* and *MES EXTRUDED* space sets was generated from floor plans (Section 8.1.1). The quality of these spaces is mostly dependent on the quality of the two-dimensional (2D) floor plans from which they are produced (see Recommendation R3). In addition, the representation is dependent on choosing a single vertical co-ordinate height to represent the elevation of the floor and a single height from which to calculate the elevation of the ceiling.

The *BTH* and *MES MANIFOLD* space sets was generated by performing a watershed segmentation algorithm on the building elements using the program described in Section 8.1.3. Out of the results from using nine combinations of voxel size and discretisation factor in Section 8.3, only one particular output was suitable for further use. Because the *MANIFOLD* spaces stop short of representing the full volume of each space, a third set of spaces, referred to as the *DILATED* set, was also formed from a one-voxel dilation of the voxels used to create the *MANIFOLD* sets (Section 8.2.11).

In the course of creating spaces a number of observations can be made which are discussed in the following subsections.

10.1.3.1 Manual Intervention

When creating the *MANIFOLD* and *DILATED* space sets, it was necessary to make a manual intervention in the process in order to create usable output sets. Further work is required to find automated solutions to replace these manual interventions (Further Work FW3) in order to scale up production of spaces across the whole estate of a project. If the process can be made to be fully automatic on subsequent runs (after manual intervention on the first run) then spaces can be regenerated following minor changes to the original model.

Interior Spaces – Firstly it was necessary to identify which segmentations represented the spaces inside the model and which segmentations were formed outside the bounds of the model (i.e. in the earth). In Chapter 8, a simple heuristic was used to make an initial attempt at classifying spaces as being inside or outside the building elements. The spaces were then inspected and corrected manually with the aid of an on-screen visualisation. Although this makeshift workflow was suitable for this research, a more appropriate automatic method would be required in a commercial workflow (Further Work FW3).

Orientation of Voxel Grid – In the case of the *Broadgate Ticket Hall* sets, the orientation of the voxel grid was manually aligned with the predominant orthogonal directions in order to avoid wasted space and reduce the number of faces created in the B-Rep conversion. Further work is required to detect the predominant orientation of model geometry (Further Work FW3).

Choice of Discretisation Factor and Voxel Size – Using a smaller discretisation factor in watershed segmentation method in Chapter 8 led to the creation of over-segmented spaces. The problem of over-segmentation is a well recognised problem in the literature; indeed (Beucher and Meyer 1993) advised that the effectiveness of the watershed transform is dependent on the choosing the correct methods of marker selection, and pre-processing of the input field and post-processing of the output field. Due to the over-segmentation it was necessary to manually intervene and join up segmentations to create spaces usable for spatial queries.

10.1.3.2 Conversion of Voxel Spatial Enumeration

All representations that use voxel spatial enumeration to represent solid objects or watertight spaces are at risk of being non-manifold if voxels touch along an edge or at a corner. The voxel spatial enumeration must therefore be reinterpreted using the method described in Section 9.2.2. It should be noted that the implementation is occasionally unsuccessful, and further investigation is required into why these occasional faults occur (Further Work FW5).

As part of the development of this method, the concept has been extended to convert multiple spaces into planar B-Rep mesh, and this has been successfully demonstrated on simple test cases, but it has not yet been possible to demonstrate this method on real-world spaces with confidence. Further work to resolve the issues encountered would overcome the problem of overlapping spaces in the *DILATED* sets .

In the mesh conversion algorithm, the mesh created is shrunk back to the volume of the original voxel array by manipulating the size of vertex vectors. Rather than reduce the size of the vertex vectors, a method whereby these vertex vectors are clipped by the nearest enclosing boundary element has been investigated. Using this method would produce a more representative watertight space that would not have the rough surface otherwise found with voxelated spatial enumeration. Again this method has been proven on simple test cases, but it has not yet been able to demonstrate this method on real-world spaces.

10.1.3.3 Challenges of Quantitative Assessment

The results obtained from performing spatial queries with assets and the space sets in Section 9.3.2 reveal that the concerns illustrated in Figure 2.1 (i.e. concave walls and multi-layered spaces) manifest themselves in the analysis of the *Broadgate Ticket Hall* model; for example, the assets located above the escalator descent have wrongly been allocated to that space, and assets fixed to the surface of walls fall outside the space boundaries of the floor plan. The quantitative and illustrated results show that the *BTH-DILATED* space set was most suited for identifying the location of assets, although it did suffer from finding instances where an asset is located in more than one space.

Without access to a benchmark of correctly identified asset locations, it is difficult to assess the quality of the spatial queries. It is only possible to compare the results of using each of the three space sets. Further research might develop a method to identify the proportion of an asset that lies within a particular space as a numerical value and attribute the space location according to the space in which the asset is predominantly located.

10.1.3.4 Coverage of Dilated Spaces

The method used to create the *DILATED* space sets is also impaired in that it is not able to represent the full extent of the space when reaching into complicated corners. This problem is illustrated in Figure 10.2 where a cylindrical column is situated in front of an alcove. In Figure 10.2b a voxelised space is created which is then dilated by one voxel in Figure 10.2c. The dilated voxel space is then converted to a B-Rep mesh and clipped with the original bounding elements. From Figure 10.2b, it can be seen that the newly created space does not reach into the alcove. This problem can be seen in the *Broadgate Ticket Hall* model around the top of the columns in Figure 10.3.

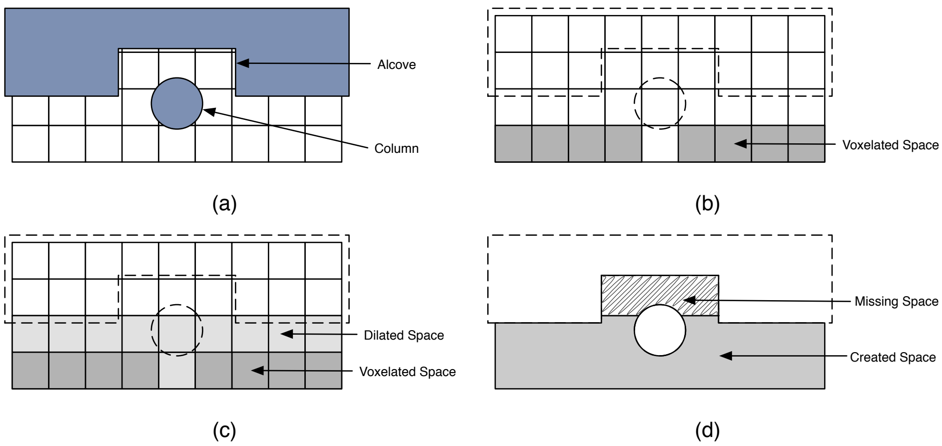


Figure 10.2: Diagram explaining missing space

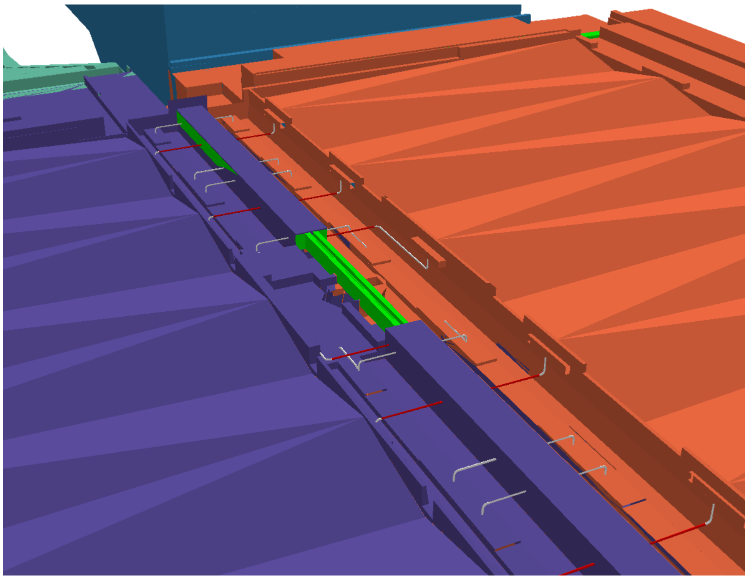


Figure 10.3: Example of missing space

10.1.3.5 Use of Clipped Spaces

An attempt was made to create a fourth set of spaces, referred to as the *BTH-CLIPPED* set, by clipping the *BTH-DILATED* set of spaces with the surrounding building element geometry. The outputted spaces formed when this operation was performed using the *FME Clipper* transformer were validated and found to be beyond repair (Section 8.3.2). Although these spaces are suitable for visualisation they cannot be used to identify the location of assets.

10.1.3.6 Closing of Openings

This thesis has demonstrated that watershed segmentation can, in principle, be used to create useful spaces for AM. In the case of the spaces found in the *Broadgate Ticket Hall*, the best parameters to use were 0.25 m and a Discretisation Factor (DF) of 0.5. It may well be that other parameters are suited to spaces and portals of other dimensions. Further research should now explore the most appropriate parameters to use in order to implement a reliable method for automatic production.

The workflows developed have also shown that watershed segmentation is capable of closing off openings and portals that connect spaces. It would appear that the method is most suited to closing off small openings in planar objects that have a distinctive "*pinch*", while openings at the end of corridors are more difficult to close off. Again further research is recommended into whether there are more appropriate methods for closing off corridors.

10.1.3.7 GPU-Assisted Methods

The adaptation of the watershed segmentation method to create watertight spaces was inspired by the work of Haumont, et al. (2003) who used the principles of the watershed transform to create Cell and Portal Graph (CPG) of architectural scenes. In their publication, Haumont, et al. (2003) adapted the classic morphological implementation of the watershed transform to use the capabilities of the GPU to create planar portals between spaces.

For the research in this thesis, it was decided to explore the potential of the watershed transform by using the original computational method. Further research should investigate whether the method advocated by Haumont, et al. (2003) can be implemented to create watertight spaces suitable for spatial queries (Further Work FW3).

10.1.3.8 Computational Resources

At every stage of the workflow starting with the extraction of CAD elements from *MicroStation*, to generating spaces using watershed segmentation, to performing spatial queries within a GIS platform there has to be a trade-off between the efficient use of computational resources and the quality of output. The methods developed in this thesis were mindful of the demand on computational resources, such as memory and computational time; however, the focus has been on proving the principle that the methods can be implemented.

Once further research has identified how the watershed transform can best be exploited to create practical spaces for asset management, attention should be switched on solving computational bottlenecks in the workflow (Further Work FW3). In its current design iteration, the most demanding process is the selection source markers with which to seed the watershed transform. The process uses brute force creation of graph networks to analyse the topology of the discretised distance field. This implementation is simple to understand, but it is demanding on computational resources; as such, it is first in the queue for a redesign.

10.1.3.9 Use of Virtual Space Boundaries

The focus of this research has been on the creation of practical watertight spaces from the raw CAD model files. Working with these raw model files has been a challenge, but the method implemented has shown that practical spaces can be created from rudimentary models. In a commercial environment, it would be impractical to rely on *watershed segmentation* to close off spaces. Designers should be aware of the requirement for 3D spaces (Recommendation R3) and more importance should be placed on model authors to design their models complete with virtual space boundaries (Section 3.9) to close space openings and portals. Further work is required to develop a practical implementation that is capable of handling virtual space boundaries (Further Work FW3).

Virtual space boundaries are important in defining spatial ownership and rights in 3D cadastres. If the segmentation algorithm is capable of handling virtual space boundaries, then the algorithm could be used to create 3D spaces for use in 3D cadastral surveys.

If models were designed with virtual space boundaries, then *boolean* difference methods could be relied upon to generate spaces from building element geometry. It must be remembered that solid geometric objects are fragile and can become corrupt. In the instance that the *boolean difference* method fails to create a watertight space, the watershed segmentation method can be relied upon as an alternative.

10.1.3.10 Buffering of Extruded Spaces

The results in Chapter 9 were found to favour the *BTH-DILATED* space set, but it must be remembered that the *BTH-DILATED* space set was specially created due to the *BTH-MANIFOLD* space set falling short of the space boundary. It could be argued that the *floor plan extrusion* method has been unfairly treated as no attempt has been made to create a buffered version of the *BTH-EXTRUDED* space set.

While it is reasonable to expect that a buffered space set would pick up more identifications than an unbuffered set, there is no reasoning to justify how to choose a suitable buffer distance. The dilation of the voxelised space is limited to one voxel because it is known that this region includes both building elements and portions of a space (except in the rare case that a voxel face is coincident with the boundary of the space). The danger of buffering (or dilating) spaces by an arbitrary amount is that there is a greater risk of obtaining false positives.

Just as the boolean difference method was used to clip the *BTH-DILATED* space set, the same method could be extended to a buffered version of the *BTH-EXTRUDED* set. If there are any openings in the bounding geometry, either intentional or through minuscule cracks, and if a buffer distance is chosen that is greater than the thickness of the bounding geometry, then the operation will back-fill any space behind the geometry. It should be noted that this back-fill can also occur when clipping the *BTH-DILATED* space set, but because the dilation is limited to a non-arbitrary distance, the risk is minimised.

10.1.3.11 Automation of space creation

The workflow developed in Chapter 8 for the generation of 3D spaces is still very much a manual operation. The workflow and algorithms developed so far are not yet ready to be used to create 3D volumetric spaces that correspond to the 2D floor plans. Further work is still needed to select suitable parameters such as grid orientation, voxel size and discretisation factor that can be used by the *watershed segmentation* algorithm to generate suitable spaces.

10.1.3.12 Addressing Supporting Question 3

With regard to answering the supporting question, the *floor plan extrusion* method and the *watershed segmentation* method can both be used to create watertight spaces for use in AM. However, *watershed segmentation* places a greater demand on computational resources, both in terms of generating the space, and in terms of performing the spatial join. For these reasons, although *floor plan extrusion* is the most practical method in the majority of use cases, the *watershed segmentation* and *boolean difference* methods provide alternative methods for creating those spaces which cannot be represented as an extruded polygon (Contribution C3). The 3D space plan should be made up of spaces created using a variety of methods advocated in this thesis (Recommendation R4).

10.1.4 Principal Research Question

The principal research question (RQ) is based on the hypothesis that the challenges that hinder integration are not entirely technical, but instead a derived from inherent conceptual differences. By stepping back from the tyranny of technical issues, the conceptual nature of the problem can be made clear.

The task of researching in a multi-disciplinary environment and understanding interoperability issues is made more complicated by the use of disparate nomenclature and colloquialisms, which prevent proper comparison from a neutral perspective. Therefore, it is necessary to split the systems up into their "*lowest common denominators"* and reinterpret the systems using a common Spatial Information Systems Framework. Once the levels of the systems have been aligned, it becomes much easier to identify different levels of heterogeneity. Furthermore, because the heterogeneities can be classed according to their level in the Framework, it is easier to see if conceptual differences are having an effect on the technical issues.

A comprehensive analysis of the Crossrail Technical Information Systems was conducted using the Spatial Information System Framework developed in Chapter 4. A methodical analysis identified that there are three relevant information systems in this investigation, rather just BIM and GIS. The challenge is not dichotomous as first expected but is instead trichotomous once AIMS is included.

As well as analysing the interoperability and integration between the Crossrail Technical Information Systems, this case study also delved deeper into one challenge namely the linking of CAD elements in *MicroStation* and assets in AIMS and this research has endeavoured to find a practical solution to this particular challenge. The challenge and the work undertaken to find a workflow has created opportunities for a better understanding of interoperability issues in terms of their technical and conceptual nature.

Firstly there are differences in semantics used to describe the CAD elements and the AIMS assets. These differences are compounded by how the CAD elements are aggregated together following a Work Breakdown Structure (WBS) suited for construction and how the AIMS assets are structured by an Asset Breakdown Structure (ABS) suited for AM. These technical differences are derived from differences in the conceptual requirements for the information that all stakeholders need to be aware of (Recommendation R5).

It was not possible to research the challenges of linking CAD elements and AIMS assets as the data sources were not fully available at the time of investigation (Section 2.1.2). Instead, it was decided to investigate the prerequisite steps (PRS 1, 2 & 3) required to prepare the data for linking, as sufficient data was available for this purpose. Once practical solutions for the prerequisite steps have been implemented, further research is required into the challenges of linking elements between spatial information systems (Further Work FW8).

The first pre-requisite step (PRS 1) involved performing an ETL operation to extract CAD elements from *MicroStation* and upload transformed elements into a GIS spatial database. Many technical challenges hindered this workflow as documented in Chapter 6 and Chapter 7 and discussed above in Section 10.1.2. These technical challenges exist because of the constraints of adopting a particular *Spatial Information Platform*, *MicroStation* in the case of the CAD elements and *Oracle Spatial* in the case of the GIS features.

Behind the scenes, there are conceptual differences that cause these challenges, or else a single platform would have been selected for the project. As such, tools in *MicroStation* are more developed for supporting the needs of the design community when compared to *Oracle Spatial*. Likewise, the *Spatial Information Model* formats in *MicroStation* are more developed for supporting engineering design tools, whereas the OGC-compliant *Spatial Information Model* formats are better suited for spatial analysis and information management. Adopting a single platform would constrain users and, therefore, may not be the best solution.

The second prerequisite step (PRS 2) involved creating explicit 3D spaces with which to perform spatial containment queries with the CAD elements extracted and transformed into GIS features. There is a requirement to create explicit 3D spaces because of conceptual differences between the design and operator communities. The CAD elements are explicitly prescriptive in that they communicate prescriptive intent for the constructors to build the object in the real world. The spaces on the hand are implicitly defined as the space that exists in between the CAD elements.

The design community do not have a requirement to define 3D spaces explicitly but are instead only required to deliver 2D floor plans. There is, therefore, a conceptual difference in the requirement to represent spaces. The future owner/operator has only identified a requirement for 2D spaces in the form of line drawings, whereas the spatial analyst preparing data for AM has a requirement for watertight 3D spaces. Because a conceptual difference exists, the technical challenge of creating explicit 3D watertight spaces arises.

The third prerequisite step (PRS 3) is to assign a named location to each CAD element using the spatial containment query. Here again, technical challenges arise in that the spatial query will identify all the spaces that a particular element passes through, whereas AIMS assets will be subjectively identified with reference to a single space. The technical challenge again arises from differences in conceptual requirements.

From these exercises conducted as part of this case study into the Crossrail Technical Information Systems, it can be seen that there are many instances where conceptual differences ultimately lead to technical difficulties further down the line. A better understanding of the conceptual and technical differences can be achieved by adopting a standard Spatial Information System Framework that can be applied to both BIM, GIS and any other relevant spatial information system (Contribution C5).

10.2 General Discussion

10.2.1 Asset Linking

The motivation for this thesis, as identified in Chapter 2, is to overcome the challenges that hinder integration, in particular the challenge of linking the information in the CAD model files with operational information systems that will satisfy the specifications in PAS 1192-3:2014 and ISO 19650-3:2020 that was developed through discussion with Crossrail (Contribution C4).

At the time of identifying the research gaps, the Infrastructure Manager (IM) had not specified any requirement for establishing a relationship between the asset information that will be used to populate their asset register and the AIM (Crossrail 2013) as stated in Section 2.1.2. During the course of this research, the IM announced that they had no intention to use the 3D CAD model files but would instead convert all 3D models into 2D plans and sections (MacDonald 2016). Although this policy goes against the principles of BIM, it is understandable as the IM must adopt a consistent strategy for all information within its control; it is impractical to have one strategy for existing information and another strategy for new information.

As well as complying with PAS 1192-3:2014, establishing relationships between information is important for avoiding information silos. If the IM had elected to go ahead with providing linked access to the CAD models in the AIM then it would provide the following range of benefits:

a. Asset managers working in AIMS would have with the ability to refer to all information related to the corresponding element in ProjectWise/*MicroStation*, which might include design notes and engineering calculations.

b. All assets in AIMS would be endowed with CAD geometry; as well as providing location this would include size, shape, orientation, adjacency and access. This geometry could then be used to visualise assets, either on-screen or using Virtual Reality (VR)/Augmented Reality (AR) technology. The ability to immerse a viewer in a VR/AR improves health and safety risk assessments and off-site work planning.

c. The geometry provided by CAD could also be used to answer topological queries such as "*what is the nearest control panel to the door to this room?"* or "*how many light fittings require scaffold access?"*

d. There would be a greater incentive to keep the structured graphical CAD model up-to-date during the operation phase of the built asset (OPEX). The *as-maintained* AIM could then be handed over to contractors at the start of a refurbishment or for disposal, in realisation of the vision for the PIM/AIM life cycle illustrated in Figure 3.12 that is promoted by PAS 1192-3:2014 (BSI 2014) and ISO 19650-3:2020 (ISO 2020).

According to information theory, information without context or a purpose has limited value. A link between information is also information in its own right, and therefore if the link does not have a purpose, it is difficult to specify the syntax, schema and semantics of the link. If the employer does not specify a requirement, it is important to clarify the meaning and limitations of any link between elements and assets; otherwise, the link has the potential to be misinterpreted.

One method of linking is for both CAD element and AIMS asset to be tightly-coupled by sharing the same Globally Unique Identifier (GUID). Adopting this method of linking would signify congruence between two objects, i.e. that the CAD element and the AIMS asset are representations of the same object in different systems. This may, however, be too restrictive and prevent any changes or corrections to the information model to meet any change in requirement or conceptual understanding. Adoption of a common GUID is also restricted to linking aggregated objects that are common across the two systems.

An alternative method of linking may be loose association. The link exists only to act as a signpost to point users in the right direction as they navigate between the two systems.

Adopting a loose relationship will inevitably lead to the information in the two systems getting out of step. As a result, cross-referencing will always require an element of human reasoning to understand and overcome any nuanced differences between the two systems.

Although it has not yet been possible to scope the full extent of the relationships between the two systems, it is known that there are differences between the breakdown structures and the schemes of classification. It will, in time, be necessary to formulate the best way to link objects that are aggregated in different ways, as illustrated in Figure 5.9, and establish how such a link might be interpreted.

It should be a long-term ambition to harmonise the WBS and ABS as closely as possible. As already stated, this leads to an inflexible schema that does not serve the needs of the communities that they are created for. For this reason, loose associations which can be used to infer relationships may be needed to prevent systems from becoming irrelevant to current needs.

The research of this thesis has been driven by an aspiration to link two representations of the same object as it is found in two heterogeneous asset information management systems. These multiple representations arise due to decisions that were made that are particular to the Crossrail project (Section 10.1.1.2). However, an extensive search has not found any similar reports of this issue within the archives of published literature. As such, the need to create asset links between heterogeneous systems within Crossrail may be an isolated requirement that may be avoided in the future with more advanced project management.

Although the situation at Crossrail may be an isolated case, the need to identify associations between objects in two information systems is a problem that extends beyond the fields of BIM or AM. If this is the case, then GIS has the potential to provide solutions to similar linking problems by using its spatial query functionality.

Whatever the future requirement for establishing relation links between CAD and AIMS may be, the workflow developed in Figure 2.3 remains a practical method for identifying the space to which an element belongs. In time, it is hoped that ETL operations will be as pain-free as possible. Although there may be less need to research workarounds to extract information, there will always be a need to implement effective quality control measures.

It is also hoped that there will be greater demand from employers for fully representative volumetric spaces. Including the need for 3D spaces in contract specifications would make it much easier to perform spatial queries on assets to determine the name of the space in which they are located. Nevertheless, the ability to create meaningful 3D spaces on-the-fly using techniques such as *watershed segmentation* would be very useful as an alternative method for calculating volume and acting as a quality control measure.

10.2.2 Linked Data

In Section 10.2.1, the importance of establishing links between systems has been emphasised, meeting not just the requirements of PAS 1192-3:2014 but also for creating opportunities to exploit information wherever it may be found. Where there is a defined requirement, APIs can be designed into the system to access and exchange data. The downside of this is incorporating dedicated APIs is expensive, inflexible and reliant on the application developer. Linked data has the potential to overcome these challenges.

Linked data, sometimes referred to as the *Semantic Web* is a powerful tool for accessing and analysing data. The founding principle of linked data is to restructure and store all data in Resource Description Framework (RDF) format consisting of subject-predicate-object triples in the form (“Element A”, “Has ID”, “ID_123”) or (“Element A”, “belongs to class”, “Pump”). The RDF data is then used in conjunction with developed ontologies (Farghaly, et al. 2019) to answer queries requested between syntactically and semantically heterogeneous systems.

If an application is capable of responding to RDF requests, then there is no limit to the flexibility of how the information can be used. Linked data is, therefore, a potential candidate to link information sources that are bound by a loose association as opposed to tight coupling, as explored in Section 10.2.1. A significant barrier to linked data is the requirement to analyse data source schema to build ontologies which must then be linked and verified. Incomplete or ambiguities in the ontologies will result in unexpected query results.

10.2.3 Other UK Infrastructure Megaprojects

In Section 3.8.1, a small number of UK infrastructure megaprojects were identified in the literature that involved the application of BIM to assist in AM. The first is the High Speed Two project which is much larger in scale to Crossrail, with Phase I alone being 225 km in length, of which at least 60 km will be underground. However, the project does not, however, include any underground stations, and therefore High Speed Two is unlikely to use named spaces to locate assets, as track-side assets are more likely to use linear referencing methods.

It would appear that the issue of semantic classes is also a continuing problem at HS2. Initial investigations by Floros, et al. (2020) have already identified concerns relating to the mis-mapping of information when interchanged via IFC. Similar problems were experienced in this research when exporting BIM elements from *Bentley MicroStation* via IFC. In fairness, this may be due to Crossrail not including IFC as a project requirement. Had this requirement been specified, work would have been carried out to map *MicroStation* parts/families with the classes found in IFC, thus avoiding extensive use of the generic *IfcBuildingElementProxy* class.

The other UK megaproject for which there is literature relating to BIM and AM is the Thames Tideway Tunnel. Whyte, et al. (2019) used the Thames Tideway project to explore the suitability of the project BIM to create a *Digital Twin* of the tunnel for the purpose of exploring system relationships and interdependencies. As well as other methods, Whyte, et al. (2019) investigated the use of *BIM Query* to explore relationships between objects. In this context, *BIM Query* is a tool for performing spatial queries between BIM objects avoiding the need to perform ETL operations into a GIS platform (Borrmann and Rank 2009). However, Whyte, et al. (2019) only had limited success in establishing interdependencies due to the significant loss of data exporting information from *Bentley AECOsim* via IFC, an experience not too different to the issues experienced in this research.

10.2.4 BIM for Asset Management

It has been a stated aim of the Building Information Disciplinary Field that the BIM should deliver savings throughout the lifecycle of a built asset (Chapter 1). It is generally agreed that BIM now provides a mature framework for project stakeholders to collaborate in an effective manner leading to greater efficiencies and savings throughout the Capital Expenditure (CAPEX) phase. However, a similar consensus is not shared with regard to BIM for AM/Facilities Management (FM). Although the PIM is capable of providing a wealth of information to be used for managing assets, there are many barriers to incorporating that information in AM/FM (López, et al. 2017; van Nederveen, et al. 2014).

Within the Crossrail project, it had been decided to create and populate an Asset Information Management System (AIMS) independently of the PIM (Taylor 2017). This was necessitated because the PIM stored in *Bentley MicroStation* was not suitable for storing the depth of information on each asset. Separating the AIMS from the PIM provided an additional benefit in that Crossrail could ensure that the required asset information would be ready in time for handover. The compromise that arises from creating an independent AIMS, is that information starts to diverge when it is no longer joined together. In making the information fit for the operations and maintenance phase (Ibrahim, et al. 2016), a gap has been created between the two information sources.

The standards for managing assets as specified in PAS 1192-3:2014 and ISO 19650-2:2020 appear to justify the expense of implementing a collaborative 3D model in the design and construction phase without considering the needs of the organisations that require the information (Ibrahim, et al. 2016; Farghaly, et al. 2018). The as-maintained information model used after handover must be fully interoperable with other systems already in existence, such as IoT and CAFM systems that will have been developed in isolation to the BIM (Lu, et al. 2019). It would appear that a lot more work is needed to align the requirements of the design and construction phase with the requirements of the operations and maintenance phase for the lifecycle savings to be realised as originally promised by the advocates of BIM

10.2.5 Other Requirements for ETL

So far this thesis has focussed on one particular challenge relating to the linking information in support of AM, but there are other practices that are hindered by interoperability issues that restrict the full potential of BIM. Spatial information can be considered from semantic and geometric perspectives. In the challenges addressed in this work, it has been necessary to consider the ETL of information from both perspectives although the geometric perspective has received the most emphasis as the spatial information has primarily been used for creating spaces and performing spatial queries.

Within the context of AM, the emphasis falls more predominantly on the requirement to handle semantic information. Once the spatial location of elements according to the space in which they are contained is ascertained, the research into the current challenge will progress to exploring semantic issues.

Staying focussed on the geometric perspective, the extraction of elements from CAD models is important for the visualisation of information outside of the proprietary platform. Visualisation has an important role in that it enables humans to orientate information within its surroundings. With proper contextual perspective, humans are able to analyse spatial relationships between features and other neighbouring features and make better decisions accordingly.

10.2.6 Other Applications for Spaces

There is good evidence to show that spaces are poorly represented in architectural settings. This poor representation is caused by the shortage of tools available to create well-represented spaces.

This observation can be illustrated using the literature on the use of BIM and GIS in the context of underground railway stations in Section 3.7.2. Marzouk and Abdelaty (2014) had a special interest in climatic data within a metro station but were not able to present their data visually in a meaningful way.

Without going into a deep study into the literature, it is fair to say that better visualisations improve human interpretation of data and information. However, there is often a significant cost in presenting indoor information in a 3D architectural setting, and there must be a cost-benefit analysis on the return of investment needed to improve decision-making. Voxelated spaces created using *watershed segmentation*, are a reliable source of representative geometry that can be used to visualise interior layouts; the same spaces, dilated and clipped with enclosing building elements, are even more so. Further Work is recommended on how best to implement the visualise spaces created using the methods advocated in this thesis (Further Work FW6).

In addition to visualisation, there is a demand for 3D representation of interior spaces in the domain of building energy modelling. At the very least, spaces can be used to gauge interior volume and surface areas. If the spaces are analysed in conjunction with the building elements that separate them, then thermal energy flow can be modelled, and Heating, Ventilation and Air Conditioning (HVAC) requirements can be determined.

Indoor spaces also play an important role in the field of indoor navigation. The spaces created using the methods advocated in this thesis can be analysed to understand how pedestrians will navigate through an architectural scene, thus enabling improved layouts and safer evacuation routes. Indoor navigation tends to be interested in spaces as a 2D floor plan, but 3D representations can be used by tools to understand how floors fit together with stairways and escalators.

10.2.7 CityGML 3.0

A new version of the CityGML conceptual model and schema will shortly be published, incorporating some fundamental changes to the model. This revision has included the adoption of a universal approach to representing all features in CityGML in terms of an *AbstractSpaceObject* and an *AbstractSpaceBoundaryObject*. For internal spaces, this means that the conceptual model is now more closely aligned with IFC, which will undoubtedly eliminate some of the issues preventing the exchange of information between IFC and City Geographical Mark-up Language (CityGML).

These changes to the CityGML schema are unlikely to have an immediate effect on the issues researched in this thesis relating to the linking of CAD elements and AIMS assets. However, with the changes to the Level of Detail (LoD) model, there may be an increased understanding across the GIS and BIM communities that an internal space can be represented in 4 different LoD (Löwner, et al. 2016), namely:

- LoD 0 - 2D floor plan or central point
- LoD 1 - Extruded 2.5D floor plan
- LoD 2 - Extruded 2.5D floor plan incorporating 3D ceiling
- LoD 3 - 3D volumetric space (watertight solid)

With greater awareness and expectation of what is possible, there may then come greater demand for a higher priority for the delivery of fully represented 3D spaces.

