Building Information Modelling to Discrete Event Simulation integration towards the development of a space-process Digital Twin

Nicola Moretti[a,*], Yin-Chi Chan[b], Momoko Nakaoka[b], Jorge Merino[b], Anandarup Mukherjee[b], Ajith Kumar Parlikad[b]

[a] *Bartlett School of Sustainable Construction, University College London WC1E 6BT, United Kingdom*
[b] *Institute for Manufacturing, 17 Charles Babbage Road, Cambridge CB3 0FS, United Kingdom*

Abstract

Building Information Modelling (BIM) offers an incredibly rich set of information that can be used across the whole building's life cycle. Though research and applications have largely focused on Facilities and Asset Management, there is still little understanding on how BIM data can have an impact on core business operations. In this article, BIM and Discrete Event Simulation techniques are integrated in a space-process Digital Twin with the goal to determine and control the impact of the spatial layout and built asset performance on the core process throughput. A use case on a multi-storey histopathology laboratory has been developed and the performance loss of the core process due to faulty lifts has been studied. The application of the proposed approach gives the Operations and Facilities Managers evidence of significant increase of the laboratory Turnaround Time (approx. 9.5% on average) when samples are not transported through the optimal route.

Keywords: Building Information Modelling, Discrete Event Simulation,, Digital Twin, Process Digital Twin, Industry Foundation Classes, Healthcare facilities, Histopathology Laboratory, Operations Management, Facilities Management

---

1. Introduction

The digitalisation of the construction sector offers a variety of digital technologies that can be used for data collection, modelling, service development and system automation. The recent advancements in the Digital Twin (DT) research have demonstrated how these technologies can be used to digitise assets and processes, simulate and predict performance under various conditions, and automate their operation [1]. A DT can be broadly defined as cyber-physical system that links a computational representation of a physical asset, entity or process with a two-way flow of real-time data from a physical twin. This powerful link between the physical and the digital can help monitor, optimise, and remotely control the physical asset or process across its life cycle [2, 3].

In the past few years, the DT concept has gained momentum in the built environment sector [4], with national programmes such as the UK National DT programme [5], now conveyed into the National Cyber-Physical infrastructure Ecosystem Programme [6] and Smart Nation Singapore [7]. The DT concept and approaches have been applied to a variety of disciplines in the past few years including aerospace, manufacturing, healthcare, and infrastructure. A large part of the research within the field has focused on the digitalisation, update, and curation of digital models of physical assets, e.g., through remote sensing technologies, computer vision, and point cloud segmentation. Whereas research efforts have mostly focused on developing high-fidelity digital models of physical objects (equipment, engines, building elements, roads, etc.), significantly less attention has been given to the development and use of DT approaches in process modelling, simulation, and operation. In particular, little has been done towards the ideation, development and testing of interdisciplinary approaches, which combine physical and geometric digital models, performance models, and process flow models into an integrated DT. The DT concept in the built environment sector is often derived from Building Information Modelling (BIM) research [8], which is widely considered the main information management framework in this sector [9]. BIM data and information management techniques are broadly used in the Architecture, Engineering, Construction, and Operations (AECO) sector and form a rich source of built asset data, which can be utilised in the buildings' use phase for improved operations, maintenance, and space management [10]. The BIM techniques have been explored in the field of Facilities Management (FM), e.g., through Construction to Operations building information exchange (COBie) [11, 12] and the UK's BIM level 2 [13] (PAS 1192, now superseded by ISO 19650 [9]), demonstrating the benefits in maintenance scheduling, environmental monitoring, energy management, and other applications [14].

Despite this, the use of BIM in industry is still largely shifted towards the design and construction phase and this is in contrast with the FM sector accounting for a large part of a building's lifecycle. From 2020, facilities managers purchased about €450 billion globally and manage approx. 7.2 billion square metres of building space [15]. In 2018, FM professionals published the ISO 41000 series and started connecting it to ISO 55000 on Asset Management and ISO 19650 on BIM. However, as the industry became more aware of the benefits of BIM in FM,

---

[*] Corresponding author
Email address: `n.moretti@ucl.ac.uk` (Nicola Moretti)

the issue of interoperability became clear as there is no widely adopted protocol for seamless data exchange from the design and construction phase to the operation phase of a building's lifecycle, nor use of Common Data Environment and Information Delivery Standards, despite the Industry Foundation Classes (IFC) is an enabler for this. On the other hand, it is agreed among the industry that value can be delivered when DTs are updated and shared among stakeholders and across different building stages. Due to the lack of interoperability, the potential value of BIM in facilities operations and location-based process simulation remains largely unexplored [15].