Another development being rolled out as part of the new revision is that CityGML 3.0 will have the capability to represent spaces as point clouds. It seems that the CityGML Standards Working Group (SWG) is open to adopting new forms of geometric representation. 3D spatial enumeration is already commonly used to represent the geometry of internal spaces for navigation (Koopman 2016; Gorte, et al. 2019). There may, therefore, be a scope to include the capability in the CityGML model to represent internal spaces using voxel representation.

10.2.8 Applications of Spatial Information System Framework

One great benefit of having a unified Spatial Information System Framework is that it provides both the GIS and BIM communities with a unified perspective of understanding the systems that they use. By having a shared language, it might be possible to see whether there are advances and developments in each other's domains that might be of benefit to their own particular domain.

As an example, a GIS practitioner might be able to ask whether the principles of Information Management specified in ISO 19650 suite could be applied to a similar collaborative project not involving construction, i.e. land management. Likewise, there are many volumes of literature on the subject of Spatial Data Infrastructure (SDI). The Common Data Environment (CDE) that is central to BIM is akin to a corporate SDI, and thus literature found within the geospatial community may also have relevance to the BIM community.

10.2.9 Decomposition of the Process Level

In the development of the Framework in Chapter 4, the existing frameworks used for general information systems supported the existence of a single *Spatial Information Processes* level. It was decided to split the *Processes* level into a *Spatial Information Tools* level and an *Spatial Information Activities* level based on a demarcation of automated processes and manual processes. However, the splitting of this level at this junction is more nuanced than simply human versus computer.

Many processes in an information system involve an interaction between human and computer at what is referred to as the Human Computer Interface (HCI). Information is presented, normally in visual form, to a human user who makes decisions within the *Processes* level. These decisions are either implemented with the Human Activity System (HAS) or are fed back to the technical system.

Many tools are semi-automatic in that they are mostly algorithmic but still reliant on human intelligence to recognise patterns and guide the algorithm. As the adoption of artificial intelligence advances, an increasing number of these semi-automatic tools will be replaced by machine learning tools. The boundary between the *Tools* and *Activities* level can only become more blurred.

10.2.10 Complexity of Real World Models

The research in this thesis has hugely benefited from having access to documentation, situational examples and real-world data from within the Crossrail project together with advice and comments from experienced professionals within the organisation. The use of real-world information has, however, had its advantages and disadvantages.

The collection of Technical Information Systems within the Crossrail project has been valuable testing ground with which to prove the application of the Spatial Information Systems Framework. The challenges of real-world systems with idiosyncrasies and conflicting requirements provides credibility to the trial use of the Framework.

Performing ETL operations on CAD model files using commercial software as-installed on the company systems has been helpful for understanding the real-world problems faced by industry. This is highlighted by the failure of the installed software to export IFC models that could be read by *FME Workbench* which required the use of an updated version provided under an academic licence. It demonstrates the disruptive nature of minor quirks that are a prevalent challenge constantly requiring time and effort to fix.

Using model files from a live construction project has been of great benefit. Simple test cases are satisfactory to try out proposed workflows and algorithms, but only real-world models can provide unexpected problems that showcase the limitations of the proposed solutions.

10.2.11 Validation of Watertight Solids

Working with watertight solids has been a significant challenge when performing operations developed for this research. At various stages within the information workflow, operations are performed on watertight spaces that are represented as valid solid objects within a variety of software applications, including *MicroStation*, *AutoCAD*, *FME Workbench*, *OpenCascade*, *OpenSCAD*, *Meshlab*, *Trimesh* and the *ArcGIS 3D Analyst* toolbox.

At each stage, the operations require the objects to be validated as solids using certain criteria and tolerances. These criteria are mostly unspecified, and it is only possible to customise the tolerances in *FME Workbench* and *OpenCascade.* This has led to inconsistent validation across the workflow, with a solid being validated in one application only to be rejected in another. It is felt that a better understanding of the tolerances used in the validation process is required to ensure that solid objects are created, and if necessary repaired, with consistency.

10.2.12 Longevity and Interoperability

One important observation that can be made from this research is the importance of ensuring interoperability with information created at various times in history. The ability to read historic data formats is a ubiquitous challenge that plagues the information industry. In order for BIM data to be used throughout the life of a built asset, it must be archived in a format that will always be openly readable (Recommendation R6). Closed proprietary formats that can only be read by software maintained by a single organisation are unlikely to be suitable candidates for this purpose.

The work of this thesis has demonstrated that exporting to open formats is prone to the corruption of information due to bugs in the export tools. If an open format is used to archive information, but is otherwise unused with preference given to the closed proprietary format, then the corrupted information will not be identified as such until after it is too late.

One of the lessons learned in this research is that technical information must be interpreted using the same conceptual and dynamic context with which it was created. If information is archived, it must be accompanied by as much supporting documentation as possible to explain how the information is to be interpreted if it is to remain fit for purpose across time.

10.2.13 Federated Modelling Environments

The standard industry practice for accessing *Building Information Models* is through the use of file-based federated models that use a file management system such as *Bentley ProjectWise* to check files in and out of the server. This practice is in line with *Stage 2* of the *Information Management Stages of Maturity* in ISO 19650-1:2018 (ISO 2018) (UK *BIM Level 2*). As technology advances, the sector is expected to adopt object-based server information models, similar to those used in the geospatial community, in order to progress to *Stage 3* of this scale.

The advantage of using information models held in an Object Relational Database Management System (ORDBMS) is that they are much easier to access. Spatial indexing would facilitate searching for and querying information, increasing the spatial awareness within the model. Although the storage of information may transition from file-based to server-based, it is important to maintain model federation, i.e. information in the model must be grouped together in stand-alone collections according to location and discipline. Federation enables engineers to own their designs and certify that their design is compliant with safety regulations and the employer's requirements.

10.2.14 Naming of Spaces

Within the Crossrail project, the design contractor was required to deliver floor plans in 2D CAD (Section 5.2.3.2) which are manually drawn up in consultation with London Underground (LU) and London Fire Brigade (LFB). These spaces are named and identified in accordance with LU and LFB protocols. The names and Identifiers (IDs) of these spaces become the authoritative space names, which are then used by Crossrail and the future owner of the underground stations. As each space is also given a space ID, thus allowing the name of a space to change while still keeping a persistent space ID for referential integrity. Problems will, however, arise whenever a space is sub-divided, in which case it is important to use new space IDs rather than keeping the original space ID for one of the new spaces.

As already discussed in Section 10.1.3.9, the output generated using *watershed segmentation* are over-segmented, and manual processing is required to merge the over-segmented output to form spaces that correspond to the original floor plans. The spaces created in Section 9.2.1 are still a "*work-in-progress"* and do not yet sufficiently correspond to the real-world spaces. The names of these spaces have been created solely to assess the suitability of the spatial join operations in Section 9.3.2. It is proposed that further work be carried out to improve the creation of 3D volumetric spaces that correspond to the authoritative 2D floor plans, specifically through the use of virtual boundaries within the model (Further Work FW3).

Having addressed the principal and supporting research question and discussed other observations of interest this chapter will make way for the conclusion and recommendations for further work in Chapter 11.

11 Conclusion and Further Work

The research presented in this thesis was born out of a concern that the conceptual and technical challenges hindering the integration of Building Information Modelling (BIM) and Geographic Information Systems (GIS) impaired the overall effectiveness of BIM and its ability to drive down the cost of constructing and operating built assets and reduce their effect on the environment. The research has been supported through a case study that involved a broad examination of the Crossrail Technical Information Systems and a close investigation into the challenge of linking of information between the Computer Aided Design (CAD) system and Asset Information Management System (AIMS).

Two research gaps were identified: the first was the lack of a common framework with which to describe the range of socio-technical levels of both BIM and GIS; the second was a scarcity of literature on methods for linking spatial features across information systems. From these gaps, a principal research question and three supporting research questions were set out.

In response to the first supporting research question (SQ 1), a Spatial Information Systems Framework was developed in Chapter 4, and its effectiveness was tested by applying interoperability diagrams to the various integration approaches found in literature. The Framework was then applied to the Crossrail Technical Information Systems so as to analyse heterogeneity at each level and identify potential issues affecting interoperability.

Concerning the second question (SQ 2), a workflow was written for performing an automated Extract-Transform-Load (ETL) operation to move CAD elements to a GIS spatial database (PRS 1). The workflow developed was successfully implemented and used to supply features for creating spaces (PRS 2) and with which to perform a spatial join (PRS 3). As well as having a practical purpose, the development of the workflow contributed many examples of interoperability to the case study.

With regard to the third supporting question (SQ 3), a program was written for implementing a workflow to create interior spaces from Boundary Representation (B-Rep) features using a *watershed transform* and converting these voxel-based spatial enumerations back to B-Rep (PRS 2). These spaces were then used for performing a spatial join with the CAD elements earlier uploaded to the spatial database (PRS 3).

The results of using these three-dimensional (3D) spaces to identify the location of assets are described in Chapter 9, and it was concluded that although using spaces generated from floor plans is the fastest method, the use of spaces created using *watershed segmentation* is more accurate with regards to identifying the correct location of assets.

Development of the Spatial Information Systems Framework and the observations and challenges experienced in finding a practical workflow have contributed to gaining a better understanding of the technical and conceptual challenges that hinder integration between BIM and GIS. From this work, the following recommendations are made for the Architectural, Engineering and Construction (AEC) and Asset/Facilities Management (AM/FM) communities to consider for implementation in future infrastructure projects.

11.1 Recommendations for Future Infrastructure Project Stakeholders

R1 - Spatial Information System Framework – It is recommended that the stakeholder responsible for implementing a Common Data Environment (CDE) adopt the Spatial Information System Framework developed in Chapter 4 to gain a better understanding of how various systems fit together (Section 10.1.1). Use of the Framework will help to understand how conflicting conceptual requirements at the higher-level can have long term effects at the technical level. It should also be recognised that although information may be technically interoperable, it may not yet be fit for purpose outside of the context it was created for.

R2 - Employers' Information Requirements for ETL – Employers should identify the need to perform ETL operations on information held within the information systems to be used for a project or operation (Section 10.1.2.3) and that design contractors should produce all their CAD-based BIM models to be fit for ETL operations. It is recommended that CAD-based BIM authors create a register of elements that make up the model so as to explicitly exclude *construction* elements and other ancillary geometric features.

The Employer should establish Quality Assurance procedures for testing ETL operations so as to confirm that ETL output matches the elements in the register and that features correspond to their original geometries within the specified tolerance.

All software users are encouraged to report instances where ETL operations do not behave as expected to the software provider. There must be no restriction on users from sharing confidential information with the software provider, and software providers should be a party to confidentiality terms.

R3 - Employers' Information Requirements for Spaces – Employers should specify their need for 3D space plan within Employer's Information Requirements (EIRs) to meet their requirements for asset management, visualisation, building energy modelling and indoor navigation (Section 10.1.2.3). EIRs should also specify the need for polygon-based two-dimensional (2D) floor plans that are compatible with the 3D space plan.

The specifications of the 3D space plan may differ depending on the activity that it needs to support. A particular 3D space plan may be constructed using spaces created using a variety of methods, e.g. *watershed segmentation* and *floor plan extrusion*.

Employers should specify their need for CAD model files to include 3D virtual space boundaries to partition spaces. Provisional virtual space boundaries should be constructed during the development stage to assist the *watershed transform* with the segmentation of the model.

R4 - Watershed Segmentation – It is recommended that tools be developed within Spatial Information Systems that use the watershed transform method to segment spaces within a model (Section 10.1.3.12). These tools should ideally be available within the CAD platform on which the model is created. The *watershed segmentation* method provides a robust method for creating spaces when other methods, i.e. *boolean difference*, have the potential to fail. The method can also be used on model files in development to provide a tool for scoping the layout of spaces and for measuring space volumes.

R5 - Employers' Information Requirements for Asset Management – In many ways Crossrail has been a champion in the early involvement of asset managers within an infrastructure project (Taylor 2017). Despite leading the way in this field, there are still lessons to be learned.

Although this research did not investigate issues relating to the Work Breakdown Structure (WBS) and Asset Breakdown Structure (ABS) schemes of classification, this challenge is the motivation for the work contained within. Employers are reminded to specify in their EIRS that the WBS scheme of classification used for design and construction of buildings and infrastructure needs to be compatible with the ABS scheme of classification used for managing assets for the purposes of AM and FM (Section 10.1.4).

The ultimate use of the Asset Information Model (AIM) by the Employer should be considered by the design and construction contractors throughout the delivery phase. This will provide Asset Managers with the ability to link their current asset information back to information created during the delivery phase, thus enabling better decision-making.

R6 - Longevity of Asset Information in Archived Formats – Employers must recognise the shelf life of information stored in closed proprietary formats (Section 10.2.12). All information should be extracted from proprietary formats and archived in an open standard (and on a medium) that is capable of being accessed for the life of the built asset.

11.2 Summary of Research Contributions

In addition to the recommendations to future infrastructure projects, the work from this thesis makes the following contributions to various fields of knowledge.

C1 - Spatial Information System Framework – The work of Chapter 4 has contributed a thorough review of the existing frameworks used to structure BIM and GIS alongside various frameworks used in general information systems theory. This review was carried out alongside a review of the various approaches to achieve BIM and GIS integration and how these approaches have been grouped in literature.

Chapter 4 proposes the use of a new seven-level Spatial Information Systems Framework that can be used to describe both BIM-based and GIS-based systems. This Framework complements existing frameworks such as the framework proposed by Succar (2009) and the Levels of Conceptual Interoperability Model (LCIM) advocated by Tolk, et al. (2007). It can be used to explain the different approaches attempted towards the integration of BIM and GIS and as such, contributes to the work of Kang and Hong (2015), Amirebrahimi, et al. (2015) and Beck, et al. (2020). As well as applying it to eleven approaches to BIM/GIS integration found in literature, the Framework was also validated by applying it to the Crossrail Technical Information Systems. The Framework has the potential to assist with categorising future work on BIM/GIS interoperability and integration and hence identify gaps in the research effort (Section 10.1.1).

The review into information systems and the development of the Framework contributes to a general understanding that interoperability is not just achieved at the technical level but also at the conceptual level. Understanding the relationship between conceptual and technical interoperability challenges needs to be recognised by the work being undertaken to standardise BIM/GIS integration methods.

C2 - Extraction, Transformation and Loading – Chapter 6 and Chapter 7 provide a practical case study into ETL operations working with *Bentley MicroStation* and reports on the challenges faced extracting spatial information from proprietary models. It solved practical issues and proposes a technical workflow for automated ETL of CAD elements to GIS features (Section 10.1.2.12). An initial report on these practical solutions has been published by the author in a paper (Boyes, et al. 2017) that was written within the scope of this thesis. The export of IFC geometry from *MicroStation* is documented problem (Whyte, et al. 2019; Floros, et al. 2020) and the methods described in this thesis can be practically applied to other projects to overcome technology lock-in.

With regards to its academic contribution, an important takeaway from these chapters is the importance of implementing quality assurance procedures to monitor the quality of ETL output within the field of BIM/GIS integration (Section 10.1.2.2).

C3 - Spaces – The work in Chapter 8 and Chapter 9 advocates the importance of creating explicit watertight spaces instead of relying on undefined voids that implicitly exist within enclosing elements. It supports the extensive work being undertaken predominantly by Delft University of Technology and the University of New South Wales under the direction of Professor Sisi Zlatanova to understand the concepts of indoor space and their research into indoor navigation (Diakité and Zlatanova 2016; Xiong, et al. 2016; Zlatanova, et al. 2020). This work contributes not only in the field of Asset Management (AM) but also to any field listed in Section 3.9.1, with particular reference to performing spatial queries, building energy modelling and indoor navigation.

This thesis contributes an investigation into the various methods that can be used to create watertight spaces, in particular, *floor plan extrusion*, *boolean difference* and *watershed segmentation*. It has taken the *watershed transform*, an algorithm that is predominantly used in medical imaging, and applied it to the creation of watertight spaces for use in the realm of AM. In doing so, this thesis recognises the work of Haumont, et al. (2003) who identified the application of the *watershed transform* with regards to architectural scenes in the context of Cell and Portal Graph (CPG) in the field of computing visualisation and gaming.

A full-cycle workflow has been implemented that takes building elements extracted from CAD model files and outputs closed watertight spaces as planar B-Rep suitable for use with 3D GIS tools (Section 10.1.3.12). This workflow has been implemented as a *Python* program taking advantage of the *numpy*, *PythonOCC*, *Trimesh*, *Scikit-Image*, *iGraph* and *NetworkX* packages available for *Python*. Furthermore, a method has been developed for converting a volume represented using voxel space enumeration into a manifold watertight planar B-Rep which overcomes issues relating to the non-manifold nature of touching edges and corners (Section 9.2.2).

C4 - Linking – This thesis documents a proposed method that was developed in discussion with the Crossrail GIS team to analyse relationships between CAD-based elements and non-spatial AIMS assets by performing a spatial join on CAD-based elements with a 3D representation of the spaces that contains those elements (Section 1.2.1). A report on this method has been published by the author in a paper (Boyes, et al. 2017) written within the scope of this thesis. This thesis identified that three prerequisite steps were required before performing this analysis, namely the ETL operation, the space generation and the performing the spatial join. This thesis has confirmed that these steps are hindered by interoperability issues and has proposed solutions to overcome these challenges.

C5 - Asset Information Management – This study confirms that conceptual differences have a controlling effect on the technical challenges that hinder in the handover of spatial information from the delivery phase to the operational phase of a built asset (Section 10.1.4). Two such conceptual challenges have been identified in this thesis: firstly, the use of different hierarchical schemes of classification to support different enterprise activities, and used in the different phases; and secondly, the use of different concepts for representing space again to support different enterprise activities.

The work here contributes to the body of research being undertaken into how to achieve a better management of information in preparation of handing over to the operational phase and thus achieve the cost efficiencies and carbon reduction in the management of assets that are advocated by the use of BIM. The recommendations on how to achieve better interoperability of GIS and BIM for AM are an important contribution to the GeoBIM project (Arroyo Ohori, et al. 2018; Ellul, et al. 2018).

11.3 Further Work

The work in this thesis has just started to explore how a better understanding of the conceptual and technical issues that hinder integration will lead to improved management of information. More work is required in relation to the Spatial Information Systems Framework, the ETL operations and the creation of 3D spaces for their general application as well as contributing to solve the challenge of linking assets between systems.

FW1 - Framework – At the moment the Spatial Information Systems Framework has been applied to two test cases: firstly, the various approaches in literature aimed at achieving BIM/GIS integration; and secondly, the Technical Information Systems in use within the Crossrail Technical Information Department.

The suitability of the Spatial Information Systems Framework should be confirmed by applying it to a wider range of scenarios. In conjunction with further case studies, it is recommended that industry and academic experts in the field of BIM and GIS be approached to answer questionnaires and engage in interviews to gather feedback (Section 10.1.1). This feedback would enable a comprehensive SWOT analysis to be conducted.

FW2 - ETL – A successful workflow has been implemented that is capable of batch processing the ETL operation. The quality of the ETL operation for each model file has been performed manually. Further work should be carried out to develop a workflow that is capable of comparing the geometric properties of each element across the three export formats to determine whether there are any corrupted geometries (Section 10.1.2.4).

FW3 - Watershed Segmentation – In Chapter 8, the use of *watershed segmentation* has been demonstrated as a concept for creating watertight spaces. This has been carried out on two space complexes, namely the *Broadgate Ticket Hall* and the *Mile End Shaft*. The use of *watershed segmentation* to create spaces in many more case studies should be explored to understand its capabilities and limitations and gain confidence in its wider use. From this, a better understanding of how to choose voxel size and discretisation factor can be gained.

In the case of the *Broadgate Ticket Hall*, the orientation of the voxels was set manually. Aligning the voxel grid with the predominant direction reduces the number of faces that must be created in the B-Rep conversion. Further work is required to select and implement a suitable algorithm for automatically deciding the most suitable axes (Section 10.1.3.1).

It was found in Section 8.3 that the *watershed segmentation* method suffers from over-segmentation, an inability to determine interior and exterior spaces (Section 10.1.3.1) and an inability to identify portals at the end of corridors and further work is required to investigate these issues.

In Section 10.1.3 the possibility of using provisional virtual space boundaries was raised as a potential option for creating better spaces, and it is recommended in Section 11.1 that CAD-based BIM authors include these in their models. Further research should also be carried out on the use of virtual boundaries and floor plans to assist the *watershed segmentation* algorithm.

The *Python* program was developed in a practical manner with the aim of obtaining results from executable code. There are no doubt many areas for improvement with regards to memory use and computational time. Further work is therefore required to improve the efficiency of the code.

It was decided in this research to implement the classic mathematical morphological watershed algorithm. The opportunity may exist to explore whether the GPU-assisted algorithm proposed by Haumont, et al. (2003) can be used to create watertight spaces.

FW4 - Boolean Difference – The *boolean difference* tools tested in Chapter 8 have not created valid watertight solids, and consequently, it has not been possible to test how these clipped spaces might perform when used to perform spatial join in Chapter 9. Further investigation is required in finding a suitable software application or code library capable of performing reliable boolean operations on planar B-Rep (Section 8.3.3.9).

FW5 - Mesh Conversion – In Section 9.2.2, a concept for converting voxelated spatial enumeration of a single space into planar B-Rep mesh has been demonstrated. Further work is required to improve the reliability of the algorithm on real-world geometry (Section 10.1.3.2) and investigate extending its capabilities, including the ability to expand the voxel mesh out to the nearest enclosing geometry object.

FW6 - Spatial Joining – In Section 9.3.2, the spatial joins were performed using the *Inside3D* tool. The output produced using this tool identifies all the spaces that a contain a particular feature, even if a feature is only fractionally inside one of the spaces (Section 9.3.3). The task of analysing the relationship between CAD elements and AIMS assets could be simplified if each feature was only associated with a single named space. Research should, therefore, be conducted on the most suitable method for identifying the space in which a feature is most predominantly contained.

The spatial joins performed in Section 9.3.2 took Mechanical, Electrical and Plumbing (MEP) assets contained within the spaces as their input and not on the building elements (i.e. walls, floors) that enclose the spaces. These elements are not normally contained within the space but only touch the surface of the spaces. Research should, therefore, be conducted into the most suitable method to identify the spaces that touch the surface of enclosing building elements.

FW7 Visualisation – Within the remit of this thesis, research has been conducted to find the most suitable method for creating spaces for the purpose of performing spatial queries. However, spaces can be used in many more applications, such as indoor navigation and building energy modelling. Even within the remit of AM, it may be that spaces provide valuable functionality to visualise assets in the context of their surroundings. Because *watershed segmentation* provides a robust method for creating spaces in any situation, further research should be carried out how spaces can be used to improve visualisation of assets (Section 10.2.6).

FW8 - Asset linking – This work has been motivated in part by a requirement to investigate the relationship between CAD-based elements and AIMS assets and establish links between them. Now that a suitable workflow has been developed to link elements and assets by the name of the space that they are located in, research should now continue into analysing the relationship between WBS and ABS and the schemes of classification used in the different systems (Section 10.1.4). From this, it is maybe possible to develop a practical workflow for linking assets in support of asset management.

11.4 Concluding Remarks

The world is reliant on information systems to support every aspect of society, moreover, they have become the foundation on which civilisation is now built. Although the technical aspects of information systems are becoming increasingly open, the conceptual requirements that drive the technical requirements are less well understood, maybe even lost and forgotten. It is hoped that the theoretical framework and practical workflows developed here will provide the opportunity to gain a better understanding of the complexity that exists between spatial information systems such as BIM and GIS. Only once the question "*why do interoperability issues arise?*" has been answered can solutions towards "*how can integration be achieved?*" be found.

References

Abdelhady, I.A I. and Jones, J., 2014. Mapping of Building Information Modeling Business Process Model and Related Issues in both the Schematic Design and Design Development Phases. In: *Proceedings of the 14th International Conference on Construction Applications of Virtual Reality and Islamic Architecture, 16-18 November 2014, Sharjah, UAE*, pp. 120-130. Available from: https://research.tees.ac.uk/en/publications/proceedings-of-the-14th-international-conference-on-construction- [Accessed on: 25 August 2015].

Agdas, D. and Ellis, R.D., 2010. IT in transportation construction: opportunities and barriers to implementation . In: *Proceedings of International Conference on Computing in Civil and Building Engineering (ICCCBE), 30 June - 2 July, Nottingham, UK*, pp. 223-228. Available from: http://www.engineering.nottingham.ac.uk/icccbe/proceedings/pdf/pf112.pdf [Accessed on: 27 January 2020].

Agugiaro, G., Benner, J., Cipriano, P., and Nouvel, R., 2018. The Energy Application Domain Extension for CityGML: enhancing interoperability for urban energy simulations, *Open Geospatial Data, Software and Standards* 3(1), pp. 139-30. Available from: doi:10.1186/s40965-018-0042-y

Ahmed, S., 2018. Barriers to Implementation of Building Information Modeling (BIM) to the Construction Industry: A Review, *Journal of Civil Engineering and Construction* 7(2), pp. 107-113. Available from: doi:10.32732/jcec.2018.7.2.107

Amadi-Echendu, J.E., Willett, R., Brown, K., Hope, T., Lee, J., Mathew, J., Vyas, N., and Yang, B.-S., 2010. What Is Engineering Asset Management?. In: *Definitions, Concepts and Scope* *of Engineering Asset Management*, London, pp. 3-16. Available from: doi:10.1007/978-1-84996-178-3_1.

Amirebrahimi, S., Rajabifard, A., Mendis, P., and Ngo, T., 2015. A Data Model for Integrating GIS and BIM for Assessment and 3D Visualisation of Flood Damage to Building. In: *CEUR Workshop Proceedings Research@Locate'15, 10-12 March 2015, Brisbane, Australia*, pp. 78-89. Available from: http://ceur-ws.org/Vol-1323/paper27.pdf [Accessed on: 18 September 2015].

Amirebrahimi, S., Rajabifard, A., Mendis, P., and Ngo, T., 2016. A BIM-GIS integration method in support of the assessment and 3D visualisation of flood damage to a building, *Journal of Spatial Science*, pp. 1-34. Available from: doi:10.1080/14498596.2016.1189365

Arroyo Ohori, K., Diakité, A.A., Ledoux, H., Stoter, J., and Krijnen, T., 2018. *The GeoBIM Project*. Delft University of Technology, Netherlands. Available from: https://3d.bk.tudelft.nl/ken/files/18_geobim.pdf [Accessed on: 14 February 2020].

Atazadeh, B., Kalantari, M., Rajabifard, A., Ho, S., and Champion, T., 2016. Extending a BIM-based data model to support 3D digital management of complex ownership spaces, *International Journal of Geographical Information Science*, pp. 1-24. Available from: doi:10.1080/13658816.2016.1207775

Autodesk, 2014. Revit 2014 Documentation - Rooms Spanning Floors or Levels [Online]. Available from: http://help.autodesk.com/view/RVT/2014/ENU/?guid=GUID-50C282D8-2856-4B5C-88E8-5A2A92774F0B [Accessed on: 10 August 2020].

Bartelme, N., 2012. Geographic Information Systems. In: *Springer Handbook of Geographic Information*, Spring, Heidelberg, pp. 59-71. Available from: doi:10.1007/978-3-540-72680-7_6.

Batty, M., 2009. Urban Modeling. In: *International Encyclopedia of Human Geography*, Elsevier Science, Amsterdam, pp. 51-58. Available from: doi:10.1016/B978-008044910-4.01092-0.

Bayar, M.S., Aziz, Z., Tezel, A., Arayici, Y., and Biscaya, S., 2016. Optimizing handover of as-built data using BIM for highways. In: *1st International (UK) BIM Academic Forum*

Conference, 13-15 September 2016, Glasgow, Scotland. Available from: http://usir.salford.ac.uk/id/eprint/40138 [Accessed on: 25 April 2017].

Bazjanac, V., 2010. Space boundary requirements for modeling of building geometry for energy and other performance simulation. In: *Proceedings of CIB W79 27th International Conference, 16-19 November 2010, Cairo, Egypt*. Available from: http://itc.scix.net/paper/w78-2010-127 [Accessed on: 11 December 2014].

Becerik-Gerber, B., Jazizadeh, F., Li, N., and Calis, G., 2012. Application Areas and Data Requirements for BIM-Enabled Facilities Management, *Journal of Construction Engineering and Management* 138(3), pp. 431-442. Available from: doi:10.1061/(ASCE)CO.1943-7862.0000433

Beck, A.R., Cohn, A.G., Sanderson, M., Ramage, S., Tagg, C., Fu, G., Bennett, B., and Stell, J.G., 2008. UK utility data integration: overcoming schematic heterogeneity. In: Liu, L., Li, X., Liu, K., Zhang, X., & Chen, A., eds., *SPIE Proceedings of Geoinformatics 2008 and Joint Conference on GIS and Built Environment: Geo-Simulation and Virtual GIS Environments, 3 November 2008, Guangzhou, China*. Available from: doi:10.1117/12.812600.

Beck, F., Borrmann, A., and Kolbe, T.H., 2020. The Need for a Differentiation Between Heterogeneous Information Integration Approaches in the Field of “BIM-GIS Integration”: a Literature Review, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* VI-4/W1-2020, pp. 21-28. Available from: doi:10.5194/isprs-annals-VI-4-W1-2020-21-2020

Beetz, J., van Leeuwen, J., and de Vries, B., 2006. Towards a Topological Reasoning Service for IFC-Based Building Information Models in a Semantic Web Context. In: *Proceedings of Joint International Conference on Computing and Decision Making in Civil and Building Engineering, 14-16 June 2006, Montréal, Canada*, pp. 3426-3435. Available from: http://itc.scix.net/data/works/att/w78-2006-tf527.pdf [Accessed on: 25 February 2016].

Beetz, J., Coebergh, W., Botter, R., Zlatanova, S., and de Laat, R., 2014. Interoperable data models for infrastructural artefacts – a novel IFC extension method using RDF vocabularies exemplified with quay wall structures for harbors . In: Mahdavi, A., Martens, B., & Scherer, R., eds., *eWork and eBusiness in Architecture, Engineering and Construction : Proceedings of the 10th European Conference on Product and Process Modelling, 17-19 September 2014, Vienna, Austria*, pp. 135-140. Available from: doi:10.1201/b17396.

Béjar, R., Latre, M.Á., Nogueras-Iso, J., Muro-Medrano, P.R., and Zarazaga-Soria, F.J., 2012. An RM-ODP enterprise view for spatial data infrastructures, *Computer Standards & Interfaces* 34(2), pp. 263-272. Available from: doi:10.1016/j.csi.2011.10.001

Bell, A., 2015. Missing elements after export of 3D design model to STL file format [Online]. Available from: https://communities.bentley.com/products/microstation/w/microstation__wiki/19591/missing-elements-after-export-of-3d-design-model-to-stl-file-format [Accessed on: 28 September 2021].

Bentley Systems, 2012. *Asset Painter User Guide*. Bentley Systems Inc., Exton, PA. Available from: https://communities.bentley.com/products/assetwise/assetwise_platform/m/mediagallery/256221 [Accessed on: 29 June 2017].

Bentley Systems, 2020. The Part and Family System [Online]. Available from: https://docs.bentley.com/LiveContent/web/AECOsim\%20Building\%20Designer\%20Help-v5/en/GUID-4D680A9F-A04B-E2AB-FBF2-509FAFB8341D.html [Accessed on: 23 June 2020].

Beucher, S. and Meyer, F., 1993. The Morphological Approach to Segmentation: The Watershed Transformation. In: Dougherty, E.R., ed., *Mathematical Morphology in Image Processing*, pp. 433-481.