In this paper, we investigate the benefit of openBIM in building operation (i.e., the use of open standards and data in BIM processes, such as IFC), with a focus on location-based data integration for process simulation in the building's use phase. We propose a BIM-Discrete Event Simulation (DES) data integration approach, which enables the use of building geometric and topological data for the process simulation of a laboratory facility in operation. This research facilitates the more efficient and interoperable use of design and construction BIM information, which traditionally does not deliver enough benefits in the building use phase and core services delivery. The BIM-DES integration approach provides advancements in the research field regarding the conceptualisation and development of DTs of spatial elements and processes, in which a number of challenges are identified and described below.

1.1. Asset lifecycle challenge

When a BIM model is generated, the Level of Information Need (LOIN) is seldom specified with a focus on operations. BIM models are instead usually developed for Architectural and Construction (A&C) purposes, thus aiming for high efficiency of the design and construction process. On the other hand, in FM and operations, the location and general topology of the assets is often more important than their exact shape and geometric detail. Therefore, when BIM information is accessed programmatically, the unnecessarily high geometric complexity of A&C BIM data results in heavy computational requirements that can limit the capacity of data-driven FM applications and facilities operation. Therefore, there is a need to define the information requirements for FM (broadly including maintenance and operations services), with a focus on efficient BIM data extraction in an open format. This will allow for the automatic construction of light weighted BIM models targeting the less geometrically complex information requirements of the building's use phase.

1.2. Interoperability challenge

BIM data is processed in the use phase, e.g. through the development of Model View Definitions[1] and a set of Extract-Transform-Load (ETL) processes are implemented. The less automated and more complex the pipeline is, the more difficult it becomes to update the source information. Therefore, there is a risk of making the BIM model update process very cumbersome if not almost impossible. In the definition of the implementations of the information requirements for BIM-based building operation, it is more efficient to consider methods that automate the conversion of BIM data, so that the model update capabilities are preserved or at least still possible.

1.3. Decision-making challenge for improved asset operation and facilities management

There is a wide range of literature on the use of BIM for Virtual Design and Construction (VDC). This is usually referred to as BIM *n*D, where "D" stands for the "dimensions of the BIM data, typically extending beyond pure spatial information [16]. However, the problem of how to use BIM data for asset performance simulation in operation, beyond the native BIM software environment, remains little explored. Whereas much research exists to address the problem from a design and construction management point of view, few publications consider the operation of the buildings in the use phase. Consequently, there is little control on how the building and its parts impact on the core processes hosted in the building, e.g., delivering education, healthcare, or hospitality services.

1.4. Research questions

Within the context defined above, we consider the following research questions:

1. **Data transfer:** What are the IFC requirements for core business processes modelling and simulation in specialised buildings (e.g., hospitals)?
2. **DT-based decision-making:** How can an integrated BIM-Discrete Event Simulation model be used to impact upon the decision-making of the core business process?
3. **DT curation:** How to make sure the data is always up-to-date and fit-for-purpose?

In this article, we will focus on Questions 1 and 2 and we propose a DT architecture partially addressing Question 3.

2. Previous research on integrated BIM, Process Modelling, and Simulations

Research on the integration of BIM with process Modelling and Simulation (M&S) techniques has mostly focused on the design and construction process. For example, in lean construction, the concept of 4D BIM simulation (4D = three spatial dimensions plus time) has encouraged the use of 3D building models and process simulation to visualise the construction process over time and improve the scheduling of construction activities [17]. Many examples of BIM and M&S integration can be found in the literature and have been classified according to application and process M&S techniques, in Fig. 1 and Tables 1 and 2. The reminder of this section presents a review of the key publications in these fields.

---

[1] A Model View Definition is a subset of the Industry Foundation Classes schema for a specific use case.