Beynon-Davies, P., 2009. *Business Information Systems*, 1st ed., Palgrave Macmillan, Basingstoke.

Beynon-Davies, P., 2010. The enactment of significance: a unified conception of information, systems and technology, *European Journal of Information Systems* 19(4), pp. 389-408. Available from: doi:10.1057/ejis.2010.34

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., and Çöltekin, A., 2015. Applications of 3D City Models: State of the Art Review , *ISPRS International Journal of Geo-Information* 4(4), pp. 2842-2889. Available from: doi:10.3390/ijgi4042842

BIM Task Group, 2014. BIM Task Group FAQs [Online]. Available from: http://www.bimtaskgroup.org/bim-faqs/ [Accessed on: 25 July 2014].

BIMplus, 2021. BIM+ Homepage [Online]. Available from: https://www.bimplus.co.uk/ [Accessed on: 28 August 2021].

Bishr, Y., 1998. Overcoming the semantic and other barriers to GIS interoperability, *International Journal of Geographical Information Science* 12(4), pp. 299-314. Available from: doi:10.1080/136588198241806

Björk, B.-C., 1992. A conceptual model of spaces, space boundaries and enclosing structures, *Automation in Construction* 1(3), pp. 193-214. Available from: doi:10.1016/0926-5805(92)90013-A

Boguslawski, P., Fadli, F., Boguslawski, P., Mahdjoubi, L., Zverovich, V., and Barki, H., 2015. BIM-GIS modelling in support of emergency response applications. In: *Building Information Modelling (BIM) in Design, Construction and Operations*, WIT Press, Southampton, pp. 381-392. Available from: doi:10.2495/BIM150321.

Borrmann, A. and Rank, E., 2009. Specification and implementation of directional operators in a 3D spatial query language for building information models, *Advanced Engineering Informatics* 23(1), pp. 32-44. Available from: doi:10.1016/j.aei.2008.06.005

Borrmann, A., Kolbe, T.H., Donaubauer, A., Steuer, H., Jubierre, J.R., and Flurl, M., 2015. Multi-Scale Geometric-Semantic Modeling of Shield Tunnels for GIS and BIM Applications, *Computer-Aided Civil and Infrastructure Engineering* 30(4), pp. 263-281. Available from: doi:10.1111/mice.12090

Boyes, G., Thomson, C., and Ellul, C., 2015. Integrating BIM and GIS: Exploring the use of IFC space objects and boundaries. In: *Proceedings of the 23rd GIS Research UK Conference, 15-17 April 2015, Leeds, UK*. Available from:

http://leeds.gisruk.org/abstracts/GISRUK2015_submission_106.pdf [Accessed on: 20 April 2015].

Boyes, G.A., Ellul, C., and Irwin, D., 2017. Exploring Bim for Operational Integrated Asset Management - a Preliminary Study Utilising Real-World Infrastructure Data , *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-4/W5, pp. 49-56. Available from: doi:10.5194/isprs-annals-IV-4-W5-49-2017

Boyes, G., 2014. *Integration of Building Information Modelling and Geographic Information Systems*. Department of Civil, Environmental and Geomatic Engineering, University College London. M.Sc. dissertation.

Boyes, G., 2017. BIM / GIS Integration - Getting from MicroStation to ArcGIS. In: *Presentations of FME World Tour, 23 March 2017, London, UK*. Available from: www.safe.com/presentations/bim-gis-integration-getting-microstation-arcgis [Accessed on: 4 September 2020].

BSI, 2007. BS 1192:2007. *Collaborative production of architectural, engineering and construction information - Code of practice*. British Standards Institution, London.

BSI, 2012. BS 8210:2012. *Guide to facilities maintenance management*. British Standards Institution, London.

BSI, 2013. PAS 1192-2:2013. *Specification for information management for the capital/delivery phase of construction projects using building information modelling*. British Standards Institution, London.

BSI, 2014. PAS 1192-3:2014. *Specification for information management for the operational phase of assets using building information modelling*. British Standards Institution, London.

BSI, 2014a. BS 1192-4:2014. *Collaborative productionof information Part 4:Fulfilling employer's information exchange requirements using COBie - Code of practice*. British Standards Institution, London.

BSI, 2015. PAS 1192-5:2015. *A specification for security-minded building information modelling, digital built environments and smart asset management*. British Standards Institution, London.

BSI, 2018. PAS 1192-6:2018. *Specification for collaborative sharing and use of structured Health and Safety information using BIM*. British Standards Institution, London.

BSI, 2019. PD 19650-0:2019. *Transition guidance to BS EN ISO 19650*. British Standards Institution, London.

buildingSMART, 2007. Industry Foundation Classes IFC2x3 TC 1 [Online]. Available from: https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/TC1/HTML/ [Accessed on: 31 July 2014].

buildingSMART, 2020. IFC Documentation - IfcAlignment [Online]. Available from: http://docs.buildingsmartalliance.org/IFC4x2_Bridge/schema/ifcproductextension/lexical/ifcalignment.htm [Accessed on: 31 January 2020].

buildingSMART, 2020a. Technical Documentation - IFC Formats [Online]. Available from: https://technical.buildingsmart.org/standards/ifc/ifc-formats/ [Accessed on: 29 June 2020].

Butterfield, A., Ngondi, G.E., and Kerr, A., eds., 2019. Data Model. In: *Oxford Dictionary of Computing Science*, Oxford University Press, Oxford. Available from: https://www.oxfordreference.com/view/10.1093/acref/9780199688975.001.0001/acref-9780199688975-e-1238 [Accessed on: 10 October 2019].

Campanaro, D.M., Landeschi, G., Dell'Unto, N., and Leander Touati, A.-M., 2016. 3D GIS for cultural heritage restoration: A 'white box' workflow, *Journal of Cultural Heritage* 18, pp. 321-332. Available from: doi:10.1016/j.culher.2015.09.006

Carbonari, G., Ashworth, S., and Stravoravdis, S., 2015. How Facility Management can use Building Information Modelling (BIM) to improve the decision making process. Personal Communication.

Casey, M. and Vankadara, S., 2009. Semantics in CAD/GIS Integration. In: *CAD and GIS Integration*, pp. 143-170. Available from: doi:10.1201/9781420068061-c7.

Cavka, H.B., Staub-French, S., and Poirier, E.A., 2017. Developing owner information requirements for BIM-enabled project delivery and asset management, *Automation in Construction* 83, pp. 169-183. Available from: doi:10.1016/j.autcon.2017.08.006

CEN, 2012. PD CEN/TR 15449-1:2012. *Geographic information - Spatial data infrastructures Part 1: Reference model*. British Standards Institution, London.

CEN, 2012a. PD CEN/TR 15449-2:2012. *Geographic information — Spatial data infrastructures Part 2: Best practices*. British Standards Institution, London.

CEN, 2012b. PD CEN/TR 15449-3:2012. *Geographic information — Spatial Data Infrastructure — Data centric view*. British Standards Institution, London.

CEN, 2013. PD CEN/TR 15449-4:2013. *Geographic information — Spatial Data Infrastructure — Service centric view*. British Standards Institution, London.

Cerovsek, T., 2011. A review and outlook for a 'Building Information Model' (BIM): A multi-standpoint framework for technological development, *Advanced Engineering Informatics* 25(2), pp. 224-244. Available from: doi:10.1016/j.aei.2010.06.003

CGAL, 2020. The Computational Geometry Algorithms Library [Code Library]. Available from: https://www.cgal.org [Accessed on: 14 February 2020].

Chai, L., Zhang, L., Yu, G., Qu, Y., and Du, K., 2008. Workflow-based system design for the online inspection of spatial data. In: Liu, L., Li, X., Liu, K., & Zhang, X., eds., *SPIE Proceedings of Geoinformatics 2008 and Joint Conference on GIS and Built Environment: Advanced Spatial Data Models and Analyses, 10 November 2008, Guangzhou, China*. Available from: doi:10.1117/12.813107.

Chrisman, N.R., 1999. What Does 'GIS' Mean?, *Transactions in GIS* 3(2), pp. 175-186. Available from: doi:10.1111/1467-9671.00014

Ciribini, A.L C., Mastrolembo Ventura, S., and Paneroni, M., 2016. Implementation of an interoperable process to optimise design and construction phases of a residential building: A

BIM Pilot Project, *Automation in Construction* 71, pp. 62-73. Available from: doi:10.1016/j.autcon.2016.03.005

Cohen-Or, D. and Kaufman, A., 1995. Fundamentals of Surface Voxelization, *Graphical Models and Image Processing* 57(6), pp. 453-461. Available from: doi:10.1006/gmip.1995.1039

Construction Specifications Institute, 2020. The Omniclass Classification System [Online]. Available from: https://www.csiresources.org/standards/omniclass [Accessed on: 6 June 2020].

Costin, A., Adibfar, A., Hu, H., and Chen, S.S., 2018. Building Information Modeling (BIM) for transportation infrastructure – Literature review, applications, challenges, and recommendations, *Automation in Construction* 94, pp. 257-281. Available from: doi:10.1016/j.autcon.2018.07.001

Cowen, D., 1988. GIS versus CAD versus DBMS: What are the differences?, *Photogrammetric Engineering and Remote Sensing* 54(11), pp. 1511-1555.

CPIC, 2014. Uniclass 1.4 Legacy Release [Online]. Available from: http://www.cpic.org.uk/uniclass/1-4/ [Accessed on: 1 August 2016].

Crompton, G. and Jupe, R., 2008. Network Rail – forward or backward? Not-for-profit in British transport, *Business History* 49(6), pp. 908-928. Available from: doi:10.1080/00076790701710423

Crossrail, 2010. Crossrail awards major tunnelling contracts worth £1.25bn [Online]. Available from: http://www.crossrail.co.uk/news/articles/crossrail-awards-major-tunnelling-contracts-worth-125bn [Accessed on: 22 January 2016].

Crossrail, 2013. *Crossrail BIM Principles*. Crossrail Learning Legacy, CR-XRL-Z3-RGN-CR001-50005. Available from: https://learninglegacy.crossrail.co.uk/learning-legacy-themes/information-management-and-technology/information-management/ [Accessed on: 7 July 2020].

Crossrail, 2014. TfL announces MTR to run Crossrail services [Online]. Available from: http://www.crossrail.co.uk/news/articles/tfl-announces-mtr-to-run-crossrail-services [Accessed on: 22 January 2016].

Crossrail, 2015. Crossrail in numbers [Online]. Available from: http://www.crossrail.co.uk/news/crossrail-in-numbers [Accessed on: 22 January 2016].

Crossrail, 2016. Crossrail – from its early beginnings [Online]. Available from: http://www.crossrail.co.uk/route/crossrail-from-its-early-beginnings [Accessed on: 22 January 2016].

Crossrail, 2016a. *CAD Representative Handbook*. Crossrail Learning Legacy, CR-XRL-Z3-GUI-CR001-50480. Available from: https://learninglegacy.crossrail.co.uk/wp-content/uploads/2018/07/12A-005_01_CAD-Handbook-LL_Redacted.pdf [Accessed on: 9 July 2020].

Crossrail, 2017. *Asset Identification Standard*. Crossrail Learning Legacy, CRL1-XRL-O6-STD-CR001-00031. Available from: https://learninglegacy.crossrail.co.uk/wp-content/uploads/2018/05/12C-004-01_Asset-Identification-Standard-9.0_Redacted.pdf [Accessed on: 7 July 2020].

Cython Community, 2020. Cython [Python Package]. Available from: https://cython.org [Accessed on: 6 October 2020].

Da Luz Patacas, J.M., Dawood, N., and Kassem, M., 2014. Evaluation of IFC and Cobie as data sources for asset register creation and service life planning. In: Dawood, N. & Alkass, S., eds., *Proceedings of the 14th International Conference on Construction Applications of Virtual Reality, 16-18 November 2014, Sharjah, UAE*. Available from: https://research.tees.ac.uk/en/publications/evaluation-of-ifc-and-cobie-as-data-sources-for-asset-register-cr-3 [Accessed on: 25 August 2015].

Dado, E., Beheshti, R., and van de Ruitenbeek, M., 2010. Product Modelling in the Building and Construction Industry. In: *Handbook of research on building information modeling and* *construction informatics : concepts and technologies*, Information Science Reference, IGI Global, Hershey, PA, pp. 104-137. Available from: doi:10.4018/978-1-60566-928-1.ch005.

Daum, S. and Borrmann, A., 2014. Processing of Topological BIM Queries using Boundary Representation Based Methods, *Advanced Engineering Informatics* 28(4), pp. 272-286. Available from: doi:10.1016/j.aei.2014.06.001

Davila Delgado, J.M., Butler, L.J., Gibbons, N., Brilakis, I., Elshafie, M.Z E B., and Middleton, C., 2017. Management of structural monitoring data of bridges using BIM, *Proceedings of the Institution of Civil Engineers - Bridge Engineering* 170(3), pp. 204-218. Available from: doi:10.1680/jbren.16.00013

Dawson-Haggerty, M., 2020. TriMesh [Python Package]. Available from: https://trimsh.org/ [Accessed on: 6 October 2020].

Demian, P., Ruikar, K., Sahu, T., and Morris, A., 2016. Three-Dimensional Information Retrieval (3DIR) - Exploiting 3D Geometry and Model Topology in Information Retrieval from BIM Environments, *International Journal of 3-D Information Modeling* 5(1), pp. 67-78. Available from: doi:10.4018/IJ3DIM.2016010105

Dempsey, J., 2017. *Managing Assets in the Context of Asset Managment*. ISO/TC 251 WG4 Report. International Standards Organisation. Available from: https://committee.iso.org/files/live/sites/tc251/files/guidance/ISO\%20TC251\%20WG4\%20MACAM\%20May\%202017\%20EN2.pdf [Accessed on: 17 December 2017].

Detlor, B., 2010. Information management, *International Journal of Information Management* 30(2), pp. 103-108. Available from: doi:10.1016/j.ijinfomgt.2009.12.001

Diakité, A.A. and Zlatanova, S., 2016. Extraction of the 3D Free Space from Building Models for Indoor Navigation, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-2/W1, pp. 241-248. Available from: doi:10.5194/isprs-annals-IV-2-W1-241-2016

Diakité, A.A., Damiand, G., and Maercke, D.V., 2014. Topological Reconstruction of Complex 3D Buildings and Automatic Extraction of Levels of Detail. In: *Proceedings of Eurographics*

Workshop on Urban Data Modelling and Visualization, 6 April 2014, Strasbourg, France, pp. 25-30. Available from: doi:10.2312/udmv.20141074.025-030.

Döllner, J. and Hagedorn, B., 2007. Integrating urban GIS, CAD, and BIM data by service-based virtual 3D city models. In: *Urban and Regional Data Management: UDMS 2007 Annual*, Taylor and Francis, Netherlands, pp. 157-160.

Donkers, S., Ledoux, H., Zhao, J., and Stoter, J., 2015. Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings, *Transactions in GIS* 20(4), pp. 547-569. Available from: doi:10.1111/tgis.12162

East, E.W., 2016. The Construction-Operations Building Information Exchange Specification [Online]. Available from: https://www.wbdg.org/resources/construction-operations-building-information-exchange-cobie [Accessed on: 4 August 2020].

Eastman, C., Teicholz, P., Sacks, R., and Liston, K., 2011. *The BIM Handbook - A Guide to Building Information Modeling for Owners, Managers*, 2nd ed., John Wiley and Sons, Hoboken, NJ.

Eastman, C.M., 1975. The Use of Computers Instead of Drawings in Building Design, *American Institute of Architects Journal* 63(3), pp. 46-50. Available from: https://www.researchgate.net/publication/234643558_The_Use_of_Computers_Instead_of_Drawings_in_Building_Design [Accessed on: 13 October 2020].

Ekholm, A. and Fridqvist, S., 2000. A concept of space for building classification, product modelling, and design, *Automation in Construction* 9(3), pp. 315-328. Available from: doi:10.1016/S0926-5805(99)00013-8

El Mekawy, M.S. A., Östman, A., and Shahzad, K., 2008. Geospatial Interoperability for IFC and CityGML: Challenges of Existing Building Information Databases. In: *Proceedings of Innovations 08th, IEEE Conference, 16-18 December 2008, Dubai*. Available from: https://www.researchgate.net/publication/265743286_Geospatial_Interoperability_for_IFC_and_CityGML_Challenges_of_Existing_Building_Information_Databases [Accessed on: 1 July 2019].

El-Mekawy, M., Östman, A., and Hijazi, I., 2012. A Unified Building Model for 3D Urban GIS, *ISPRS International Journal of Geo-Information* 1(3), pp. 120-145. Available from: doi:10.3390/ijgi1020120

El-Mekawy, M., Östman, A., and Hijazi, I., 2012a. An Evaluation of IFC-CityGML Unidirectional Conversion, *International Journal of Advanced Computer Science and Applications* 3(5). Available from: doi:10.14569/IJACSA.2012.030525

Ellul, C., Boyes, G., Thomson, C., and Backes, D., 2017. Towards Integrating BIM and GIS—An End-to-End Example from Point Cloud to Analysis. In: Abdul Rahman, A., ed., *Advances in 3D Geoinformation*, Springer, Cham, pp. 495-512. Available from: doi:10.1007/978-3-319-25691-7_28.

Ellul, C., Stoter, J., Harrie, L., Shariat, M., Behan, A., and Pla, M., 2018. Investigating the State of Play of GeoBIM Across Europe, *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-4/W10, pp. 19-26. Available from: doi:10.5194/isprs-archives-XLII-4-W10-19-2018

Erie County Water Authority, National Center for Geographic Information and Analysis, and Group, G.R., 1996. *Local Government GIS Development Guides*. National Center for Geographic Information and Analysis, State University of New York at Buffalo. Available from: http://www.ncgia.buffalo.edu/ncgia/sara/ [Accessed on: 13 October 2020].

ESRI, 2015. Crossrail Journeys Towards 3D GIS, *Think GIS* Issue #38. Esri UK Ltd, Aylesbury. Available from: https://www.esriuk.com/~/media/esri-uk/ThinkGIS/EsriUK_ThinkGIS_Issue38a.pdf [Accessed on: 31 January 2020].

ESRI, 2019. ArcMap Documentation - Geoprocessing considerations for shapefile output [Online]. Available from: https://desktop.arcgis.com/en/arcmap/latest/manage-data/shapefiles/geoprocessing-considerations-for-shapefile-output.htm [Accessed on: 29 June 2020].

ESRI, 2020. ArcMap Documentation - Types of geodatabases [Online]. Available from: https://desktop.arcgis.com/en/arcmap/latest/manage-data/geodatabases/types-of-geodatabases.htm [Accessed on: 6 June 2020].

ESRI, 2020a. ArcGIS Documentation - What is Workflow Manager? [Online]. Available from: https://desktop.arcgis.com/en/arcmap/latest/extensions/workflow-manager/what-is-workflow-manager-.htm [Accessed on: 4 April 2020].

Farghaly, K., Abanda, F.H., Vidalakis, C., and Wood, G., 2018. Taxonomy for BIM and Asset Management Semantic Interoperability, *Journal of Management in Engineering* 34(4). Available from: doi:10.1061/(ASCE)ME.1943-5479.0000610

Farghaly, K., Abanda, F.H., Vidalakis, C., and Wood, G., 2019. BIM-linked data integration for asset management, *Built Environment Project and Asset Management* 9(4), pp. 489-502. Available from: doi:10.1108/BEPAM-11-2018-0136

Fei, Y., Wang, B., and Chen, J., 2012. Point-Tessellated Voxelisation. In: *Proceedings of Graphics Interface, 28-30 May 2012, Toronto, Canada*, pp. 9-18. Available from: https://dl.acm.org/doi/10.5555/2305276.2305280 [Accessed on: 24 January 2019].

Fisher, D., 2006. *An Emergent Perspective on Interoperation in Systems of Systems*. Software Engineering Institute, Carnegie Mellon University, CMU/SEI-2006-TR-003. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA.

Floros, G.S., Boyes, G., Owens, D., and Ellul, C., 2019. Developing IFC for Infrastructure: a Case Study of Three Highway Entities, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-4/W8, pp. 59-66. Available from: doi:10.5194/isprs-annals-IV-4-W8-59-2019

Floros, G.S., Ruff, P., and Ellul, C., 2020. Impact of Information Management during Design and Construction on Downstream BIM-GIS Interoperability for Rail Infrastructure, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* VI-4/W1-2020, pp. 61-68. Available from: doi:10.5194/isprs-annals-VI-4-W1-2020-61-2020

Fosu, R., Suprabhas, K., Rathore, Z., and Cory, C., 2015. Integration of Building Information Modeling (BIM) and Geographic Information Systems (GIS) – a literature review and future needs. In: *Proceedings of the 32nd International Council for Research and Innovation in Building and Construction CIB W78 Conference, 27-29 October 2015, Eindhoven*, pp. 196-204. Available from: http://itc.scix.net/data/works/att/w78-2015-paper-020.pdf [Accessed on: 1 September 2016].

FreeCAD, 2020. FreeCAD [Software]. Available from: https://www.freecadweb.org [Accessed on: 8 June 2020].

General Services Administration, 2007. *Spatial Program Validation*. General Services Administration, Washigton DC. Available from: https://www.gsa.gov/cdnstatic/BIM_Guide_Series_02_v096.pdf [Accessed on: 13 October 2020].

Geospatial World, 2016. How viable is BIM as a career? [Online]. Available from: https://www.geospatialworld.net/article/how-viable-is-bim-as-a-career/ [Accessed on: 30 August 2021].

Gielingh, W., 1988. General AEC reference model (GARM) an aid for the integration of application specific product definition models. In: Christiansson, P. & Karlsson, H., eds., *CIB Proceedings of Conceptual Modelling of Buildings. CIB W74+W78 Seminar, October, 1988. Lund, Sweeden*. Available from: https://www.researchgate.net/publication/273126714_General_AEC_reference_model_GARM_an_aid_for_the_integration_of_application_specific_product_definition_models [Accessed on: 12 January 2016].

Gilbert, T., Rönsdorf, C., Plume, J., Simmons, S., Nisbet, N., Gruler, H.-C., Kolbe, T.H., van Berlo, L., and Mercer, A., 2020. *Built environment data standards and their integration: an analysis of IFC, CityGML and LandInfra*. buildingSMART, OGC 19-091r1. Available from: https://portal.ogc.org/files/?artifact_id=92634 [Accessed on: 12 December 2020].

Gilbert, T., James, P., Smith, L., Barr, S., and Morley, J., 2021. Topological integration of BIM and geospatial water utility networks across the building envelope, *Computers, Environment and Urban Systems* 86(31). Available from: doi:10.1016/j.compenvurbsys.2020.101570

Gillies, S., Butler, H., Daly, M., Doyle, A., and Schaub, T., 2016. RFC 7946. *The GeoJSON Format*. The Internet Engineering Task Force. Available from: http://geojson.org [Accessed on: 18 June 2016].

Gillies, S., 2020. Rtree [Python Package]. Available from: https://github.com/Toblerity/rtree [Accessed on: 6 October 2020].

Gillies, S., 2020a. Shapely [Python Package]. Available from: https://github.com/Toblerity/Shapely [Accessed on: 6 October 2020].

GIS Certification Institute, 2020. The GIS Certification Institute [Online]. Available from: https://www.gisci.org [Accessed on: 20 March 2020].

Goodchild, M.F., 1992. Geographical Information Science, *International Journal of Geographical Information Systems* 6(1), pp. 31-45. Available from: doi:10.1080/02693799208901893

Gorte, B., Zlatanova, S., and Fadli, F., 2019. Navigation in Indoor Voxel Models, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-2/W5, pp. 279-283. Available from: doi:10.5194/isprs-annals-IV-2-W5-279-2019

Gröger, G., Kolbe, T.H., Nagel, C., and Häfele, K.-h., 2012. OGC 12-019. *OGC City Geography Markup Language (CityGML) 2.0 Encoding Standard*. Open Geospatial Consortium. Available from: https://www.ogc.org/standards/citygml [Accessed on: 13 October 2020].

Grus, L., Crompvoets, J., and Bregt, A.K., 2010. Spatial data infrastructures as complex adaptive systems, *International Journal of Geographical Information Science* 24(3), pp. 439-463. Available from: doi:10.1080/13658810802687319

Hagedorn, B. and Döllner, J., 2007. High-level web service for 3D building information visualization and analysis. In: *ACM Proceedings of the 15th International Symposium on Advances in Geographic Information Systems, November 2007, Seattle, Washington*. Available from: doi:10.1145/1341012.1341023.

Halfawy, M.R., 2004. *The Interoperability of Geographic Information Systems for Municipal Asset Management Applications*. Part 2. National Research Council, Ottawa, Canada.

Halttula, H., Haapasalo, H., and Herva, M., 2016. Barriers to Achieving the Benefits of BIM, *International Journal of 3-D Information Modeling* 4(4), pp. 16-33. Available from: doi:10.4018/IJ3DIM.2015100102

Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., et al., 2020. Array programming with NumPy, *Nature* 585(7825), pp. 357-362. Available from: doi:10.1038/s41586-020-2649-2

Haumont, D., Debeir, O., and Sillion, F., 2003. Volumetric cell-and-portal generation, *Computer Graphics Forum* 22(3), pp. 303-312. Available from: doi:10.1111/1467-8659.00677

Herring, J.R., 1999. The OpenGIS Data Model, *Photogrammetric Engineering and Remote Sensing* 65(5), pp. 585-588. Available from: https://www.asprs.org/wp-content/uploads/pers/1999journal/may/1999_may_585-588.pdf [Accessed on: 13 October 2020].

Herring, J.R., 2011. OGC 06-103. *OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture*. Open Geospatial Consortium. Available from: https://www.ogc.org/standards/sfa [Accessed on: 8 January 2016].

Hijazi, I.H. and Donaubauer, A., 2017. Integration of Building and Urban Information Modeling – Opportunities and Integration Approaches. In: Kolbe, T.H., Bill, R., & Donaubauer, A., eds., *Geoinformationssysteme 2017*, Herbert Wichmann Verleg, Karlsruhe. Available from: https://mediatum.ub.tum.de/doc/1350517 [Accessed on: 3 January 2021].

Hijazi, I., Ehlers, M., Zlatanova, S., Becker, T., and van Berlo, L., 2010. Initial Investigations for Modeling Interior Utilities Within 3D Geo Context: Transforming IFC-Interior Utility to CityGML/UtilityNetworkADE. In: *Advances in 3D Geoinformation Systems*, Springer, Heidelberg, pp. 95-113. Available from: doi:10.1007/978-3-642-12670-3_6.

Hjelmager, J., Moellering, H., Cooper, A., Delgado, T., Rajabifard, A., Rapant, P., Danko, D., Huet, M., Laurent, D., Aalders, H., et al., 2008. An initial formal model for spatial data infrastructures, *International Journal of Geographical Information Science* 22(11-12), pp. 1295-1309. Available from: doi:10.1080/13658810801909623

Hjelseth, E. and Thiis, T.K., 2008. Use of BIM and GIS to enable climatic adaptations of buildings. In: Zarli, A. & Scherer, R., eds., *eWork and eBusiness in Architecture, Engineering and Construction: ECPPM 2008*, CRC Press, Boca Raton, pp. 409-417.

HM Government and Greater London Council, 1974. *London Rail Study*. Department of Environment, London.

HM Government, 2011. *Government Construction Strategy*. Cabinet Office, London. Available from: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/61152/Government-Construction-Strategy_0.pdf [Accessed on: 11 December 2014].

HM Government, 2015. *Digital Built Britain – Level 3 Strategy*. Department of Business, Industry and Skills, London. Available from: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/410096/bis-15-155-digital-built-britain-level-3-strategy.pdf [Accessed on: 8 June 2015].

Ibrahim, K.F., Abanda, F.H., Vidalakis, C., and Woods, G., 2016. BIM for FM: Input versus Output data. In: *Proceedings of the 33rd CIB W78 Conference, 31 Oct – 2 Nov 2016, Brisbane, Australia*. Available from: doi:10.24384/g2k3-sm81.

igraph Core Team, 2020. python-igraph [Python Package]. Available from: https://igraph.org/python/ [Accessed on: 6 October 2020].

Iliffe, J.C., Arthur, J.V., and Preston, C., 2013. The Snake Projection: A Customised Grid for Rail Projects, *Survey Review* 39(304), pp. 90-99. Available from: doi:10.1179/003962607X165041

Institution of Civil Engineers, 2020. Career profile: BIM Manager [Online]. Available from: https://www.ice.org.uk/careers-and-training/return-to-a-career-in-civil-engineering/which-civil-engineering-role-is-right-for-you/career-profile-bim-manager [Accessed on: 9 March 2020].

Irwin, D. and Tamash, N., 2016. *Building a Spatial Data Infrastructure for Crossrail*. Crossrail Learning Legacy. Available from: https://learninglegacy.crossrail.co.uk/documents/building-a-spatial-infrastructure-for-crossrail/ [Accessed on: 9 October 2019].

Irwin, D., 2016. Conversation relating to progress of asset identification. Personal Communication.

Isikdag, U. and Zlatanova, S., 2009. Towards Defining a Framework for Automatic Generation of Buildings in CityGML Using Building Information Models. In: *3D Geo-Information Sciences*, Berlin, Heidelberg, pp. 79-96. Available from: doi:10.1007/978-3-540-87395-2_6.

Isikdag, U., Underwood, J., and Aouad, G., 2008. An investigation into the applicability of building information models in geospatial environment in support of site selection and fire response management processes, *Advanced Engineering Informatics* 22(4), pp. 504-519. Available from: doi:10.1016/j.aei.2008.06.001

ISO, 1998. BS ISO/IEC 10746-1:1998. *Information technology — Open Distributed Processing — Reference model: Overview*. British Standards Institution, London.

ISO, 2005. BS EN ISO 19101:2005. *Geographic information — Reference model*. British Standards Institution, London.

ISO, 2005a. BS EN ISO 19107:2005. *Geographic Information - Spatial Schema*. British Standards Institution, London.

ISO, 2010. BS ISO 29481 - 1:2010. *BIM - Information Delivery Manual*. British Standards Institution, London.

ISO, 2014. BS ISO 55000:2014. *Asset management: Overview, principles and terminology*. British Standards Institution, London.

ISO, 2014a. BS ISO 6707-1:2014. *Buildings and civil engineering works — Vocabulary Part 1: General terms*. British Standards Institution, London.

ISO, 2015. BS ISO 12006-2:2015. *Building construction — Organization of information about construction works Part 2: Framework for classification*. British Standards Institution, London.

ISO, 2017. ISO 41011:2017. *Facility management - Vocabulary*. British Standards Institution, London.

ISO, 2018. BS EN ISO 19650-1:2018. *Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) - Information management using building information modelling* . British Standards Institution, London.

ISO, 2018a. BS EN ISO 19650-2:2018. *Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) - Information management using building information modelling* . British Standards Institution, London.

ISO, 2020. BS EN ISO 19650-3:2020. *Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) — Information management using building information modelling — Part 3: Operational phase of assets* . British Standards Institution, London.

ISO, 2020a. BS EN ISO 19650-5:2020. *Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) — Information management using building information modelling Part 5: Security-minded approach to information management* . British Standards Institution, London.

ISO, 2020b. BS EN ISO 19136-1:2020. *Geographic information – Geography Markup Language - Part 1: Fundamentals*. British Standards Institution, London.

Jernigan, F., 2007. *BIG BIM, little bim*, 2nd ed., 4Site Press, Salisbury, MD.

Johnston, R.J., 2005. Geography and GIS. In: *Geographical Information Systems: Principles, Techniques, Management and Applications, 2nd Edition, Abridged*, Wiley, Ney York, pp. 27-35. Available from: https://www.geos.ed.ac.uk/~gisteac/gis_book_abridged/files/ch03.pdf [Accessed on: 25 October 2020].