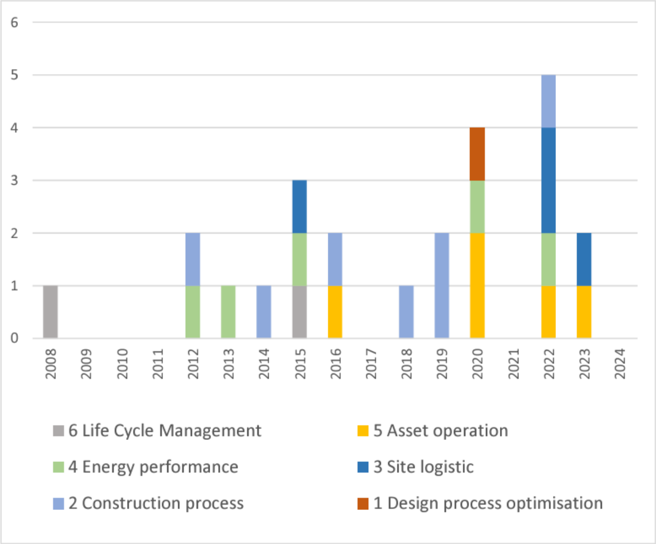

Figure 1: Key studies on BIM-process M&S integration and applications, by year

Table 1: Applications of BIM and process M&S integration

Application	**Papers**
Design authorising	[18]
Construction process	[19–26]
Site logistics	[27–30]
Energy performance	[31–36]
Asset operation	[37–39]
Lifecycle management	[40]

2.1. *Design and Construction*

In the design process optimisation context, [18] developed a method to capture digital footprints of project stakeholders and create event logs for design authoring in building projects, using the IFC schema and process mining analysis techniques. The result is the automation of a number of processes for design authorisation, but no production process of real architecture model nor human activity process of collaborative design has been considered. Virtual models of the construction processes are presented in a Cyber-Physical Systems (CPS) framework for construction in [25]. Image processing techniques (e.g., image matching, image stitching) and computer vision applications — e.g., Augmented Reality — AR, BIM viewer software - are combined to model and simulate the construction process. [19] demonstrated a concept of a proactive construction fault management in a case study of field inspection using a designed defect specific domain ontology, AR, and image-matching. [21] proposed a novel inspection process model by developing a BIM API and demonstrate the model in a tile work construction quality control case study. The approach is validated by an expert survey about its practicality and efficiency. To improve the architectural design process, [29] develop an image stitching algorithm based and a camouflage stitching algorithm which utilises BIM and other digital technologies to simulate the construction process of building decorations. [23] propose a BIM-DES integration framework for the construction works. They improve the DES development process through the automatic extraction of the relevant information on resources and activities from the BIM model. However, in these examples, there is still part of the information which needs to entered manually and proprietary software platforms are largely used.

Table 2: BIM and M&S techniques and integration approaches

Technique	**Papers**
Business process model	[18, 21, 24, 27, 30, 31, 37, 38]
Work Breakdown Structure (WBS) method	[27, 28, 30, 36]
Activity network model	[18, 23, 25]
Simulation, discrete event	[23, 27, 34, 35]
Simulation, energy	[31, 34, 35]
Simulation, fluid dynamics	[33, 39]
Simulation, multi-agent	[30]
Particle swarm optimisation	[27]
OpenBIM	[19, 22, 25, 31, 37]
Computer vision	[19, 29]

Furthermore, several approaches have been introduced to combine Site logistics of construction projects with BIM product models in 4D BIM simulation. [27] achieved an in-depth integration of BIM product models with work package information, process simulations, and optimisation algorithms. [30] developed a BIM and Multi-Agent System (MAS) combined method to model construction actors that have time–space occupancy and short-term construction processes at the component-level. [28] proposed an open BIM-based automated scheduling approach for a project with logical constraints between components or construction activities. BIM facilitates automated scheduling considering multiple construction processes and multiple components in the model. However, the approaches mentioned lack a method to reflect the changes of processes on the models.