Jones, N.L., McCrone, C.J., Walter, B.J., Pratt, K.B., and Greenberg, D.P., 2013. Automated translation and thermal zoning of digital building models for energy analysis. In: *Proceedings* *of the 13th International Conference of the International Building Performance Simulation Association, 26-28 August 2013, Chambéry, France*. Available from: https://www.osti.gov/servlets/purl/1089753 [Accessed on: 26 October 2020].

Jones, M.C., 2015. Composability. In: *Modeling and Simulation Support for System of Systems Engineering Applications*, John Wiley & Sons, Hoboken, NJ, pp. 43-73. Available from: doi:10.1002/9781118501757.ch3.

Jusuf, S.K., Mousseau, B., Godfroid, G., and Soh Jin Hui, V., 2017. Integrated modeling of CityGML and IFC for city/neighborhood development for urban microclimates analysis, *Energy Procedia* 122, pp. 145-150. Available from: doi:10.1016/j.egypro.2017.07.329

Kang, T.W. and Hong, C.H., 2015. A study on software architecture for effective BIM/GIS-based facility management data integration, *Automation in Construction* 54, pp. 25-38. Available from: doi:10.1016/j.autcon.2015.03.019

Kang, T., 2018. Development of a Conceptual Mapping Standard to Link Building and Geospatial Information, *ISPRS International Journal of Geo-Information* 7(5). Available from: doi:10.3390/ijgi7050162

Kara, A., Çağdas, V., Işıkdağ, Ü., and Turan, B.O., 2018. Towards harmonizing property measurement standards, *Journal of Spatial Information Science* 17. Available from: doi:10.5311/JOSIS.2018.17.412

Kassem, M., Kelly, G., Dawood, N., Serginson, M., and Lockley, S., 2015. BIM in facilities management applications: a case study of a large university complex, *Built Environment Project and Asset Management* 5(3), pp. 261-277. Available from: doi:10.1108/BEPAM-02-2014-0011

Kavrakov, D., 2015. Comparative Analysis of Asset vs. Facility Management – Contemporary Roles and Functions [Online]. Available from: http://www.tcapital.bg/presentations/AMvsFM_Eng.pdf [Accessed on: 1 September 2016].

Kenley, R., Harfield, T., and Behnam, A., 2016. BIM Interoperability Limitations: Australian and Malaysian Rail Projects. In: Kamaruzzaman, S.N B., Ali, A.S B., Azmi, N.F B., & Chua, S.J L., eds., *Proceedings of the 4th International Building Control Conference,7-8 March 2016, Kuala Lumpur, Malaysia*. Available from: doi:10.1051/matecconf/20166600102.

Kennerley, B., 2013. What is BIM? [Online]. Available from: http://www.wsp-pb.com/en/Who-we-are/In-the-media/News/2013/What-is-BIM/ [Accessed on: 17 March 2016].

Kim, S., Kim, J., Jung, J., and Heo, J., 2015. Development of a 3D Underground Cadastral System with Indoor Mapping for As-Built BIM: The Case Study of Gangnam Subway Station in Korea, *Sensors* 15(12), pp. 30870-30893. Available from: doi:10.3390/s151229833

Kitsakis, D. and Dimopoulou, E., 2017. Addressing Public Law Restrictions within a 3D Cadastral Context, *ISPRS International Journal of Geo-Information* 6(7). Available from: doi:10.3390/ijgi6070182

Kiviniemi, A. and Codinhoto, R., 2014. Challenges in the Implementation of BIM for FM—Case Manchester Town Hall Complex. In: *The Proceedings of the 2014 International Conference on Computing in Civil and Building Engineering, 23-25 June 2014 Orlando, Florida*, pp. 665-672. Available from: doi:10.1061/9780784413616.083.

Kohoutek, T., Mautz, R., and Wegner, J., 2013. Fusion of Building Information and Range Imaging for Autonomous Location Estimation in Indoor Environments, *Sensors* 13(2), pp. 2430-2446. Available from: doi:10.3390/s130202430

Konde, A., Tauscher, H., Biljecki, F., and Crawford, J., 2018. Floor Plans in CityGML, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-4/W6, pp. 25-32. Available from: doi:10.5194/isprs-annals-IV-4-W6-25-2018

Koopman, M., 2016. *3D Path-finding in a voxelized model of an indoor environment*. Delft University of Technology, The Netherlands. M.Sc. dissertation. Available from: https://repository.tudelft.nl/islandora/object/uuid:13788271-e19d-41e1-b827-fe7535a66281 [Accessed on: 28 June 2017].

Kornilov, A. and Safonov, I., 2018. An Overview of Watershed Algorithm Implementations in Open Source Libraries, *Journal of Imaging* 4(10). Available from: doi:10.3390/jimaging4100123

Kothuri, R., Godfrind, A., and Beinat, E., 2007. *Pro Oracle Spatial for Oracle Database 11g*, Apress, Berkeley, CA. Available from: doi:10.1007/978-1-4302-0446-6.

Kottman, C. and Reed, C., 2009. OGC 08-126. *The OpenGIS Abstract Specification Topic 5: Features*. Open Geospatial Consortium. Available from: https://portal.opengeospatial.org/files/?artifact_id=29536 [Accessed on: 11 December 2014].

Krijnen, T., 2020. IfcOpenShell [Code Library]. Available from: http://ifcopenshell.org [Accessed on: 8 June 2020].

Kroenke, D., Bunker, D., and Wilson, D., 2013. *Experiencing MIS*, 3rd ed., Pearson Education, Upper Saddle River, NJ..

Kumar, K., Labetski, A., Ohori, K.A., Ledoux, H., and Stoter, J., 2019. The LandInfra standard and its role in solving the BIM-GIS quagmire, *Open Geospatial Data, Software and Standards* 4(1), pp. 1-16. Available from: doi:10.1186/s40965-019-0065-z

Kurwi, S., Demian, P., and Hassan, T.M., 2017. Integrating BIM and GIS in railway projects: A critical review. In: Chan, P.W. & Nielson, C.J., eds., *Proceedings of 33rd Annual ARCOM Conference, 4-6 September 2017, Cambridge, UK*, pp. 45-53. Available from: http://www.arcom.ac.uk/-docs/proceedings/9d1a43cfcbac53e072a6d9e7446c71d0.pdf [Accessed on: 18 September 2019].

Kutzner, T., Chaturvedi, K., and Kolbe, T.H., 2020. CityGML 3.0: New Functions Open Up New Applications, *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science* 88(1), pp. 43-61. Available from: doi:10.1007/s41064-020-00095-z

Laiserin, J., 2002. Comparing Pommes and Naranjas [Online]. Available from: http://www.laiserin.com/features/issue15/feature01.php [Accessed on: 18 December 2015].

Laiserin, J., 2008. Foreword to the First Edition of BIM Handbook. In: Eastman, C.M., ed., *The BIM Handbook*, John Wiley & Sons, Hoboken, NJ.

Lapierre, A. and Cote, P., 2007. Using Open Web Services for urban data management: A testbed resulting from an OGC initiative for offering standard CAD/GIS/BIM services. In: *Urban and Regional Data Management: UDMS 2007 Annual*, CRC Press, Boca Raton, FL.

Lattuada, R., 2006. Three-dimensional representations and data structures in GIS and AEC. In: *Large-scale 3D data integration: challenges and opportunities*, CRC Press, Boca Raton, FL.

Lavy, S. and Jawadekar, S., 2014. A Case Study of Using BIM and COBie for Facility Management, *International Journal of Facility Management* 5(2). Available from: https://faculty.arch.tamu.edu/media/cms_page_media/2861/LavyJawadekar_2014.pdf [Accessed on: 25 October 2020].

Ledoux, H., 2013. On the Validation of Solids Represented with the International Standards for Geographic Information, *Computer-Aided Civil and Infrastructure Engineering* 28(9), pp. 693-706. Available from: doi:10.1111/mice.12043

Lee, J.K., Lee, J., Jeong, Y.S., Sheward, H., Sanguinetti, P., Abdelmohsen, S., and Eastman, C.M., 2012. Development of space database for automated building design review systems, *Automation in Construction* 24, pp. 203-212. Available from: doi:10.1016/j.autcon.2012.03.002

Lee, J., Li, K.-J., Zlatanova, S., Kolbe, T.H., Nagel, C., and Becker, T., 2018. OGC 14-005. *IndoorGML*. Open Geospatial Consortium. Available from: https://www.ogc.org/standards/indoorgml [Accessed on: 10 August 2020].

Lewiner, T., Lopes, H., Vieira, A.W., and Tavares, G., 2012. Efficient Implementation of Marching Cubes' Cases with Topological Guarantees, *Journal of Graphics Tools* 8(2), pp. 1-15. Available from: doi:10.1080/10867651.2003.10487582

Lewis, R. and Séquin, C., 1998. Generation of 3D Building Models from 2D Architectural Plans, *Computer-Aided Design* 30(10), pp. 765-779. Available from: doi:10.1016/S0010-4485(98)00031-1

Li, S. and Coleman, D., 2005. Modeling distributed GIS data production workflow, *Computers, Environment and Urban Systems* 29(4), pp. 401-424. Available from: doi:10.1016/j.compenvurbsys.2003.12.002

Li, F., Zlatanova, S., Koopman, M., Bai, X., and Diakité, A., 2018. Universal path planning for an indoor drone, *Automation in Construction* 95, pp. 275-283. Available from: doi:10.1016/j.autcon.2018.07.025

Lin, S., Gao, J., Koronios, A., and Chanana, V., 2007. Developing a data quality framework for asset management in engineering organisations, *International Journal of Information Quality* 1(1). Available from: doi:10.1504/IJIQ.2007.013378

Liu, X., Wang, X., Wright, G., Cheng, J., Li, X., and Liu, R., 2017. A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and Geographic Information System (GIS), *ISPRS International Journal of Geo-Information* 6(2). Available from: doi:10.3390/ijgi6020053

Lloyd, C., 2015. Case study: London Underground. In: Lloyd, C., ed., *International Case Studies in Asset Management*, ICE Publishing, London, pp. 131-139. Available from: doi:10.1680/icsiam.57395.131.

Longley, P.A., Goodchild, M.F., Maguire, D.J., and Rhind, D.W., 2005. Introduction. In: *Geographical Information Systems: Principles, Techniques, Management and Applications, 2nd Edition, Abridged*, Wiley, New York.

López, A.J. G., Crespo Márquez, A., Sanz, J.A., Kobbacy, K.A H., Shariff, S.M., Le Page, E., and González-Prida, V., 2017. Asset Management for Buildings within the Framework of Building Information Modeling Development. In: *Optimum Decision Making in Asset Management*, pp. 121-138. Available from: doi:10.4018/978-1-5225-0651-5.ch006.

Lou, R., 2011. *Modification of semantically enriched FE mesh models*. Arts et Métiers ParisTech. Ph.D. dissertation. Available from: https://pastel.archives-ouvertes.fr/pastel-00608202 [Accessed on: 7 August 2020].

Löwner, M.O., Gröger, G., Benner, J., Biljecki, F., and Nagel, C., 2016. Proposal for a New LoD and Multi-Representation Concept for CityGML, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-2-W1, pp. 3-12. Available from: doi:10.5194/isprs-annals-IV-2-W1-3-2016

Lu, Q., Xie, X., Heaton, J., Parlikad, A.K., and Schooling, J., 2019. From BIM Towards Digital Twin: Strategy and Future Development for Smart Asset Management. In: *Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future*, pp. 392-404. Available from: doi:10.1007/978-3-030-27477-1_30.

MacDonald, I., 2016. *Information Handover Principles*. Available from: https://learninglegacy.crossrail.co.uk/documents/information-handover-principles/ [Accessed on: 16 June 2020].

Maguire, D.J., 1991. An overview and definition of GIS. In: *Geographical Information Systems - Principles and Applications*, Longman Scientific & Technical, London.

Maher, M.L., Simoff, S.J., and Mitchell, J., 1997. Formalising building requirements using an Activity/Space Model, *Automation in Construction* 6(2), pp. 77-95. Available from: doi:10.1016/S0926-5805(96)00171-9

Maier, M.W., 1998. Architecting principles for systems-of-systems, *Systems Engineering* 1(4), pp. 267-284. Available from: doi:10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D

Mäntylä, M., 1988. *An introduction to solid modeling*, Computer Science Press, Rockville, MD.

Marzouk, M. and Abdelaty, A., 2014. BIM-based framework for managing performance of subway stations, *Automation in Construction* 41, pp. 70-77. Available from: doi:10.1016/j.autcon.2014.02.004

Marzouk, M. and Abdelaty, A., 2014a. Monitoring thermal comfort in subways using building information modeling, *Energy and Buildings* 84, pp. 252-257. Available from: doi:10.1016/j.enbuild.2014.08.006

Mason, A., 2017. Asset Management and Facilities Management-The Same but Different [Online]. Available from: https://www.linkedin.com/pulse/asset-management-facilities-management-the-same-different-mason/ [Accessed on: 10 September 2019].

Maurer, C.R., Qi, R., and Raghavan, V., 2003. A Linear Time Algorithm for Computing Exact Euclidean Distance Transforms of Binary Images in Arbitrary Dimensions, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25(2), pp. 265-270. Available from: doi:10.1109/TPAMI.2003.1177156

McPartland, R., 2014. BIM Levels explained [Online]. Available from: https://www.thenbs.com/knowledge/bim-levels-explained [Accessed on: 3 August 2020].

Meadows, D., 2008. *Thinking in Systems : A Primer*, Chelsea Green Publishing, White River Junction, Vermont.

MeshLab, 2020. MeshLab [Software]. Available from: https://www.meshlab.net [Accessed on: 9 September 2020].

Meyer, F. and Beucher, S., 1990. Morphological segmentation, *Journal of Visual Communication and Image Representation* 1(1), pp. 21-46. Available from: doi:10.1016/1047-3203(90)90014-M

Meyer, F., 1994. Topographic distance and watershed lines, *Signal Processing* 38(1), pp. 113-125. Available from: doi:10.1016/0165-1684(94)90060-4

Mignard, C. and Nicolle, C., 2014. Merging BIM and GIS using ontologies application to urban facility management in ACTIVe3D, *Computers in Industry* 65(9), pp. 1276-1290. Available from: doi:10.1016/j.compind.2014.07.008

Munir, M., Kiviniemi, A., and Jones, S., 2018. Business value of integrated BIM-based asset management, *Engineering, Construction and Architectural Management* 26(6), pp. 1171-1191. Available from: doi:10.1108/ECAM-03-2018-0105

Nagel, C., Stadler, A., and Kolbe, T.H., 2009. Conceptual requirements for the automatic reconstruction of building information models from uninterpreted 3D models, *ISPRS*

International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVIII-3-4/C3, pp. 46-53. Available from: https://www.isprs.org/proceedings/XXXVIII/3_4-C3/Paper_GeoW09/paper26_Nagel_Stadler_Kolbe.pdf [Accessed on: 11 December 2014].

National Institute of Building Sciences, 2014. Frequently asked questions about the National BIM Standard - United States [Online]. Available from: http://www.nationalbimstandard.org/faq.php [Accessed on: 22 July 2014].

National Institute of Building Sciences, 2015. *National BIM Standard - United States Version 3*. National Institute of Building Sciences, Washington, DC. Available from: https://www.nationalbimstandard.org [Accessed on: 21 January 2017].

NBS, 2015. *Uniclass 2015*. NBS, Newcastle, UK. Available from: https://www.thenbs.com/our-tools/uniclass-2015 [Accessed on: 6 February 2016].

NetworkX Developers, 2020. NetworkX [networkx.github.io]. Available from: https://networkx.github.io [Accessed on: 6 October 2020].

Noardo, F., Ellul, C., Harrie, L., Overland, I., Shariat, M., Arroyo Ohori, K., and Stoter, J., 2019. Opportunities and challenges for GeoBIM in Europe: developing a building permits use-case to raise awareness and examine technical interoperability challenges, *Journal of Spatial Science* 42(9), pp. 1-25. Available from: doi:10.1080/14498596.2019.1627253

Noardo, F., Ellul, C., Harrie, L., Devys, E., Arroyo Ohori, K., Olsson, P., and Stoter, J., 2019a. EuroSDR GeoBIM Project A Study in Europe on How to Use the Potentials of BIM and Geo Data in Practice, *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-4-W15, pp. 53-60. Available from: doi:10.5194/isprs-archives-XLII-4-W15-53-2019

Obermeyer, N.J., 1994. GIS: A New Profession?, *The Professional Geographer* 46(4), pp. 498-503. Available from: doi:10.1111/j.0033-0124.1994.00498.x

Obermeyer, N.J., 2007. GIS: The Maturation of a Profession, *Cartography and Geographic Information Science* 34(2), pp. 129-132. Available from: doi:10.1559/152304007781002280

OGC and buildingSMART, 2017. *Joint OGC / bSI Whitepaper - Agreed Framework for Collaboration*. OGC 17-055. Open Geospatial Consortium / buildingSMART. Available from: https://portal.opengeospatial.org/files/?artifact_id=74203&version=1 [Accessed on: 25 January 2021].

OGC, 2016. OGC 15-111r1. *OGC® Land and Infrastructure Conceptual Model Standard (LandInfra)*. Open Geospatial Consortium. Available from: https://www.ogc.org/standards/infragml [Accessed on: 29 June 2020].

OGC, 2020. OGC 19-008r4. *GeoTIFF Standard*. Open Geospatial Consortium. Available from: https://www.ogc.org/standards/geotiff [Accessed on: 29 June 2020].

OGC, 2021. GitHub - opengeospatial/CityGML-3.0CM: CityGML 3.0 Conceptional Model, *www.ogc.org*. Available from: https://github.com/opengeospatial/CityGML-3.0CM [Accessed on: 28 August 2021].

OGC, 2021a. 12-128r17. *GeoPackage Encoding Standard 1.3*. Open Geospatial Consortium. Available from: https://docs.ogc.org/is/12-128r17/12-128r17.html [Accessed on: 8 April 2021].

Open Cascade, 2020. Open Cascade [Code Library]. Available from: https://www.opencascade.com [Accessed on: 8 June 2020].

OpenSCAD, 2020. OpenSCAD [Software]. Available from: http://www.openscad.org [Accessed on: 14 February 2020].

Oracle Corporation, 2020. cxOracle [Python Package]. Available from: https://oracle.github.io/python-cx_Oracle/ [Accessed on: 6 October 2020].

Oxford English Dictionary, 2019. Model. In: *Oxford English Dictionary*, Oxford University Press, Oxford. Available from: https://www.oed.com [Accessed on: 10 October 2019].

Palmgren, P.O., 2017. *Approach to Master Data Management at Crossrail - Crossrail Learning Legacy*. Crossrail Learning Legacy. Available from: https://learninglegacy.crossrail.co.uk/documents/approach-master-data-management-crossrail/ [Accessed on: 7 July 2020].

Pang, Y., Zhang, C., Zhou, L., Lin, B., and Lv, G., 2018. Extracting Indoor Space Information in Complex Building Environments, *ISPRS International Journal of Geo-Information* 7(8). Available from: doi:10.3390/ijgi7080321

Patel, P., 2018. *Crossrail Learning Legacy: Management and Client Assurance of Contractual Drawing Deliverables*. Available from: https://learninglegacy.crossrail.co.uk/documents/management-and-client-assurance-of-contractual-drawing-deliverables/ [Accessed on: 23 June 2020].

Paviot, T., 2019. PythonOCC [Python Package]. Available from: http://www.pythonocc.org [Accessed on: 6 October 2020].

Peuquet, D.J., 1984. A conceptual framework and comparison of spatial data models, *Cartographica: The International Journal for Geographic Information and Geovisualization* 21(4), pp. 66-113. Available from: doi:10.3138/D794-N214-221R-23R5

PostGIS, 2020. PostGIS — Spatial and Geographic Objects for PostgreSQL [Software]. Available from: https://postgis.net [Accessed on: 29 June 2020].

Prospects, 2020. Geographical Information Systems Officer Job Profile [Online]. Available from: https://www.prospects.ac.uk/job-profiles/geographical-information-systems-officer [Accessed on: 26 March 2020].

Python Software Foundation, 2020. Python [Language]. Available from: https://www.python.org [Accessed on: 6 October 2020].

Qt Company, 2020. Qt Graphical User Interface [Code Library]. Available from: https://www.qt.io [Accessed on: 6 October 2020].

Reeve, D.E. and Petch, J., 1999. *GIS, Organisations and People: A Socio-technical Approach*, CRC Press, Boca Raton, FL.

Requicha, A.G., 1980. Representations for Rigid Solids: Theory, Methods, and Systems, *ACM Computing Surveys* 12(4), pp. 437-464. Available from: doi:10.1145/356827.356833

Riverbank Computing, 2020. PyQt [Python Package]. Available from: https://riverbankcomputing.com/software/pyqt/intro [Accessed on: 6 October 2020].

Roerdink, J.B T M. and Meijster, A., 2000. The Watershed Transform: Definitions, Algorithms and Parallelization Strategies, *Fundamenta Informaticae* 41(1,2), pp. 187-228. Available from: doi:10.3233/FI-2000-411207

Romero-Zaliz, R. and Reinoso-Gordo, J.F., 2017. An Updated Review on Watershed Algorithms. In: *Soft Computing for Sustainability Science*, Cham, pp. 235-258. Available from: doi:10.1007/978-3-319-62359-7_12.

Roswell, C., 2012. Modeling of Geographic Information. In: *Springer Handbook of Geographic Information*, Springer, Heidelberg, pp. 3-17. Available from: doi:10.1007/978-3-540-72680-7_1.

Sackey, E., Tuuli, M., and Dainty, A., 2015. Sociotechnical Systems Approach to BIM Implementation in a Multidisciplinary Construction Context, *Journal of Management in Engineering* 31(1). Available from: doi:10.1061/(ASCE)ME.1943-5479.0000303

Sacks, R., Eastman, C.M., and Lee, G., 2004. Parametric 3D modeling in building construction with examples from precast concrete, *Automation in Construction*. Available from: doi:10.1016/S0926-5805(03)00043-8

Safe Software, 2017. FME [Software]. Available from: www.safe.com [Accessed on: 27 April 2017].

Safe Software, 2017a. MicroStation DGN Reader/Writer [Online]. Available from: https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_ReadersWriters/igds/igds.htm [Accessed on: 27 Apr 17].

Santos, R., Costa, A.A., and Grilo, A., 2017. Bibliometric analysis and review of Building Information Modelling literature published between 2005 and 2015, *Automation in Construction* 80, pp. 118-136. Available from: doi:10.1016/j.autcon.2017.03.005

Schaller, J., Gnaedinger, J., Reith, L., Freller, S., and Mattos, C., 2017. GeoDesign: Concept for Integration of BIM and GIS in Landscape Planning, *Journal of Digital Landscape Architecture* 2, pp. 102-112. Available from: doi:10.14627/537629011

Scherer, R.J., 2007. Product Model Based Collaboration. In: *Proceedings of the 24th CIB W78 Conference, 26-29 June 2007, Maribor, Slovenia*, pp. 11-20. Available from: https://itc.scix.net/pdfs/w78-2007-002-201-Scherer.pdf [Accessed on: 23 September 2015].

Schevers, H., Mitchell, J., Akhurst, P., Marchant, D., Bull, S., McDonald, K., Drogemuller, R., and Linning, C., 2007. Towards digital facility modelling for Sydney opera house using IFC and semantic web technology, *Journal of Information Technology in Construction (ITcon)* 12, pp. 347-362. Available from: http://www.itcon.org/2007/24 [Accessed on: 9 September 2019].

scikit-image, 2020. SciPy Image - Watershed Algorithm [Python Package]. Available from: http s://github.com/scikit-image/scikit-image/blob/master/skimage/segmentation/_watershed.py [Accessed on: 21 February 2020].

SciPy Community, 2019. The SciPy Distance Transform [Online]. Available from: https://docs.scipy.org/doc/scipy/reference/generated/scipy.ndimage.distance_transfo rm_edt.html#scipy.ndimage.distance_transform_edt [Accessed on: 17 May 2019].

Scott, J., 2015. Transforming asset management: a UK perspective. In: *Proceedings of APTA 2015 Rail Conference, 21-24 June, Salt Lake City, Utah*. Available from: https://www.networkrailconsulting.com/news-and-publications/publications/transforming-as set-management-a-uk-perspective/ [Accessed on: 4 February 2020].

Sergeeva, Natalya and Zanello, Chiara, 2018. Championing and promoting innovation in UK megaprojects, *International Journal of Project Management* 36(8), pp. 1068-1081. Available from: doi:10.1016/j.ijproman.2018.09.002

Shen, G., Sakkalis, T., and Patrikalakis, N., 2001. Analysis of boundary representation model rectification. In: *Proceedings of the 6th ACM Symposium on Solid Modeling and Applications, May 2001, Ann Arbor, MI*, pp. 149-158. Available from: doi:10.1145/376957.376975.

Shewchuk, J.R., 2002. Delaunay refinement algorithms for triangular mesh generation, *Computational Geometry* 22(1-3), pp. 21-74. Available from: doi:10.1016/S0925-7721(01)00047-5

Sijbers, J., Scheunders, P., Verhoye, M., Van der Linden, A., van Dyck, D., and Raman, E., 1997. Watershed-based segmentation of 3D MR data for volume quantization, *Magnetic Resonance Imaging* 15(6), pp. 679-688. Available from: doi:10.1016/S0730-725X(97)00033-7

Soille, P.J. and Ansoult, M.M., 1990. Automated basin delineation from digital elevation models using mathematical morphology, *Signal Processing* 20(2), pp. 171-182. Available from: doi:10.1016/0165-1684(90)90127-K

Soille, P. and Vincent, L.M., 1990. Determining watersheds in digital pictures via flooding simulations. In: Kunt, M., ed., *SPIE Proceedings of Visual Communications and Image Processing, 2-4 October 1990, Lausanne, Switzerland*, pp. 240-250. Available from: doi:10.1117/12.24211.

Solibri, 2017. Solibri Model Viewer - Solibri [Software]. Available from: https://www.solibri.com/products/solibri-model-viewer/ [Accessed on: 27 April 2017].

Somers, R., 1998. Developing GIS Management Strategies for an Organization, *Journal of Housing Research* 9(1), pp. 157-178. Available from: doi:10.5555/jhor.9.1.j9756u260n255224

Succar, B., 2009. Building information modelling framework: A research and delivery foundation for industry stakeholders, *Automation in Construction* 18(3), pp. 357-375. Available from: doi:10.1016/j.autcon.2008.10.003

Svensson, K., 1998. *Integrating Facilities Management Information - A Process and Product Model Approach*. Royal Institute of Technology. D.Tech. dissertation. Available from: http://swepub.kb.se/bib/swepub:oai:DiVA.org:kth-2646?language=en [Accessed on: 10 February 2020].

Taylor, M., 2017. Crossrail Project: Building a Virtual Version of London's Elizabeth line, *Proceedings of the Institution of Civil Engineers - Bridge Engineering* 170(6), pp. 56-63. Available from: doi:10.1680/jcien.17.00022

Thabet, W., Lucas, J., and Johnston, S., 2016. A Case Study for Improving BIM-FM Handover for a Large Educational Institution. In: *Construction Research Congress 31 May– 2 Jun 2016, San Juan, Puerto Rico*, pp. 2177-2186. Available from: doi:10.1061/9780784479827.217.

Then, D.S. and Akhlaghi, F., 1992. A framework for defining facilities management education. In: *Facilities Management: Research Directions*, Royal Institution of Chartered Surveyors, London.

Tobler, W.R., 1970. A Computer Movie Simulating Urban Growth in the Detroit Region, *Economic Geography* 46. Available from: doi:10.2307/143141

Tolk, A., Diallo, S.Y., and Turnitsa, C.D., 2007. Applying the Levels of Conceptual Interoperability Model in Support of Integratability, Interoperability, and Composability for System-of-Systems Engineering, *Journal of Systems, Cybernetics, and Informatics* 5(5), pp. 65-74. Available from: https://digitalcommons.odu.edu/msve_fac_pubs/27 [Accessed on: 14 January 2020].

Tolk, A., 2006. What Comes After the Semantic Web - PADS Implications for the Dynamic Web. In: *Proceedings of 20th Workshop on Principles of Advanced and Distributed Simulation, 24-26 May 2006, Singapore*, pp. 55-62. Available from: doi:10.1109/PADS.2006.39.

Tomlinson, R., 2013. *Thinking about GIS: Geographic Information System Planning for Managers*, 5th ed., Esri Press, Redlands, CA.

Transport for London, 2011. LU 1-026. *Topographical Surveys and Mapping*. Transport for London, London. Available from: http://www.engineeringsurveyor.com/software/1-026\%20-\%20Topographical\%20Surveys\%20and\%20Mapping.pdf [Accessed on: 27 April 2017].

Tsou, M.-H., 2018. The Future Development of GISystems, GIScience, and GIServices. In: *Comprehensive Geographic Information Systems*, pp. 1-4. Available from: doi:10.1016/B978-0-12-409548-9.09631-7.

University College London, 2020. Geospatial Sciences (Building Information Modelling and Surveying) Master of Science Prospectus [Online]. Available from: https://www.ucl.ac.uk/prospective-students/graduate/taught-degrees/geospatial-sciences-building-information-modelling-and-surverying-msc [Accessed on: 9 March 2020].

University of Derby, 2020. University Diploma in CAD and BIM Programme Specification [Online]. Available from: https://www.derby.ac.uk/undergraduate/architecture-architectural-technology-courses/university-diploma-in-cad-and-bim-architecture/ [Accessed on: 9 March 2020].

van Berlo, L. and Krijnen, T., 2014. Using the BIM Collaboration Format in a Server Based Workflow, *Procedia Environmental Sciences* 22, pp. 325-332. Available from: doi:10.1016/j.proenv.2014.11.031

van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., and Yu, T., 2014. scikit-image: image processing in Python, *PeerJ* 2(2). Available from: doi:10.7717/peerj.453

van Nederveen, S., Wolfert, R., and van de Ruitenbeek, M., 2014. From BIM to life cycle information management in infrastructure. In: *eWork and eBusiness in Architecture, Engineering and Construction: ECPPM 2014*, Taylor & Francis, London, pp. 115-119. Available from: doi:10.1201/b17396-22.

van Treeck, C. and Rank, E., 2006. Dimensional reduction of 3D building models using graph theory and its application in building energy simulation, *Engineering with Computers* 23(2), pp. 109-122. Available from: doi:10.1007/s00366-006-0053-7

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al., 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python, *Nature Methods* 17(3), pp. 261-272. Available from: doi:10.1038/s41592-019-0686-2

Wallace, R., 2013. Phantom Elements [Online]. Available from: https://communities.bentley.com/products/microstation/f/microstation-forum/89753/phantom-elements [Accessed on: 28 August 2021].

Wang, H., Hamilton, A., Song, Y., and Tah, J., 2007. Implementation of a Data Integration Service for Building and Urban Planning, *Architectural Engineering and Design Management* 3(3), pp. 195-208. Available from: doi:10.1080/17452007.2007.9684642

Wegner, S., Oswald, H., and Fleck, E., 1998. 3D Watershed Transformation on Graphs. In: Hanson, K.M., ed., *SPIE Proceedings of Medical Imaging, 24 June 1998, San Diego, US*, pp. 714-725. Available from: doi:10.1117/12.310951.

Whyte, J., Coca, D., Fitzgerald, J., Mayfield, M., Pierce, K., Shah, N., Chen, L., Gamble, C., Genes, C., Babovic, F., et al., 2019. *Analysing Systems Interdependencies Using a Digital Twin*. CDBB. 2018/19 General Project funded by CDBB, Final Report. Available from: https://www.repository.cam.ac.uk/handle/1810/296269 [Accessed on: 8 August 2021].

Wise, S., 2010. GIS data modelling-lessons from the analysis of DTMs, *International Journal of Geographical Information Science* 14(4), pp. 313-318. Available from: doi:10.1080/13658810050024250

Wix, J. and Bloomfield, D., 1995. Standardisation in the Building Industry - The STEP Building Construction Core Model. In: Fischer, M.A., Law, K.H., & Luiten, B., eds., *Proceedings of the 1996 W78 Conference, Stanford, USA, 22-24 August, 1995.*, pp. 184-195. Available from: http://itc.scix.net/paper/w78-1995-184 [Accessed on: 7 May 2020].

Worboys, M. and Duckham, M., 2004. *GIS: A Computing Perspective*, 2nd ed., CRC Press, Boca Raton, FL.

Worboys, M., 1994. Object-oriented approaches to geo-referenced information, *International Journal of Geographical Information Science* 8(4), pp. 385-399. Available from: doi:10.1080/02693799408902008

Xiong, Q., Zhu, Q., Du, Z., Zlatanova, S., Zhang, Y., Zhou, Y., and Li, Y., 2016. Free multi-floor indoor space extraction from complex 3D building models, *Earth Science Informatics* 10(1), pp. 69-83. Available from: doi:10.1007/s12145-016-0279-x

Zach, J., 2017. Asset Management is about Managing Assets [Online]. Available from: http://www.uberlytics.com/asset-management-is-about-managing-assets/ [Accessed on: 17 December 2017].

Zack, M., 1999. Developing a Knowledge Strategy, *California Management Review* 41(3). Available from: https://www.proquest.com/docview/216150288 [Accessed on: 30 August 2021].

Zhang, X., Arayici, Y., Wu, S.-W., Abbot, C., and Aouad, G., 2009. Integrating BIM and GIS for large scale (building) asset management: a critical review. In: *Proceedings of 12th International Conference on Civil, Structural and Environmental Engineering Computing, 1-4 Sep 2009, Funchal, Madeira, Portugal*. Available from: http://usir.salford.ac.uk/11418/ [Accessed on: 11 December 2014].

Zhu, J., Wright, G., Wang, J., and Wang, X., 2018. A Critical Review of the Integration of Geographic Information System and Building Information Modelling at the Data Level, *ISPRS International Journal of Geo-Information* 7(2). Available from: doi:10.3390/ijgi7020066

Zhu, J., Wang, X., Chen, M., Wu, P., and Kim, M.J., 2019. Integration of BIM and GIS: IFC geometry transformation to shapefile using enhanced open-source approach, *Automation in Construction* 106. Available from: doi:10.1016/j.autcon.2019.102859

Zigelski, S., 2015. STEP export not complete - missing parts [Online]. Available from: https://communities.bentley.com/products/microstation/w/microstation__wiki/19220/step-export-not-complete-missing-parts [Accessed on: 28 August 2021].

Zlatanova, S., Yan, J., Wang, Y., Diakité, A., Isikdag, U., Sithole, G., and Barton, J., 2020. Spaces in Spatial Science and Urban Applications—State of the Art Review, *ISPRS International Journal of Geo-Information* 9(1). Available from: doi:10.3390/ijgi9010058

Zottie, 2005. File:Csg tree.png - Wikimedia Commons, *commons.wikimedia.org*. Available from: https://commons.wikimedia.org/wiki/File:Csg_tree.png [Accessed on: 8 June 2020].

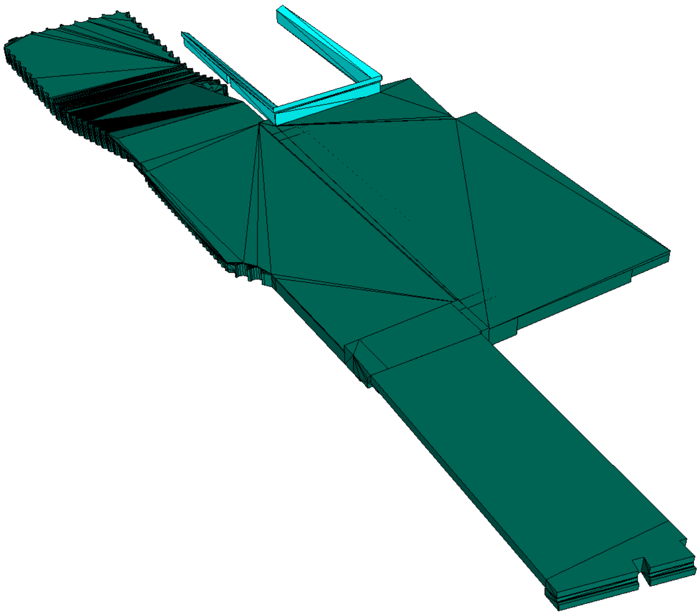
A Crossrail MicroStation CAD Model Files

List of model files

File	Type	Description
LPL-C-1-41051	Structural	Broadgate ticket hall ceiling
LPL-C-2-41052	Structural	Broadgate ticket hall and escalator descent
LPL-E-1-42201	Electrical	Broadgate ticket hall electrical fittings
LPL-E-2-42205	Electrical	Broadgate ticket hall electrical fittings
LPL-C4-00301	Structural	Liverpool Street station platform and access tunnels
MES-S-00004	Structural	Mile End Shaft bottom level
MES-S-00007	Structural	Mile End Shaft mid level
MES-A-2-00001	Architectural	Mile End Shaft bottom level non-structural walls and doors
MES-A-Z-31749	Architectural	Mile End Shaft stairway

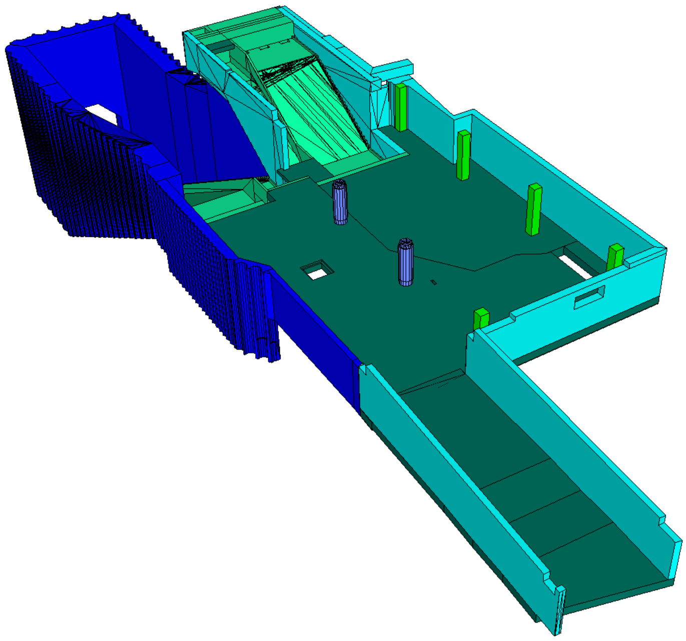
A.1 LPL-C-1-41052

- Broadgate ticket hall ceiling



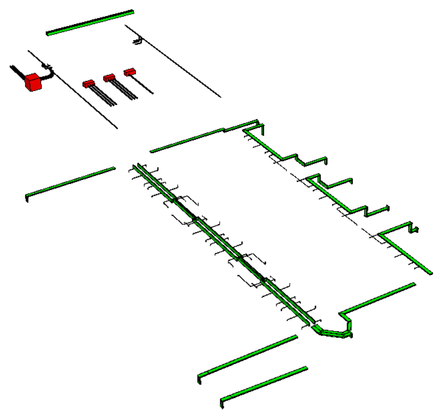
A.2 LPL-C-2-41051

- Broadgate Ticket hall and escalator descents



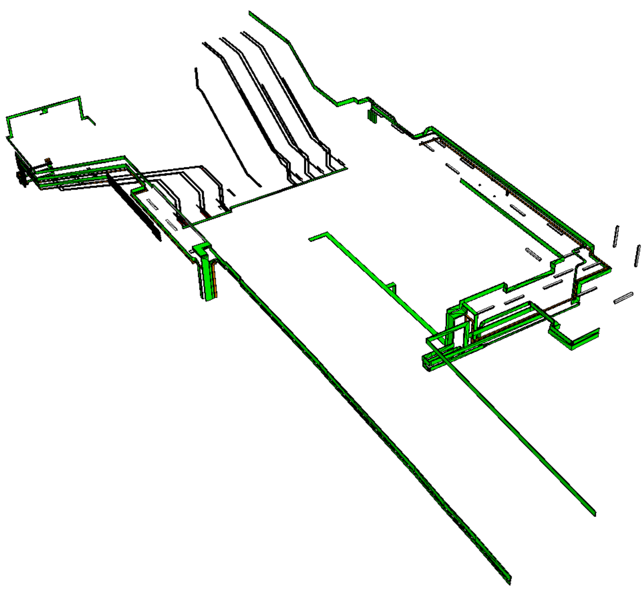
A.3 LPL-E-1-42201

- Broadgate ticket hall electrical fittings (Level 1)



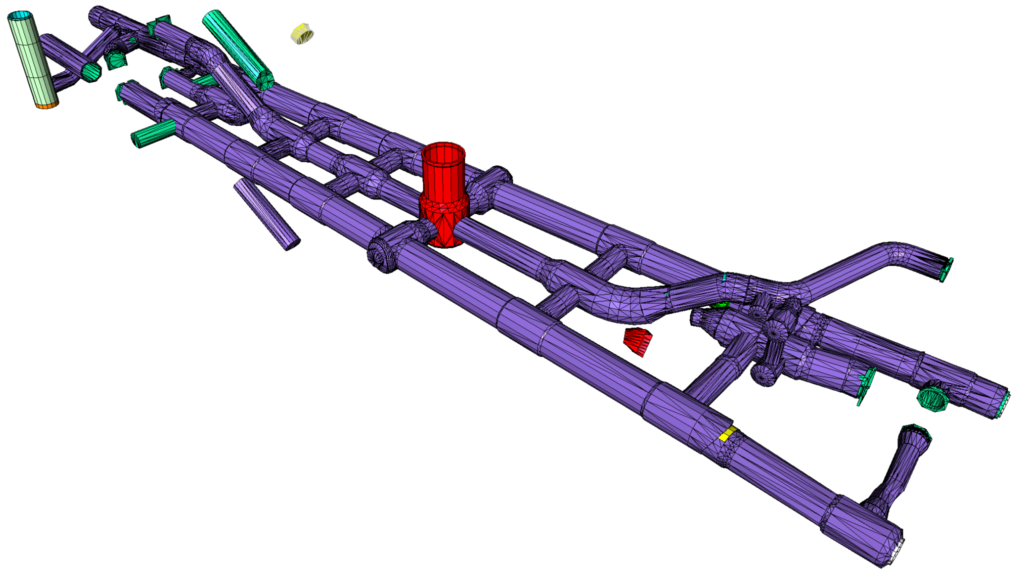
A.4 LPL-E-2-42205

- Broadgate ticket hall electrical fittings (Level 2)



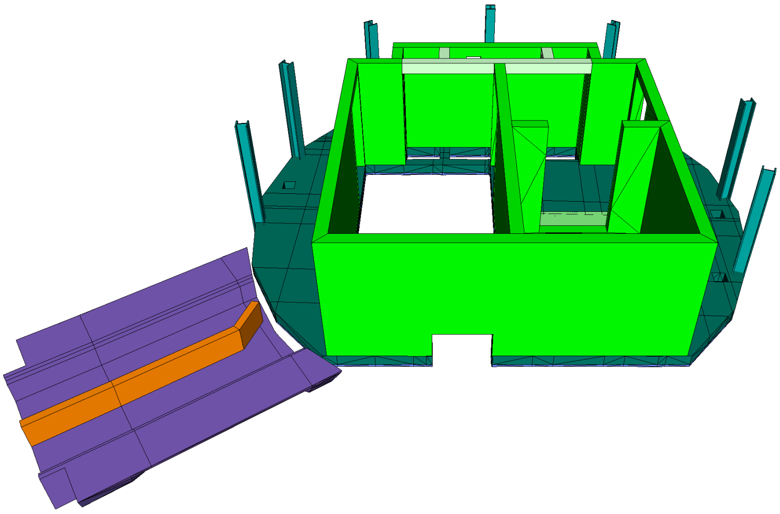
A.5 LPL-C4-00301

- Liverpool Street station platform tunnels and access tunnels



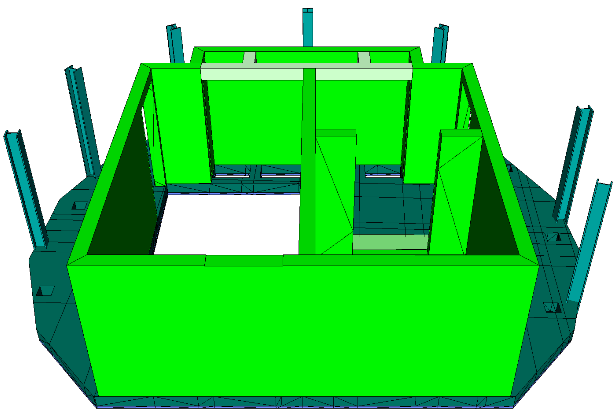
A.6 MES-S-00004

- Mile End Shaft bottom level



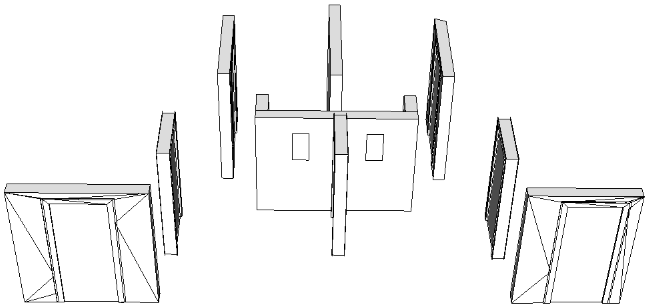
A.7 MES-S-00007

- Mile End Shaft mid level



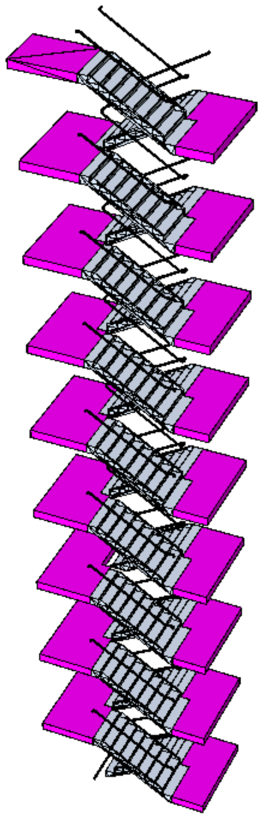
A.8 MES-A-2-00001

- Mile End Shaft bottom level with non-structural walls and doors



A.9 MES-A-Z-31749

- Mile End Shaft staircase



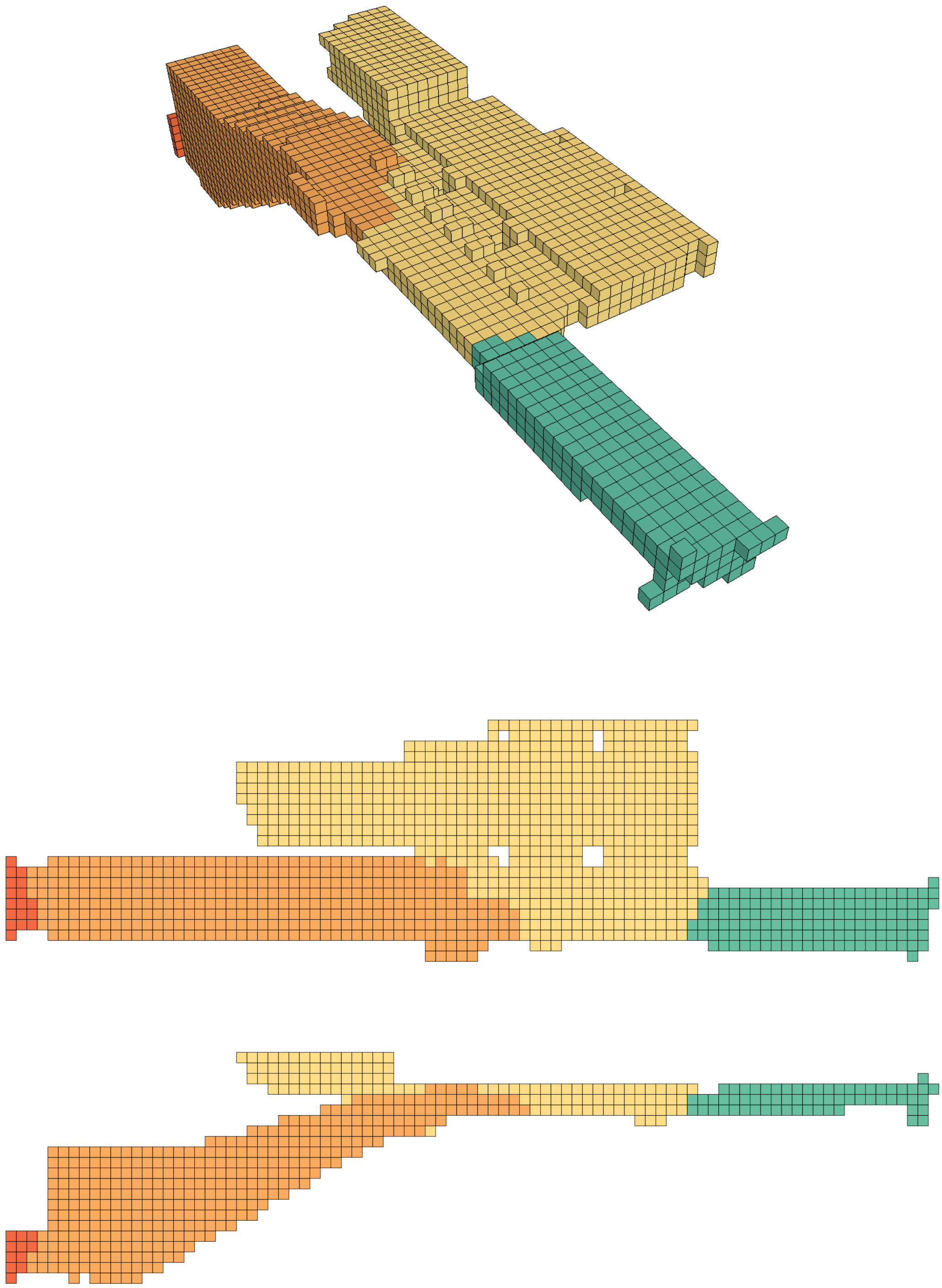
B Broadgate Ticket Hall Watershed Segmentation Output

List of voxelised output

	Location	Size	Discretisation
B1	Broadgate Ticket Hall	1.0	2.0
B2	Broadgate Ticket Hall	0.5	2.0
B3	Broadgate Ticket Hall	0.25	2.0
B4	Broadgate Ticket Hall	1.0	1.0
B5	Broadgate Ticket Hall	0.5	1.0
B6	Broadgate Ticket Hall	0.25	1.0
B7	Broadgate Ticket Hall	1.0	0.5
B8	Broadgate Ticket Hall	0.5	0.5
B9	Broadgate Ticket Hall	0.25	0.5

B.1 Segmentation using 1.0 m, DF = 2.0

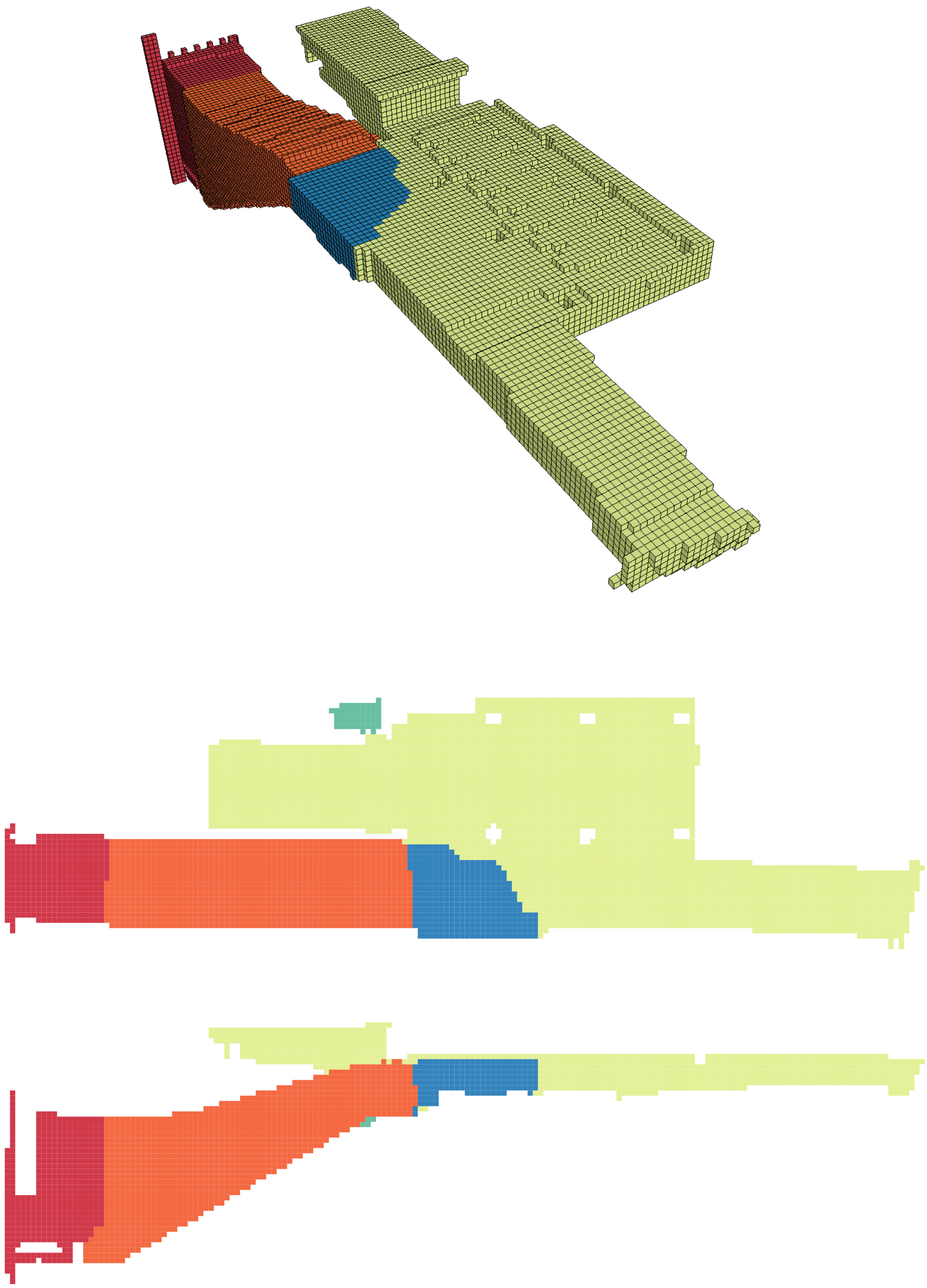
- Broadgate Ticket Hall
- Voxel size 1.0 m
- Discretisation Factor 2.0



Perspective, Plan and Side Elevation views

B.2 Segmentation using 0.5 m, DF = 2.0

- Broadgate Ticket Hall
- Voxel size 0.5 m
- Discretisation Factor 2.0



Perspective, Plan and Side Elevation views

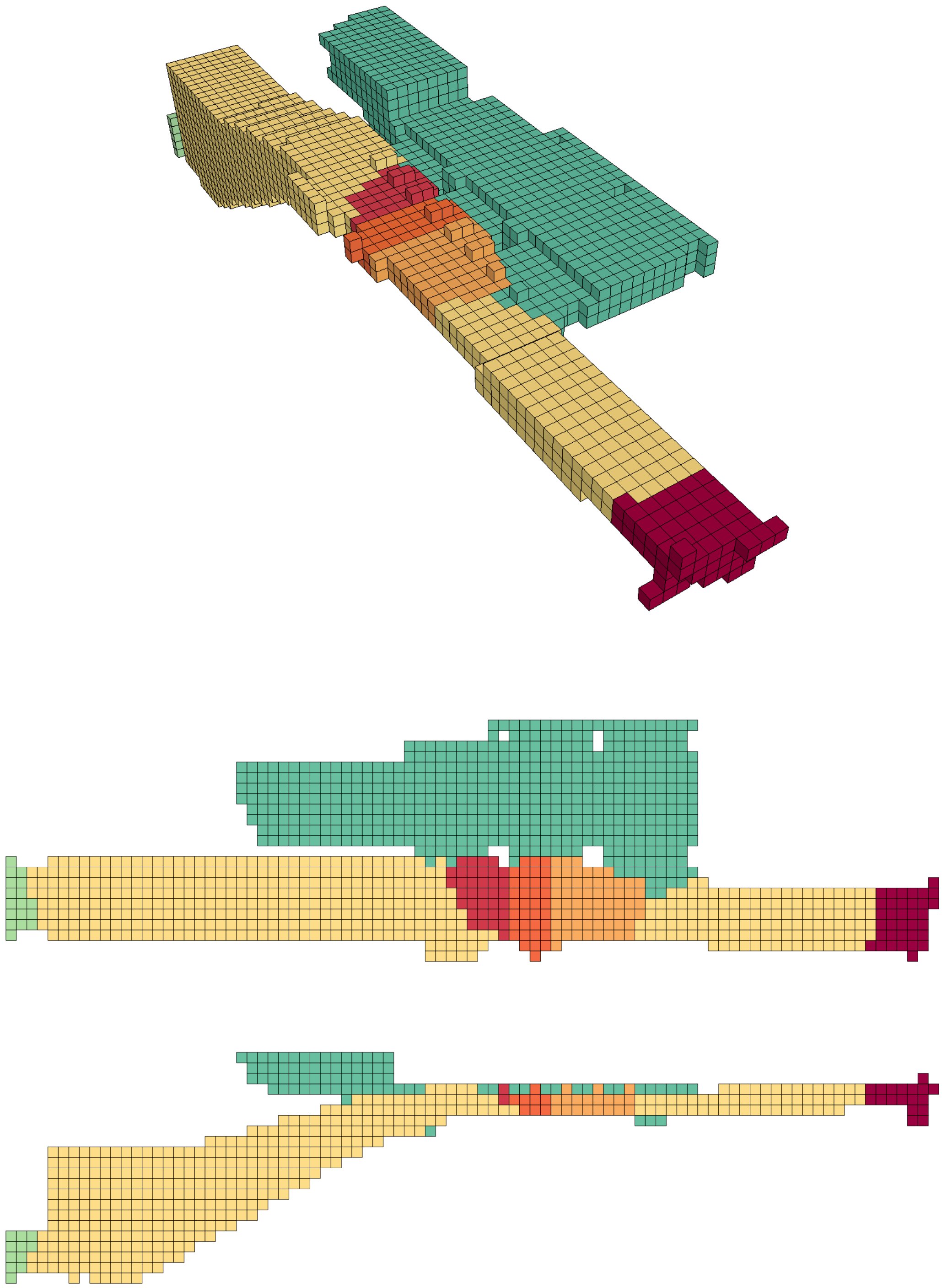
B.3 Segmentation using 0.25 m, DF = 2.0

- Broadgate Ticket Hall
- Voxel size 0.25 m
- Discretisation Factor 2.0

Perspective, Plan and Side Elevation views

B.4 Segmentation using 1.0 m, DF = 1.0

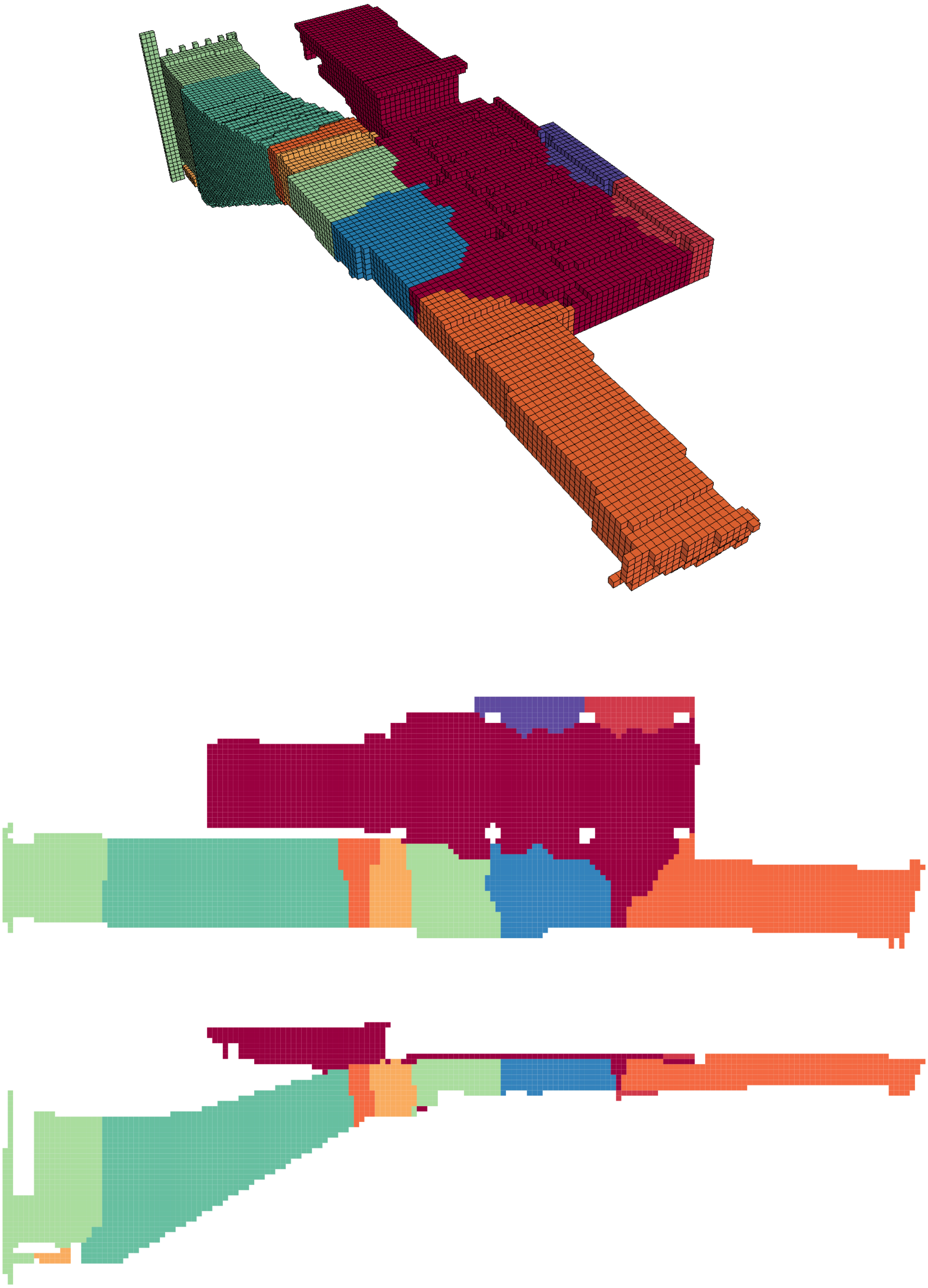
- Broadgate Ticket Hall
- Voxel size 1.0 m
- Discretisation Factor 1.0



Perspective, Plan and Side Elevation views

B.5 Segmentation using 0.5 m, DF = 1.0

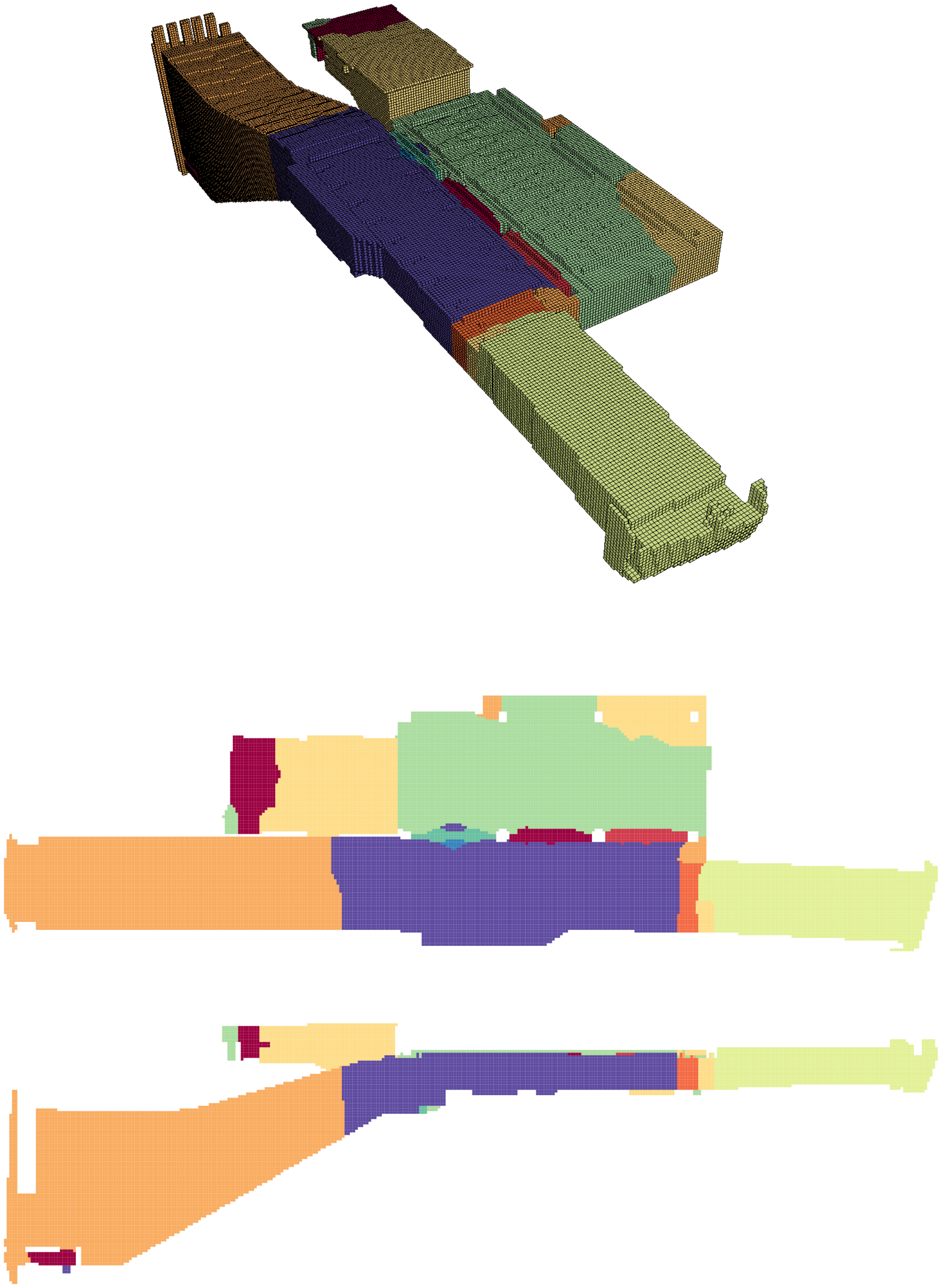
- Broadgate Ticket Hall
- Voxel size 0.5 m
- Discretisation Factor 1.0



Perspective, Plan and Side Elevation views

B.6 Segmentation using 0.25 m, DF = 1.0

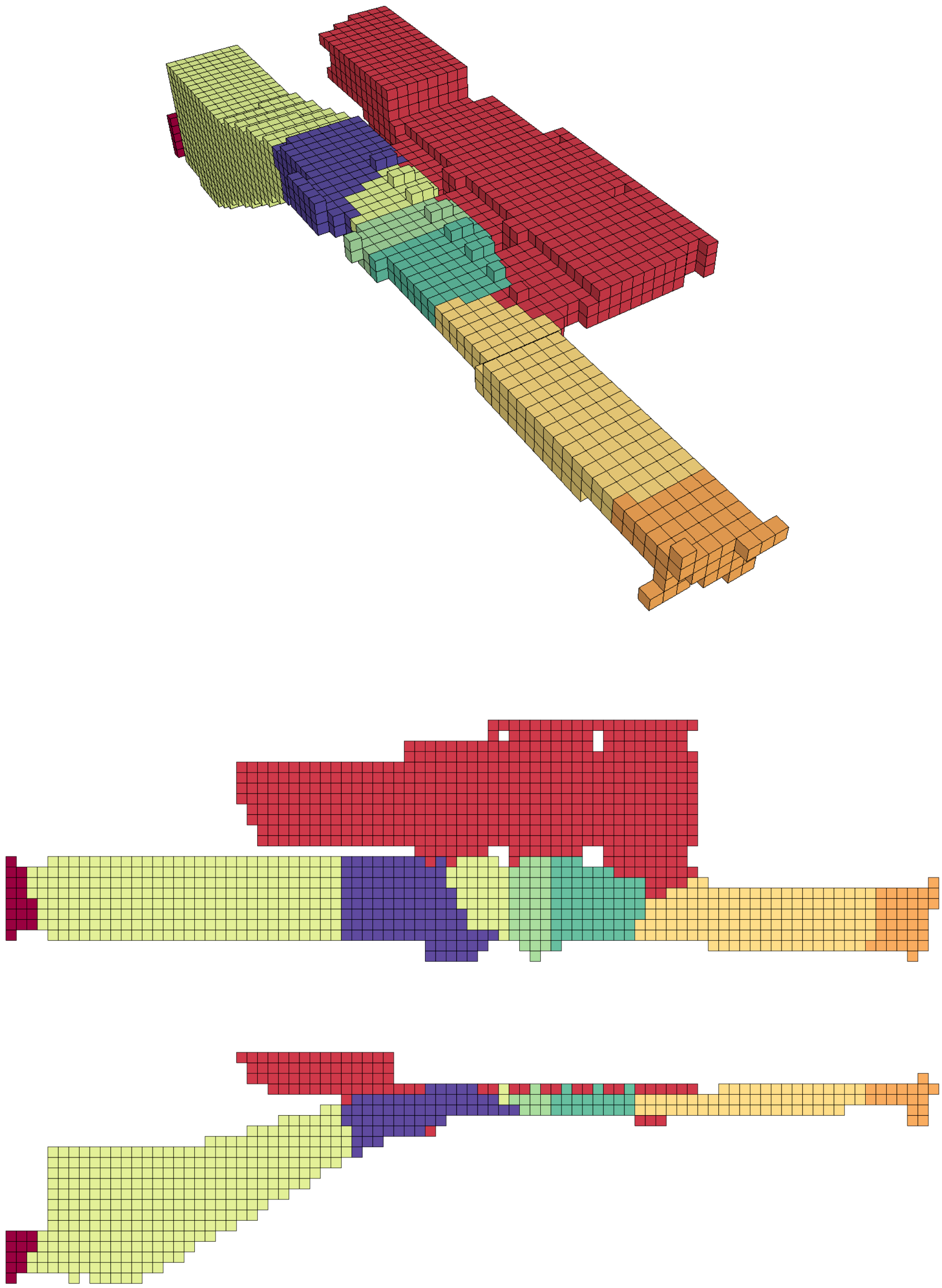
- Broadgate Ticket Hall
- Voxel size 0.25 m
- Discretisation Factor 1.0



Perspective, Plan and Side Elevation views

B.7 Segmentation using 1.0 m, DF = 0.5

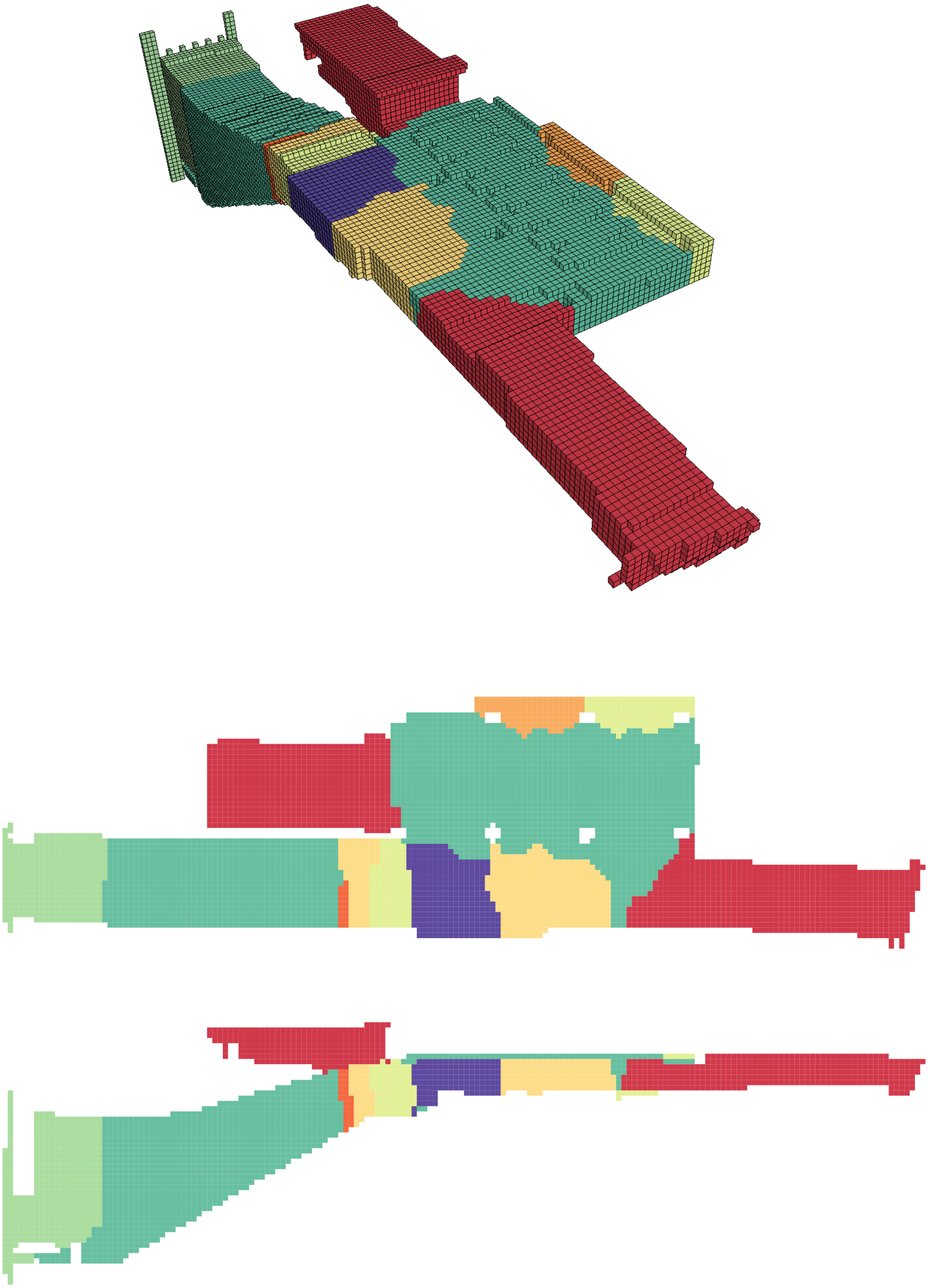
- Broadgate Ticket Hall
- Voxel size 1 m
- Discretisation Factor 0.5



Perspective, Plan and Side Elevation views

B.8 Segmentation using 0.5 m, DF = 0.5

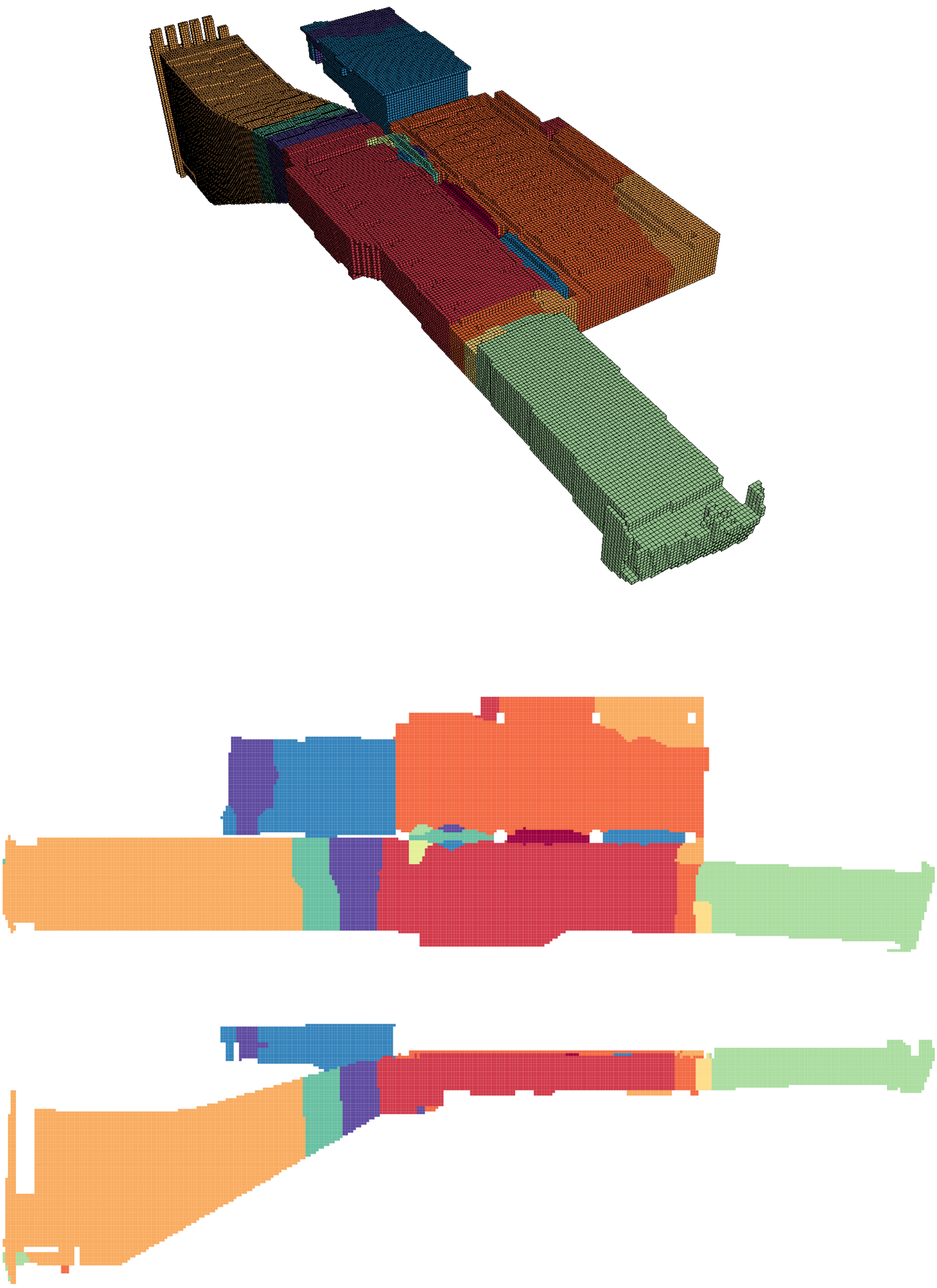
- Broadgate Ticket Hall
- Voxel size 0.5m
- Discretisation Factor 0.5



Perspective, Plan and Side Elevation views

B.9 Segmentation using 0.25 m, DF = 0.5

- Broadgate Ticket Hall
- Voxel size 0.25m
- Discretisation Factor 0.5



Perspective, Plan and Side Elevation views

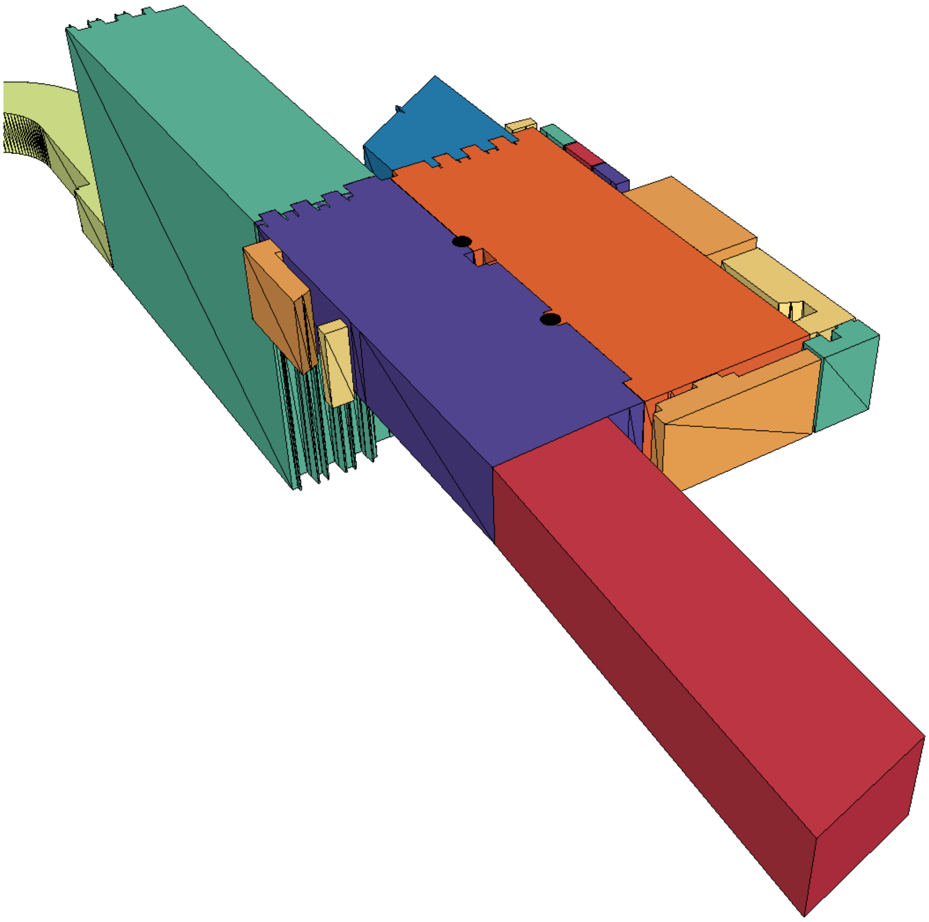
C Broadgate Ticket Hall Spaces Prepared for Spatial Query

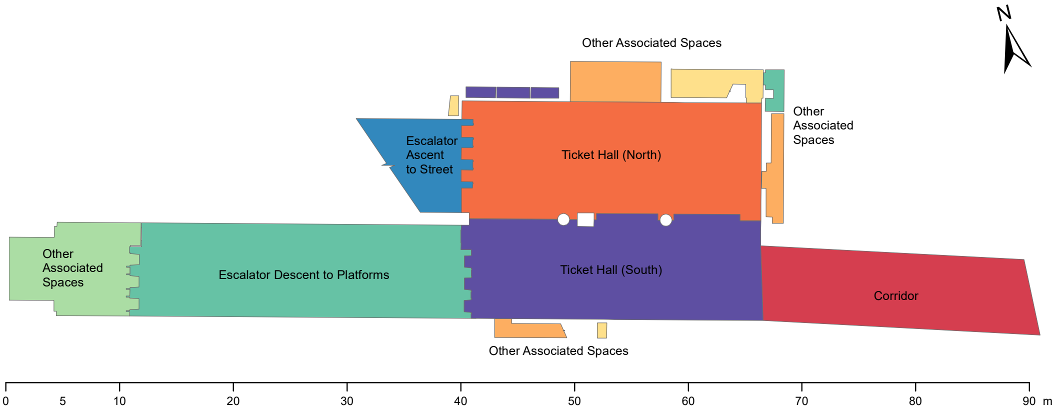
List of space sets

Appendix	Space Set	Description
C1	BTH-EXTRUDED	Extruded Floor Plan
C2	BTH-AGGREGATED	Extruded Floor Plan Minor spaces aggregated into 5 principal spaces
C3	BTH-MANIFOLD	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 5 principal spaces Converted to Manifold B-Rep
C4	BTH-DILATED	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 5 principal spaces Dilated by one voxel (0.25 m) Converted to Manifold B-Rep
C5	BTH-CLIPPED	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 5 principal spaces Dilated by one voxel (0.25 m) Converted to Manifold B-Rep Clipped with building element features Used for visualisation only

C.1 BTH-EXTRUDED Space Set

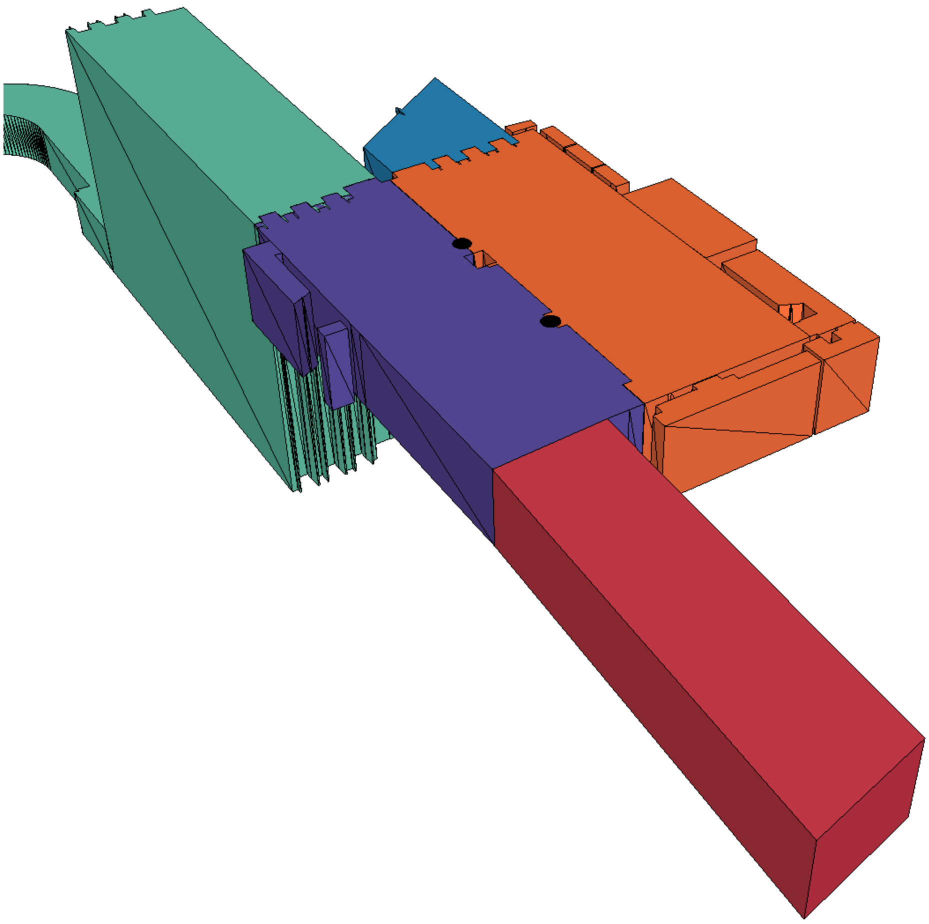
- Broadgate Ticket Hall
- Extruded floor plan

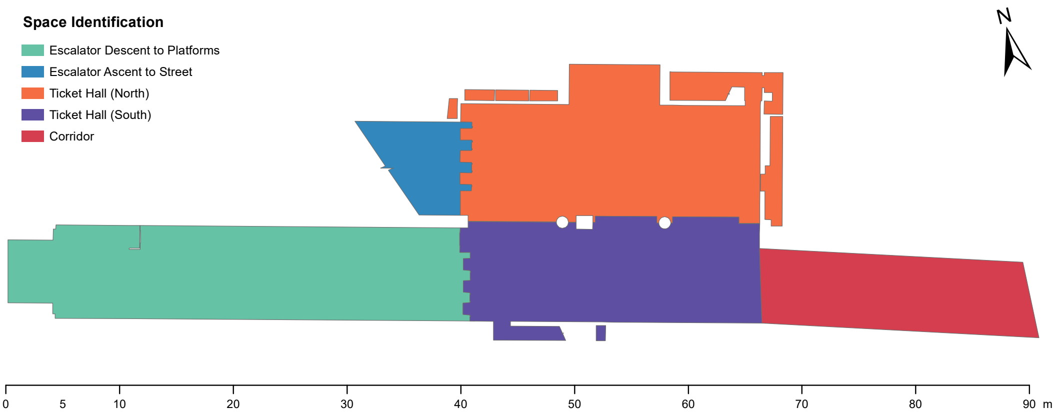




C.2 BTH-AGGREGATED Space Set

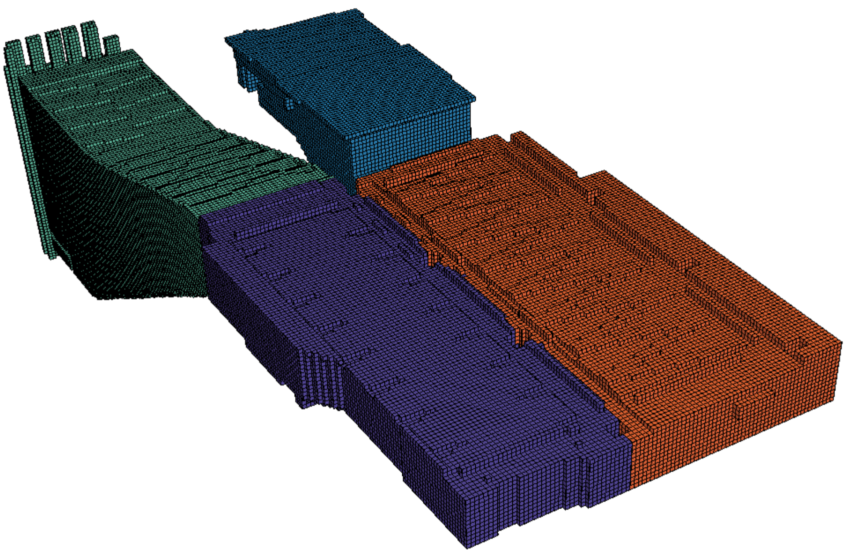
- Broadgate Ticket Hall
- Aggregated spaces created from extruded floor plan

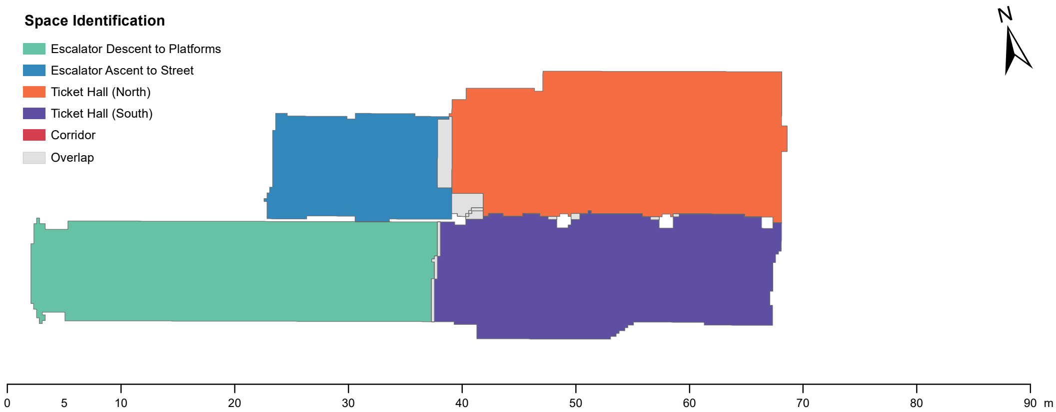




C.3 BTH-MANIFOLD Space Set

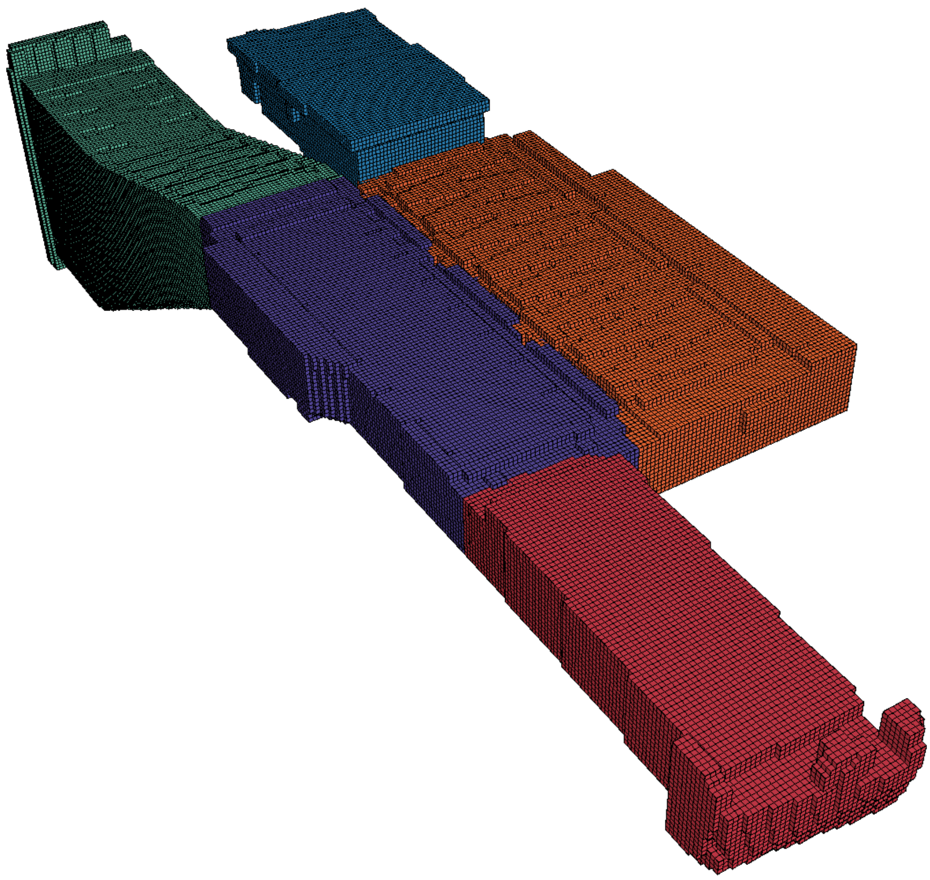
- Broadgate Ticket Hall
- Voxel size 0.25 m
- Discretisation Factor 0.5
- Manually merged into 5 spaces

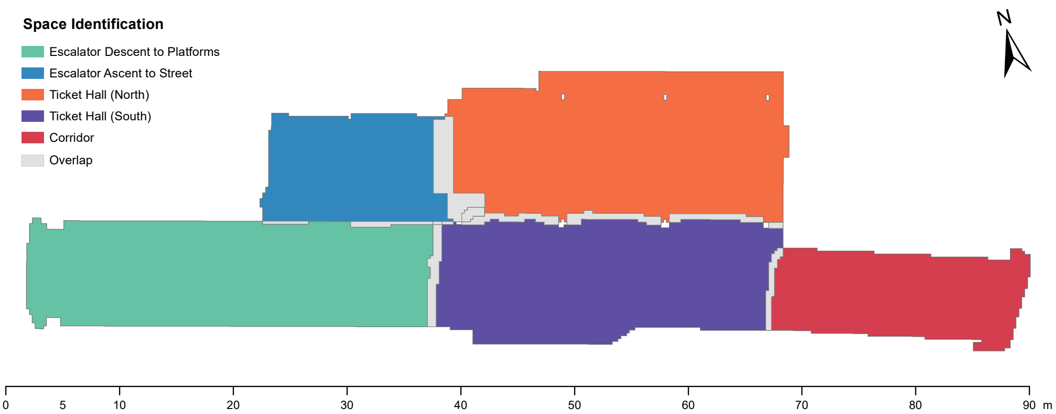




C.4 BTH-DILATED Space Set

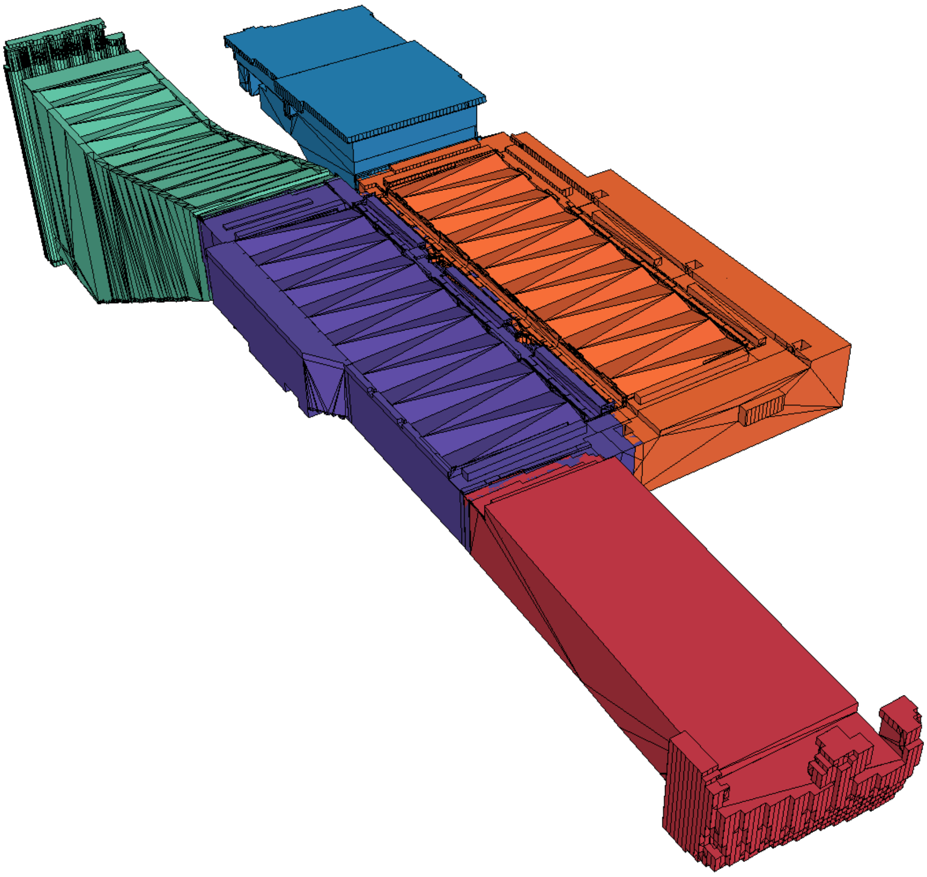
- Broadgate Ticket Hall
- Dilated spaces

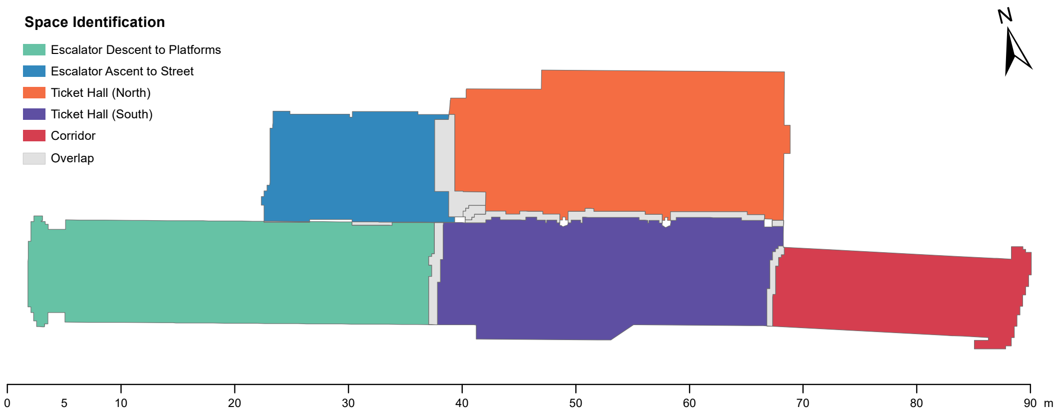




C.5 BTH-CLIPPED Space Set

- Broadgate Ticket Hall
- Clipped spaces

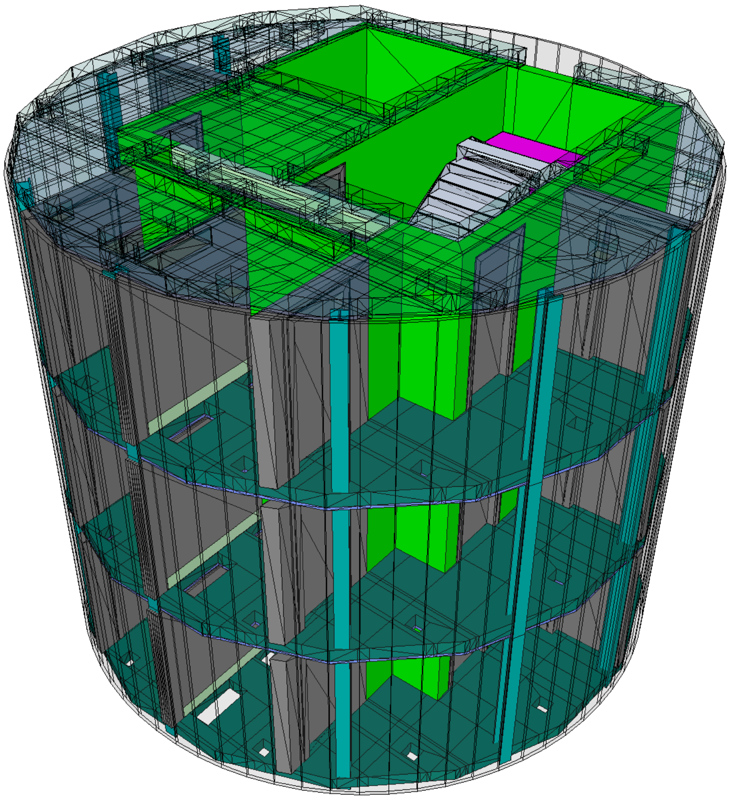




D Mile End Shaft Watershed Segmentation Output

List of voxelised output - Mile End Shaft

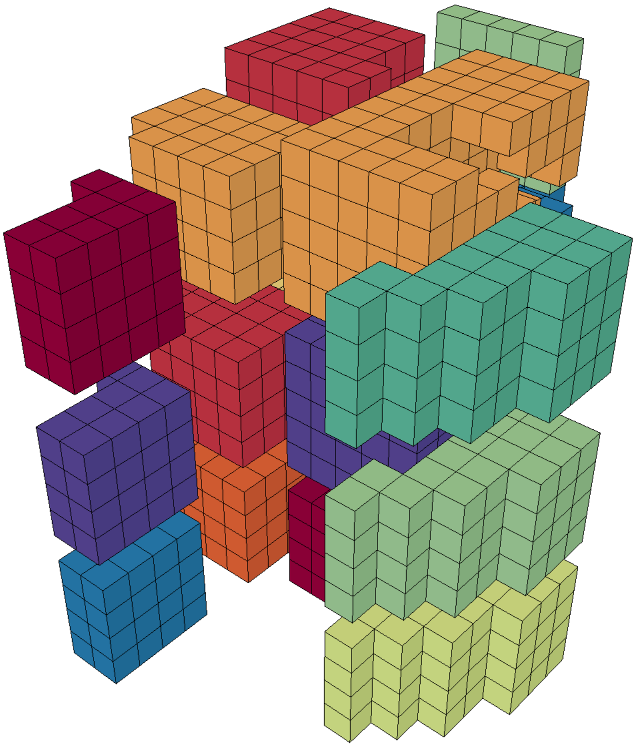
	Location	Size	Discretisation
D1	Mile End Shaft	1.0	2.0
D2	Mile End Shaft	0.5	2.0
D3	Mile End Shaft	0.25	2.0
D4	Mile End Shaft	1.0	1.0
D5	Mile End Shaft	0.5	1.0
D6	Mile End Shaft	0.25	1.0
D7	Mile End Shaft	1.0	0.5
D8	Mile End Shaft	0.5	0.5
D9	Mile End Shaft	0.25	0.5

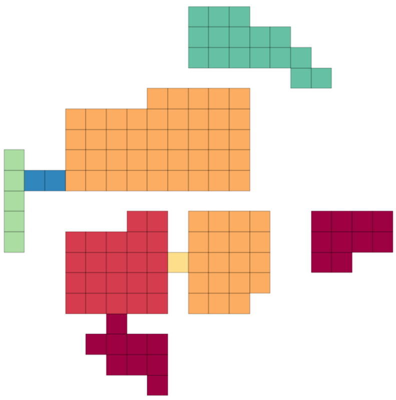


Mile End Shaft building elements

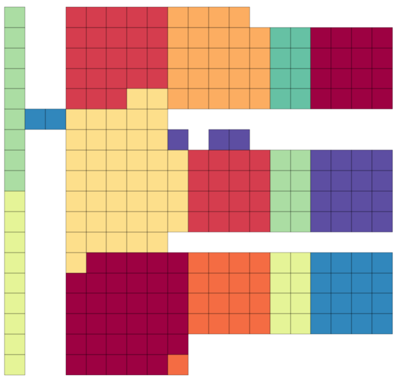
D.1 Segmentation using 0.5 m, DF = 1.0

- Mile End Shaft
- Voxel size 0.5 m
- Discretisation Factor 1.0





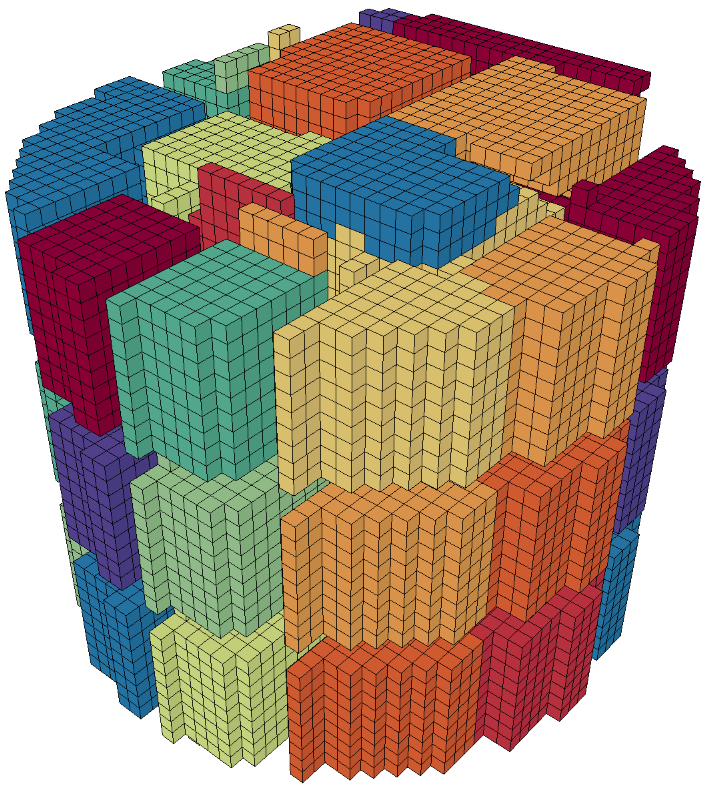
Plan

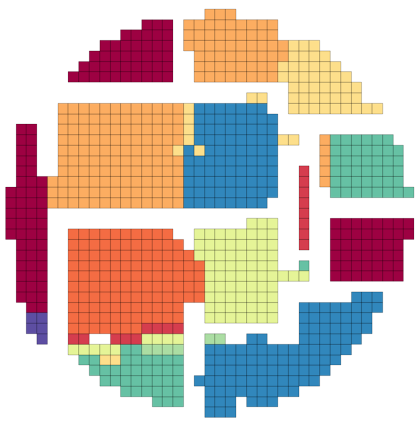


Side Elevation

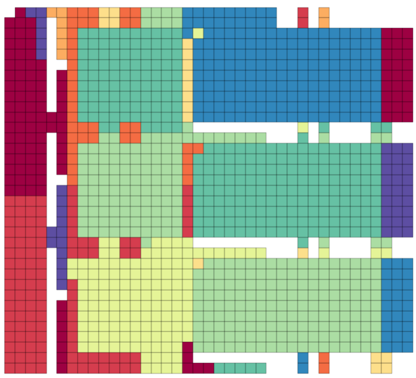
D.2 Segmentation using 0.25 m, DF = 1.0

- Mile End Shaft
- Voxel size 0.25 m
- Discretisation Factor 1.0





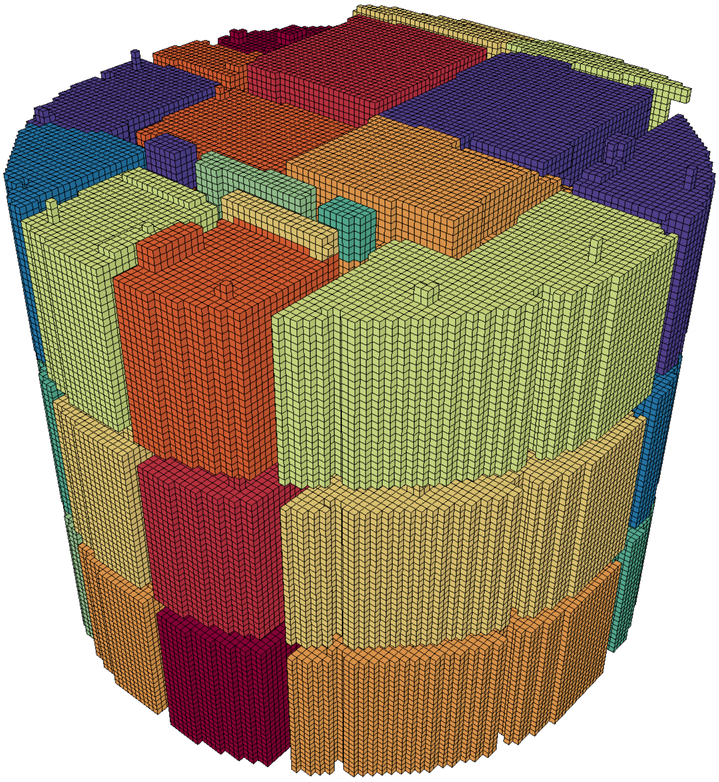
Plan



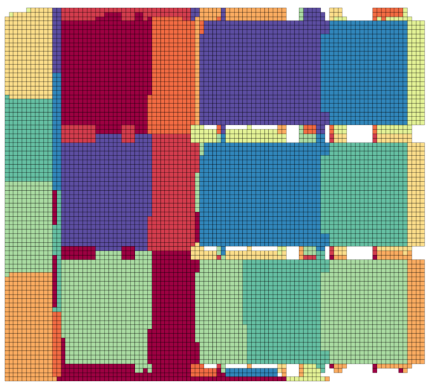
Side Elevation

D.3 Segmentation using 0.1 m, DF = 1.0

- Mile End Shaft
- Voxel size 0.1 m
- Discretisation Factor 1.0



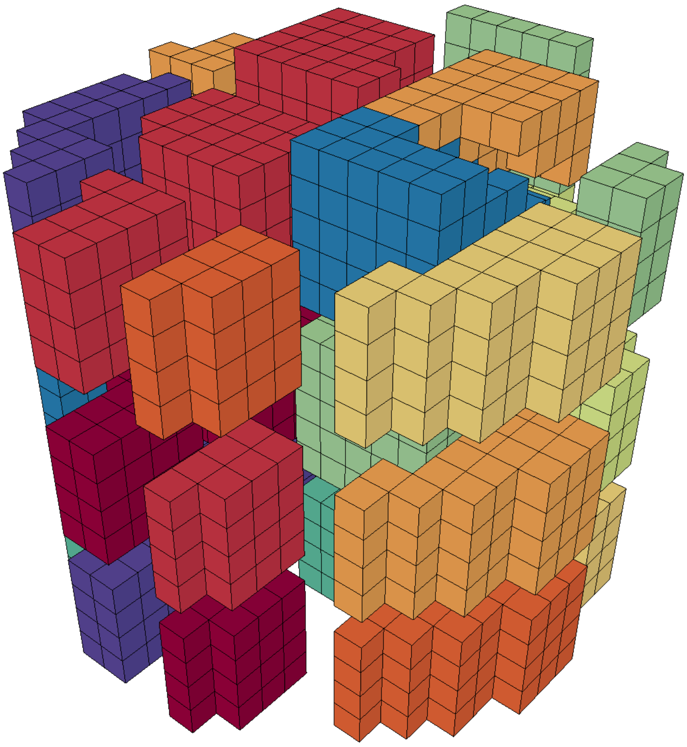
Plan

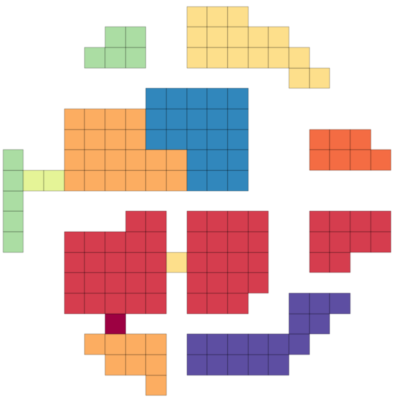


Side Elevation

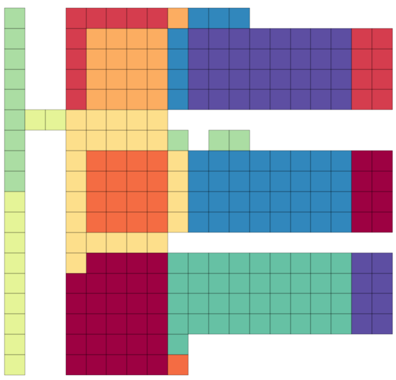
D.4 Segmentation using 0.5 m, DF = 0.5

- Mile End Shaft
- Voxel size 0.5 m
- Discretisation Factor 0.5





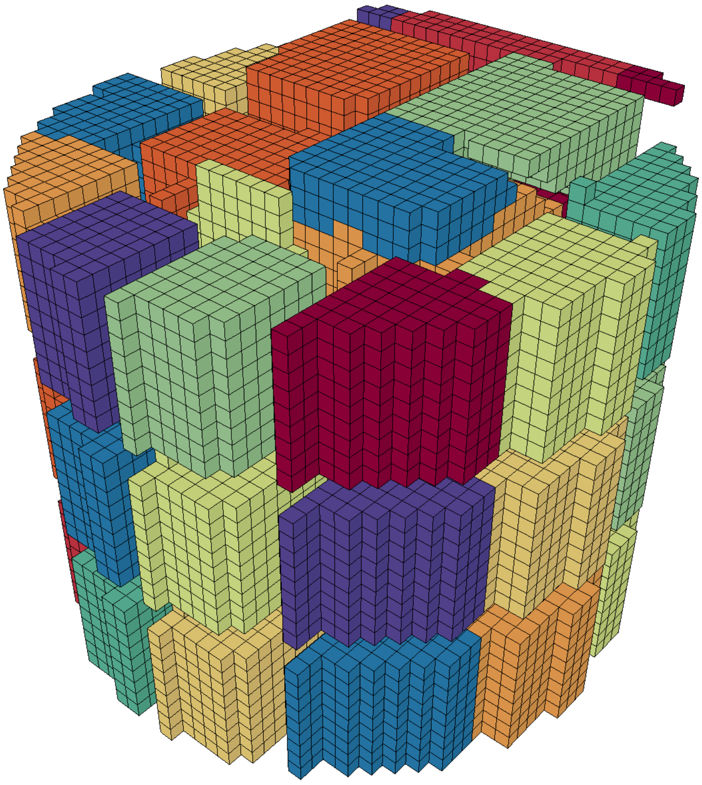
Plan

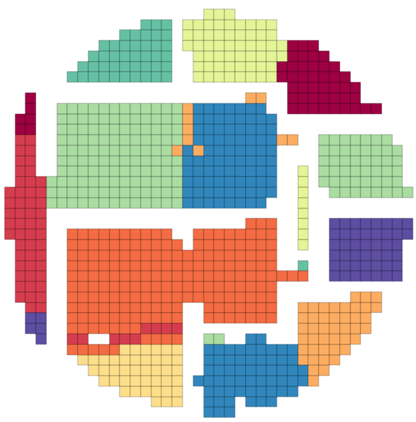


Side Elevation

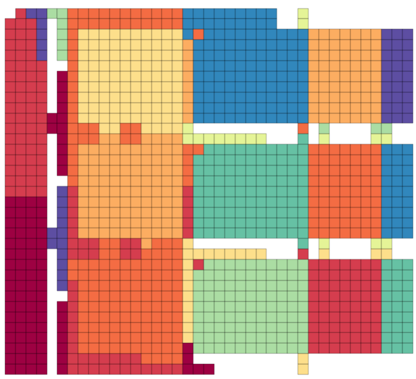
D.5 Segmentation using 0.25 m, DF = 0.5

- Mile End Shaft
- Voxel size 0.25 m
- Discretisation Factor 0.5





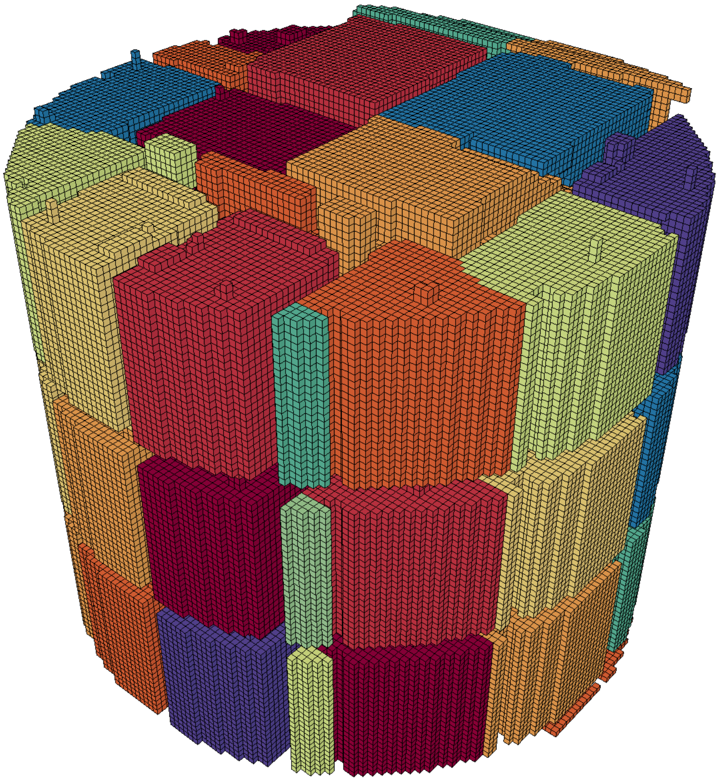
Plan



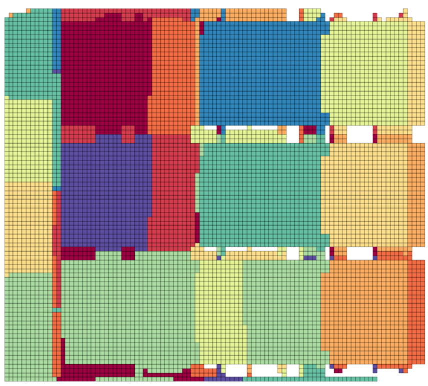
Side Elevation

D.6 Segmentation using 0.1 m, DF = 0.5

- Mile End Shaft
- Voxel size 0.1 m
- Discretisation Factor 0.5



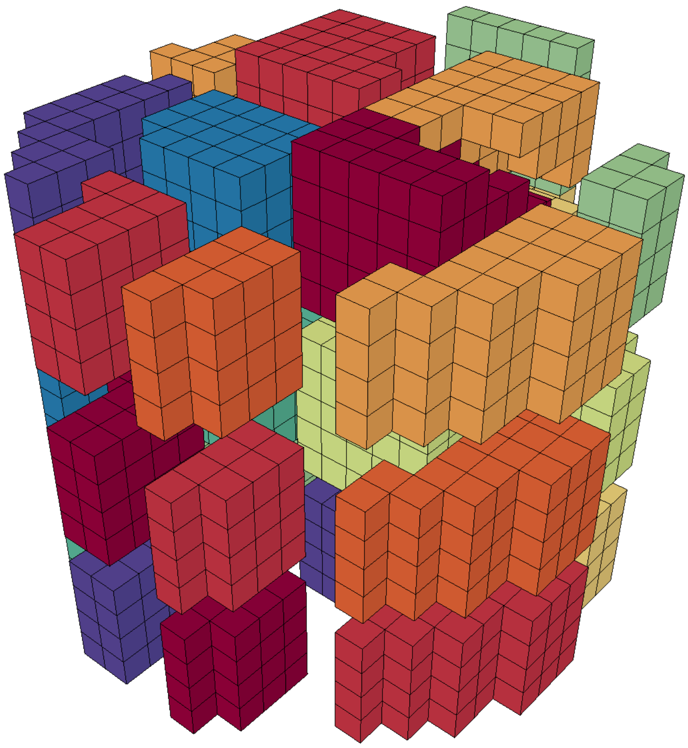
Plan

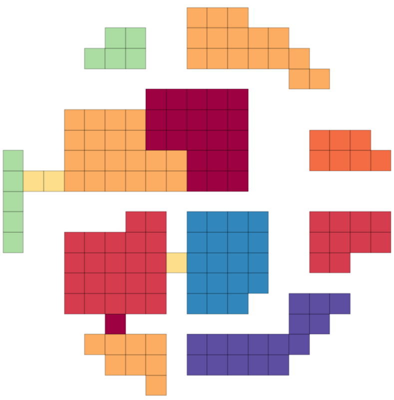


Side Elevation

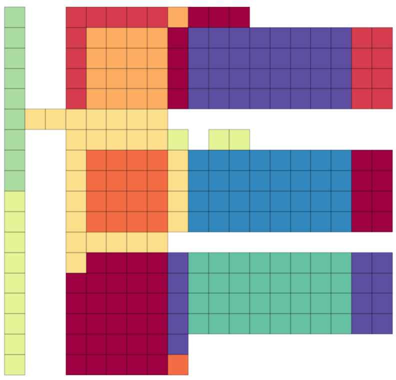
D.7 Segmentation using 0.5 m, DF = 0.25

- Mile End Shaft
- Voxel size 0.5 m
- Discretisation Factor 0.25





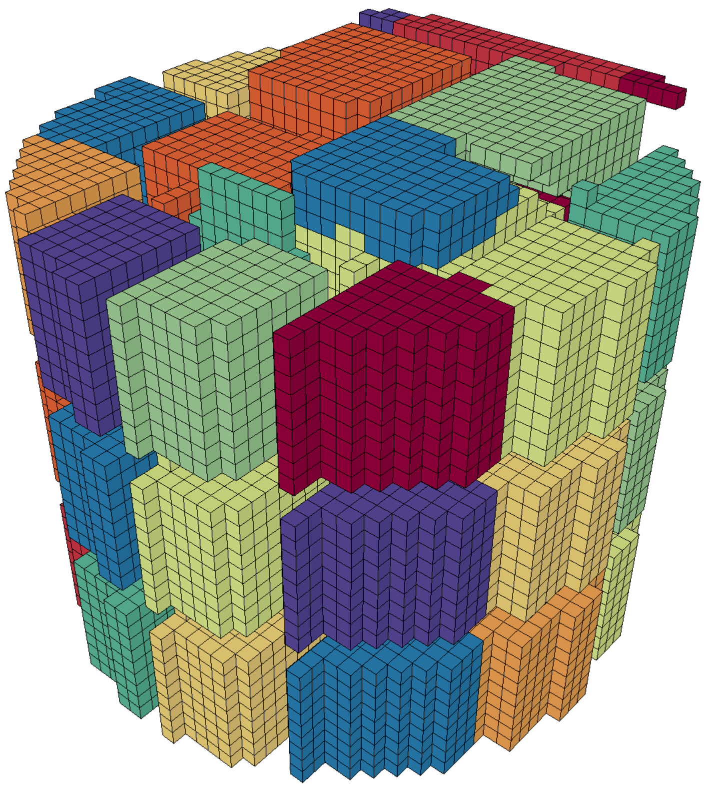
Plan

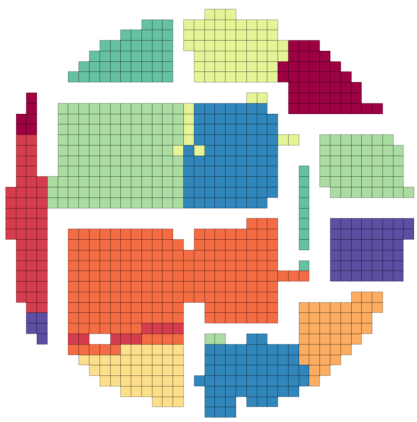


Side Elevation

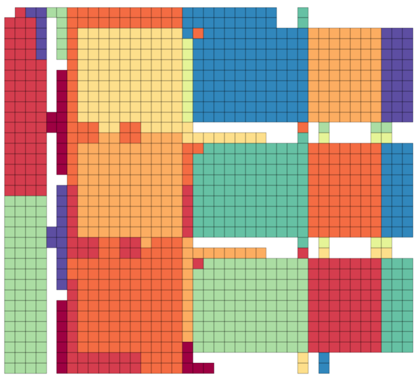
D.8 Segmentation using 0.25 m, DF = 0.25

- Mile End Shaft
- Voxel size 0.25 m
- Discretisation Factor 0.25





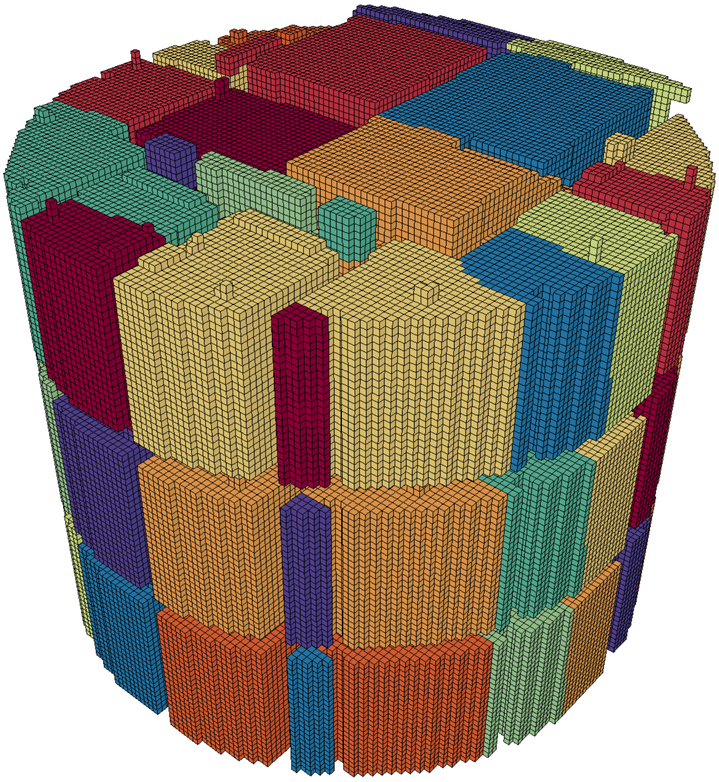
Plan



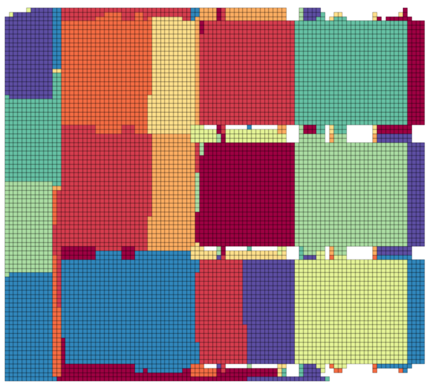
Side Elevation

D.9 Segmentation using 0.1 m, DF = 0.25

- Mile End Shaft
- Voxel size 0.1 m
- Discretisation Factor 0.25



Plan



Side Elevation

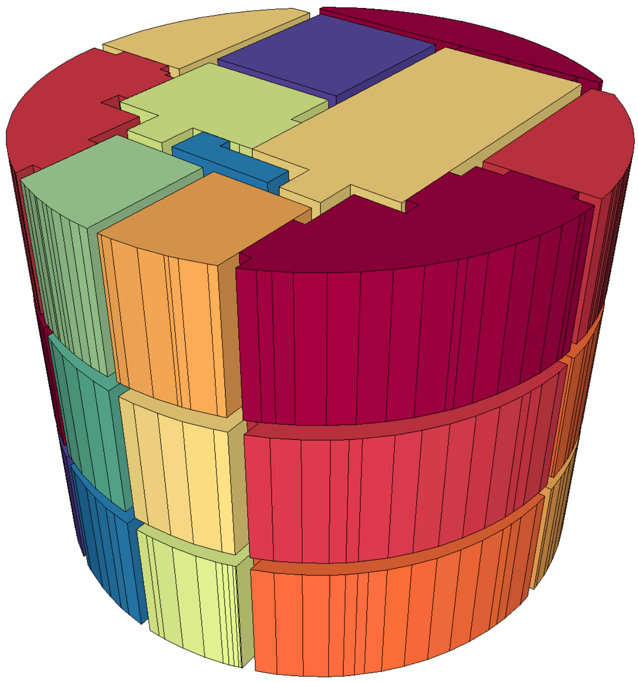
E Mile End Shaft Spaces Prepared for Spatial Query

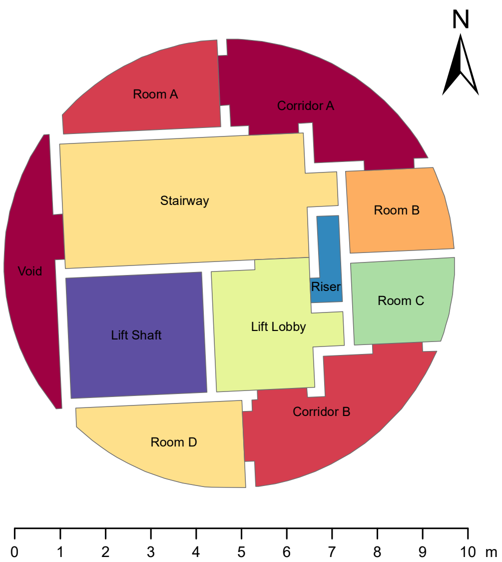
List of space sets

Appendix	Space Set	Description
E1	MES-EXTRUDED	Extruded Floor Plan 38 spaces extruded
	MES-EXTRUDED-11	11 spaces on Level 4 selected for Spatial Join
E2	MES-MANIFOLD	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 25 principal spaces Converted to Manifold B-Rep
	MES-EXTRUDED-11	11 spaces on Level 4 selected for Spatial Join
E3	MES-DILATED	Watershed Segmentation using Voxel Size 0.25 m / 0.5 DF) Over-segmented spaces merged into 25 principal spaces Dilated by one voxel (0.25 m) Converted to Manifold B-Rep
	MES-DILATED-11	11 spaces on Level 4 selected for Spatial Join

E.1 MES-EXTRUDED Space Set

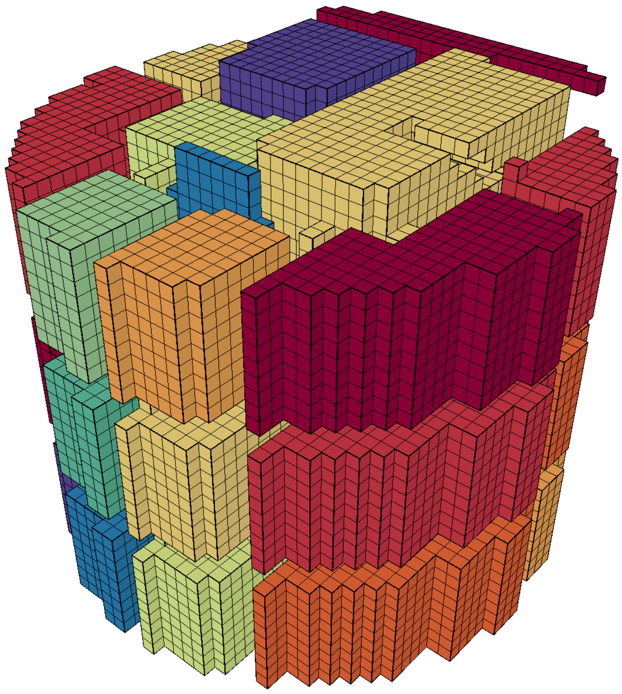
- Mile End Shaft
- Extruded floor plan

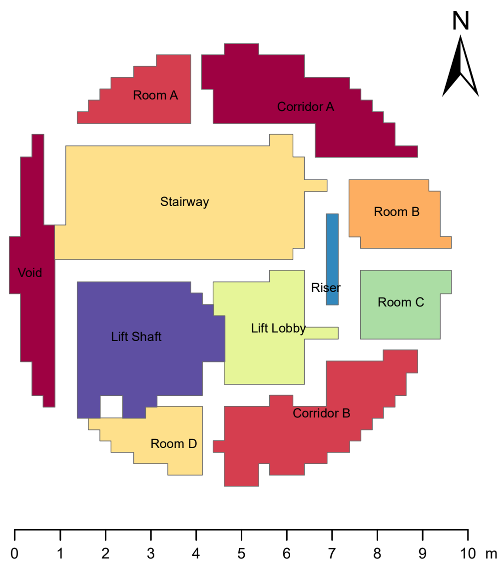




E.2 MES-MANIFOLD Space Set

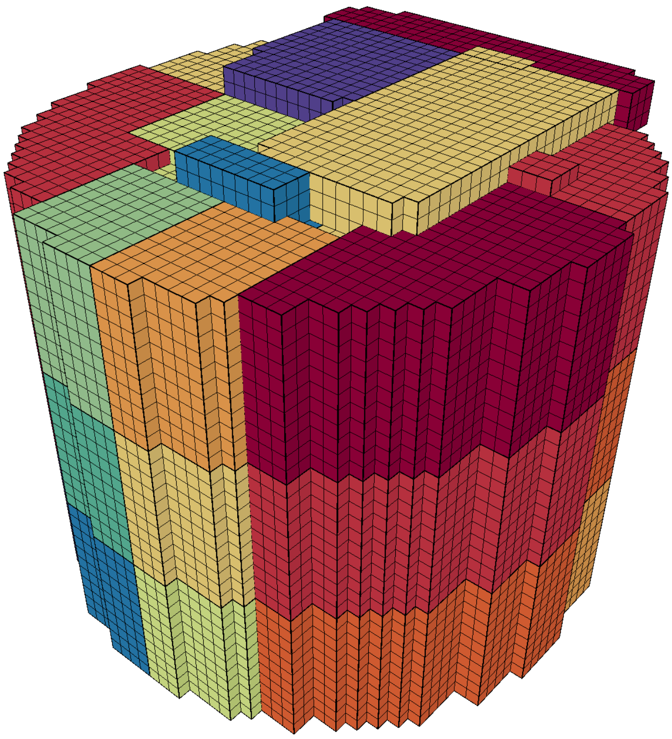
- Mile End Shaft
- Voxel size 0.25 m
- Discretisation Factor 0.5
- Manually merged

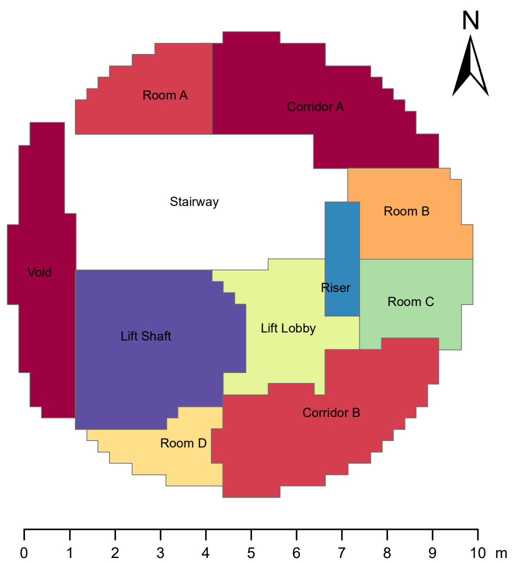




E.3 MES-DILATED Space Set

- Mile End Shaft
- Voxel size 0.25 m
- Discretisation Factor 0.5
- Dilated spaces





F Processor Specifications

F.1 Macbook

- Processor: Intel Core i7
- Processor Speed: 2.8 GHz
- Processor Cores: 2
- Memory (RAM): 16 GB
- GPU: Intel Iris 15 GB
- VRAM: 1.5 GB
- Principal Drive: SSD

F.2 Windows PC

- Processor: Intel Xeon E3
- Processor Speed: 3.3 GHz
- Processor Cores: 4
- Memory: 64 GB
- GPU: GeForce GTX 1060
- Adapter RAM: 6 GB
- Principal Drive: SSD

G Catalogue of Attachments

The following files are saved in a storage medium attached to this thesis.

All code is the original work of Gareth Boyes except that contain in the *pythonocc_display_fork* directory.

The code in the *pythonocc_display_fork* directory is a fork of github repository maintained by Thomas Paviot. Additional code in this fork has been commented with Gareth Boyes's moniker *GXB*.

All original code is Copyright Gareth Boyes, 2021 and licensed in accordance with UCL examination regulations.

G.1 MicroStation Batch Process Command file

```
+ batch/
    • Batch_Process.bprc.txt
```

G.2 MicroStation Visual Basic Scripts

```
+ mvba/
    • MVBA_Script.mvba.txt
```

G.3 Python Implementation of Watershed Segmentation

```
+ watershed_segmentation/
   + py3dgeom/
       • Test_Py3DGeom.py
       • TestData_Py3DGeom.py
      + src/
          • P3DGeom_setup.py
          • topology_py.py
          • Py3DGeom_py.py
   + watershed/
       • logging.json
       • main_program.py
```

```
    • test_program.py
  + settings/
      • scene_set.py
      • load_settings.py
      • mac_settings.py
      • __init__.py
      • progress_set.py
      • display_set.py
      • grid_set.py
      • setting_combinations.py
  + neighbour_skimage/
      • logging.json
      • neighbour_ski_ig_py.py
      • neighbour_setup.py
  + utils/
      • cython_setup.py
      • logging.json
      • logging_env.py
      • __init__.py
  + manifold/
      • check_manifold_setup.py
      • check_manifold_py.py
  + voxel_creator_/
      • voxel_creator_setup.py
      • voxel_creator_py.py
  + modules/
      • class_progress.py
      • display_q_voxels.py
      • class_group.py
      • form_solid_from_voxels.py
      • display_room.py
      • class_grid.py
      • grow_barrier_array.py
      • filing.py
      • thirtytwo_bit.py
      • __init__.py
      • hull.py
      • scene.py
      • display.py
      • analyse_settings.py
      • triangulate.py
      • two_rooms.py
      • adjacency.py
      • palgen.py
      • class_scene.py
      • add_mat.py
      • disp_illust_modules.py
+ pythonocc_display_fork/
    • qtDisplay.py
    • backend.py
    • SimpleGui.py
    • __init__.py
    • wxDisplay.py
    • LegendGui.py
    • OCCViewer.py
  + WebGl/
      • __init__.py
      • threejs_renderer.py
      • simple_server.py
      • x3dom_renderer.py
      • jupyter_renderer.py
  + icons/
      • cursor-pan.png
      • cursor-magnify.png
      • cursor-magnify-area.png
      • cursor-rotate.png
```

H Reproduction of MEP Features Joined with by Space Sets

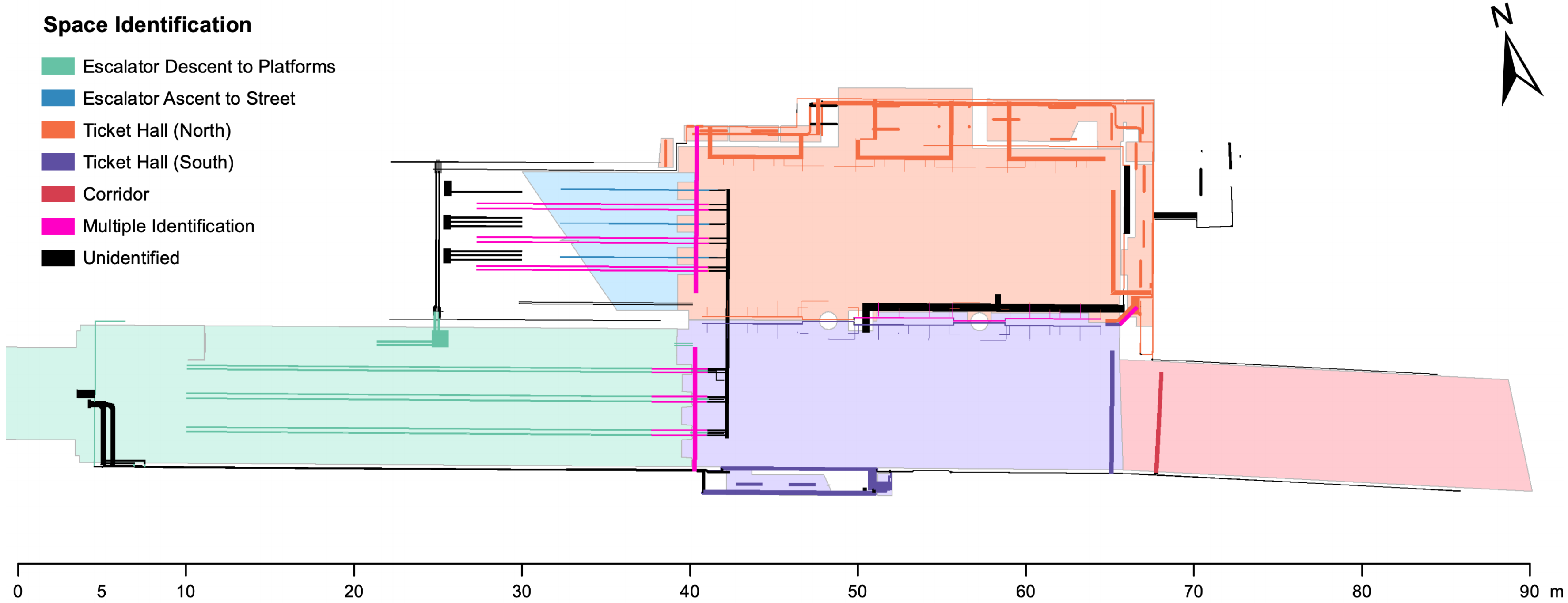


Figure H.1 MEP Features joined with BTH-EXTRUDED space set

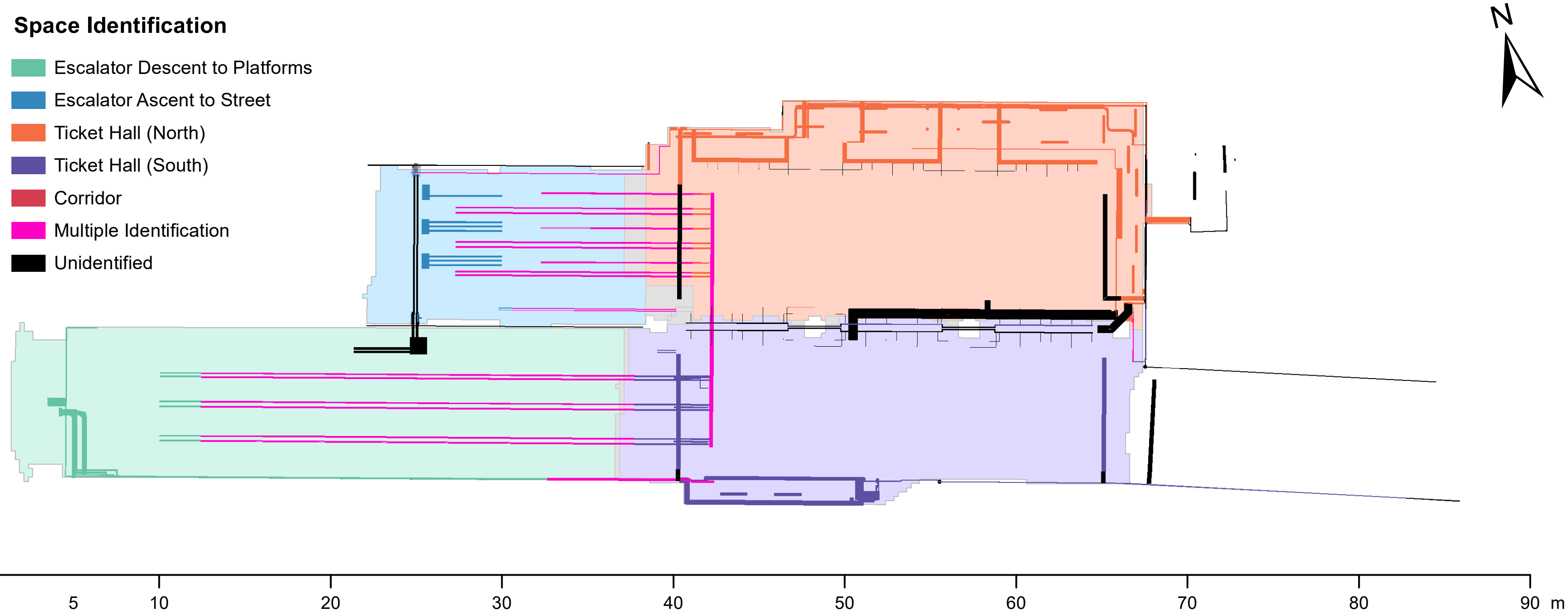


Figure H.2 MEP Features joined with BTH-MANIFOLD space set

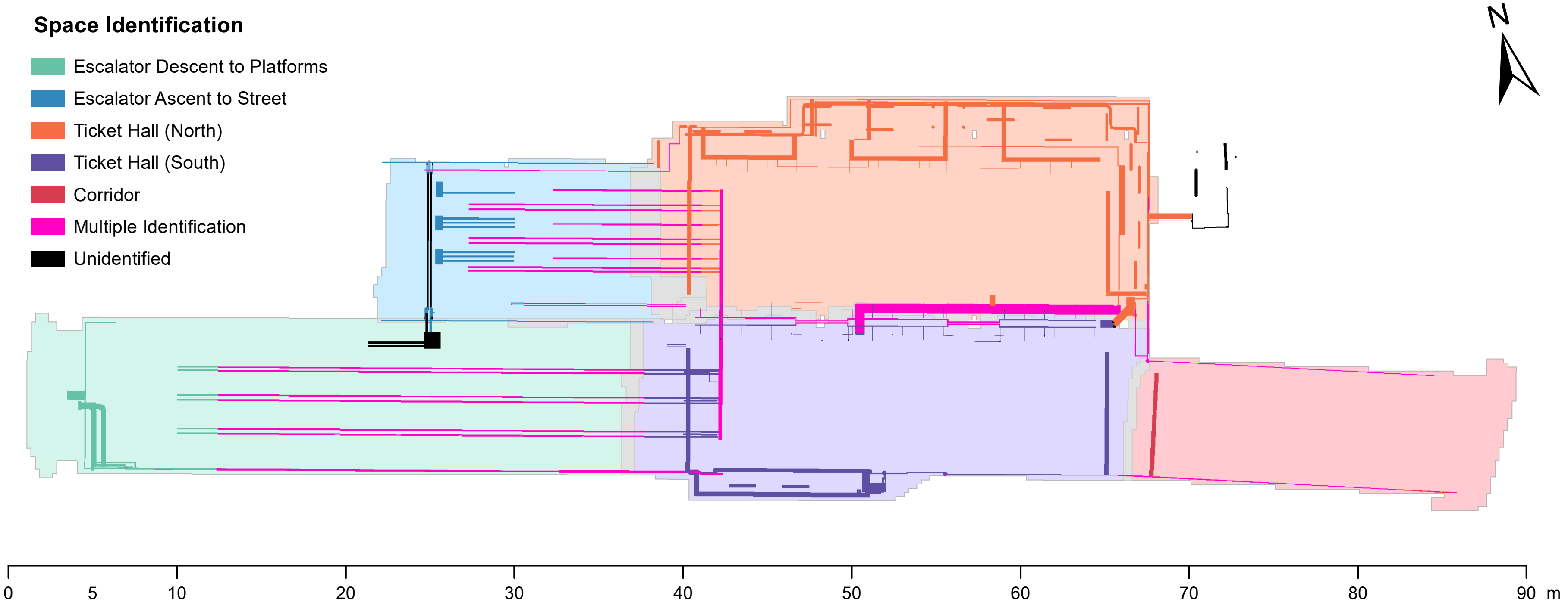


Figure H.3 MEP Features joined with BTH-DILATED space set

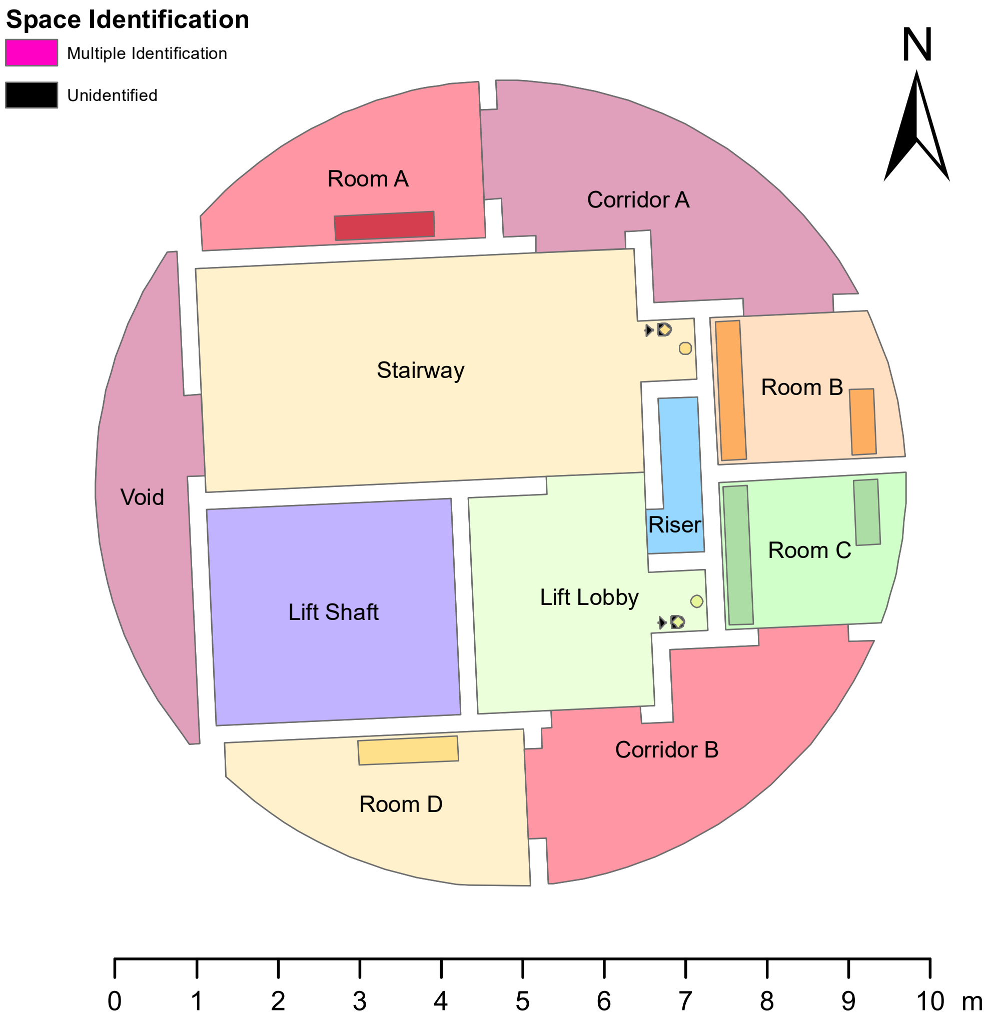


Figure H.1: MEP features joined with MES-EXTRUDED space set

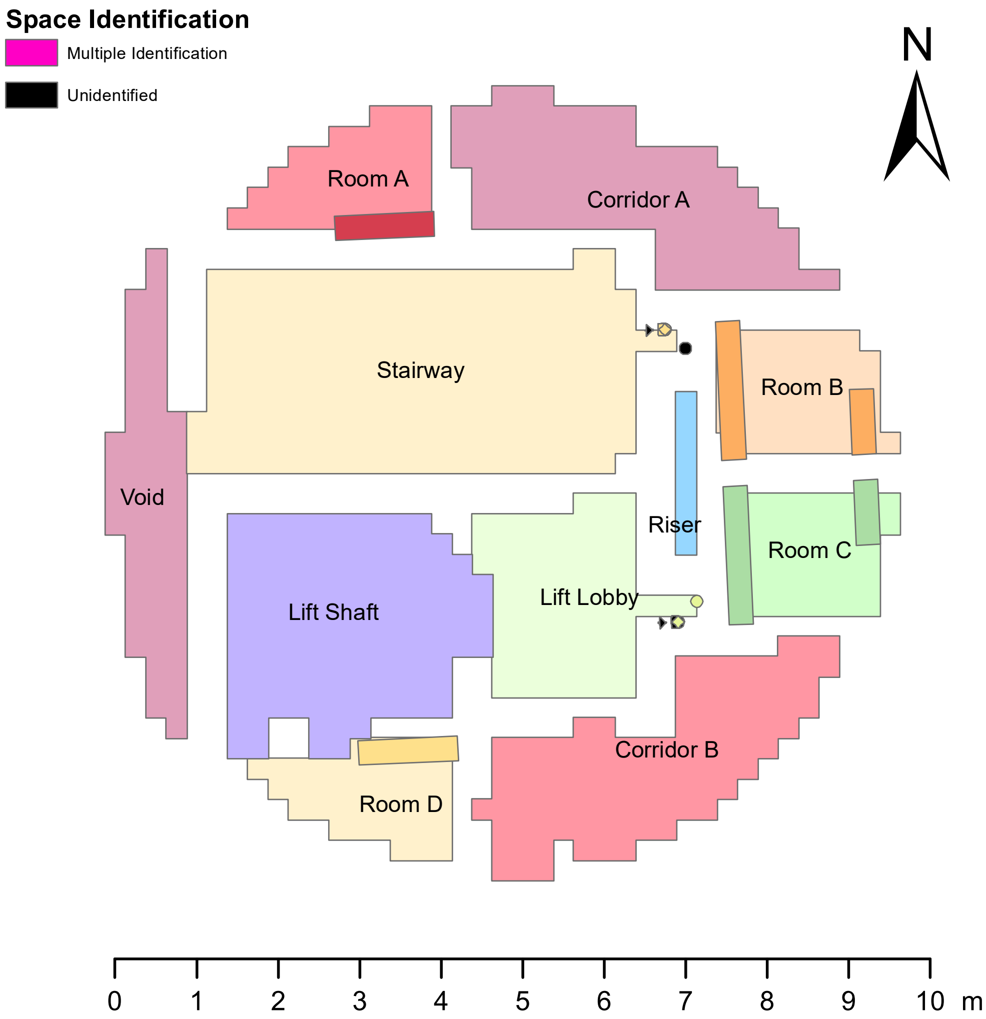


Figure H.2: MEP features joined with MES-MANIFOLD space set

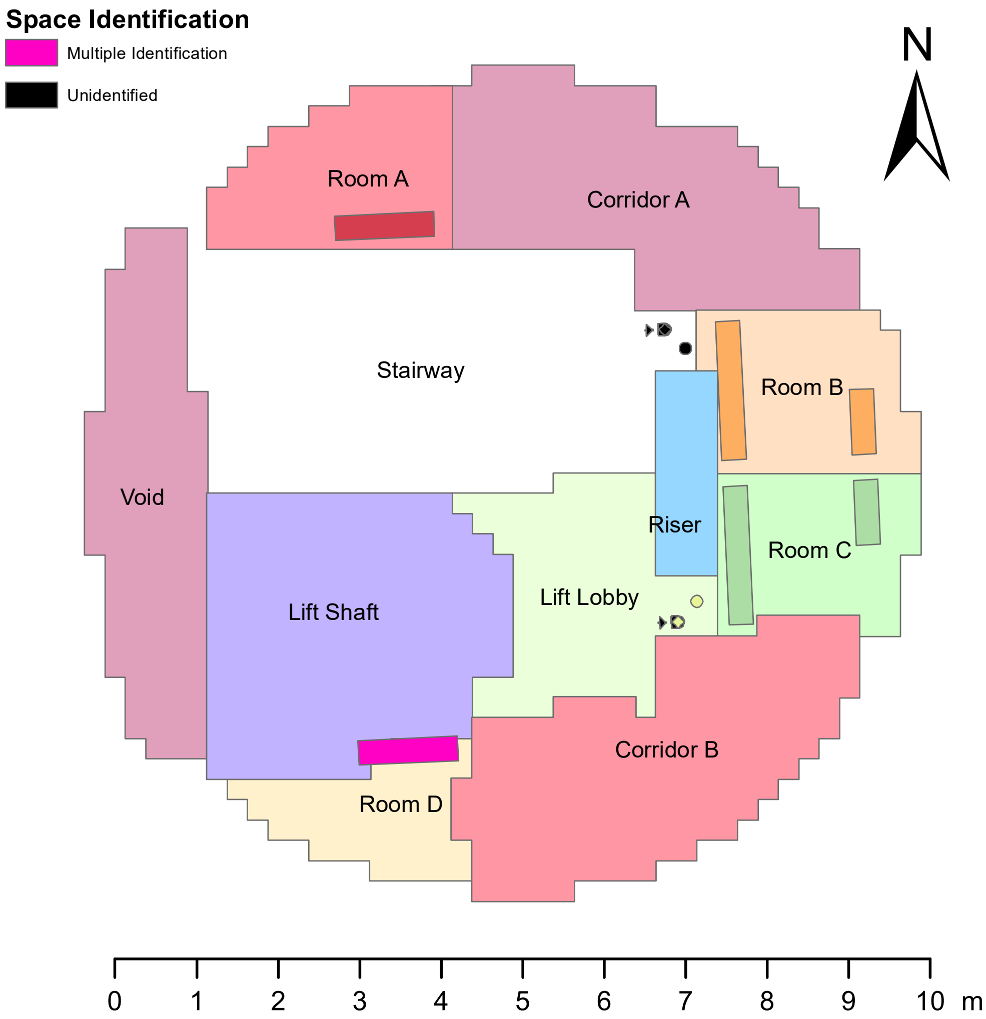


Figure H.3: MEP features joined with MES-DILATED space set

[FINAL PAGE]