2.2. *BIM and M&S for the use phase*

Aside from construction projects, there are studies of BIM and M&S integration for Energy performance analysis, Asset operation and Life Cycle Management. For Energy performance analysis, various approaches combine occupants' behaviour with building energy models. [37] demonstrated that data on the activity and behaviour of occupants can significantly improve the performance of building performance simulation tools. [33] displayed four modelling scenarios of high-resolution modelling to capture boiler dynamics, thermal performance, energy consumption and occupant behaviour. [34] applied an energy modelling approach to industrial environments, focusing on the production processes using the Balanced Manufacturing (BaMa) approach. [35] partially automated the building competent creation for a holistic DT modelling for industrial facilities, utilising Discrete Event System Specification (DEVS), Dynamic Energy System Simulation (DESS) and

Building Energy Modelling (BEM), BIM, visual programming and a semi-automated data acquisition workflow. These studies integrated occupant behaviour processes and industrial production processes with building energy models, but they only consider energy aspect of a building and do not consider other links between a facility and business processes such as the impact of asset failures on human activities process or production processes.

Although there have been a large number of studies on energy performance analysis, studies on BIM and process M&S for asset operation are limited. [38] applied case narratives and process models to FM operations of complex buildings and described the potential BIM use cases for routine maintenance and emergency reactions. [39] explored BIM application for fire evacuation planning through a pilot study in which BIM-based fire process simulation was used to understand the relationship between evacuation exits as well as evacuation time and behaviour. These studies showed the potential of BIM use in FM; however, more case studies are needed on FM decisions such as maintenance planning and scheduling or resource allocation and planning.

Other studies propose improved data integration methods between BIM and MS for Life-cycle management. [22] designed an interoperable tool model to characterise the information exchange required in the process of constructing the building and installing the tools/equipment in a semiconductor manufacturing facility, following current information and data standards. [24] developed specifications for a cloud-based BIM governance platform using Business Process Modelling Notation (BPMN) and Unified Modelling Language (UML) to investigate the requirements for BIM governance and demonstrated it in two case studies of BIM projects. [26] presented a systematic overview of how Internet of Things (IoT) can be used in the BIM life-cycle of complex structures. [25] presented a framework in which CPS are based on virtual models of construction processes, implemented via Petri Nets and connected to BIM models as well as hardware working in on-site production or assemblies. [31] applied gbXML schema developed by commercial software vendors and [28] utilised IFC to deliver the set of models. Thus, some studies have explored data integration methods between data models in life-cycle. However, the methods based on BIM modelling standards need further investigation.

Besides application specific use cases, [41] propose a software architecture for the integrate visualisation of BIM 3D models and the results of various DES models. The authors develop an API to incorporate the tool called "ARSLab DEVS web viewer" (`ARSLabDEVSwebviewer`) in the Autodesk Froge API. The user can visualise in 3D the results of the simulation. However, the approach is not completely based on open data and platform, hindering generalisation and further applications.

2.3. *Knowledge gaps identification*

To conclude this review section, [42] describe the works of the Simulation Task force of the Visualisation, Information Modelling, and Simulation (VIMS) Committee of the American Society of Civil Engineers (ASCE). In 2015, the authors identify the challenges in modular design, data integration, simulation model development and validation in civil engineering. Issues that appear to be still under discussion today. They identify the DEVS method as one of the most effective to overcome these issues and propose a multi-level framework formed of atomic models (the individual DEVS), High Level Architecture (HLA), and data exchange for multi-modal simulation supporting interoperability. Building on the works of these pioneering papers (though the terminology is slightly different), we developed our research which provides advancements in the definition of clear BIM-DES information requirements, the demonstration of the benefits of using BIM methods for core building process operations and the capabilities of openBIM standards in delivering enhanced interoperability.

In fact, the review of the literature highlights some research gaps which help refining the research questions. First, most studies focus on design and construction applications — including the Design process optimisation, Construction process and Site logistics — and Energy performance (see Fig. 1), while fewer explored possible cases scenarios relating to Asset operation and Life-cycle Management. Second, although some papers have presented a method for semi-automated information extraction from BIM models, there are still limitations in fully utilising the capabilities of BIM data in practical ways. Thirdly, some projects heavily depend on private software platforms [39][32], which make it difficult to apply the findings widely and to scale them up effectively. BIM modelling standards facilitate to carry additional metadata for different purposes, but gbXML mostly looses the semantic relation among elements[31]. For example, [31] and [42] mention limited specification of information requirements and formats and this hinders generalisation and further applications, revealing a challenge in standardisation and data interoperability.

3. Methodology

In this paper, we develop a BIM-to-DES integration approach which enables us to use streamlined BIM data to inform the simulation of the core-process operation hosted in specialised buildings, i.e., facilities where the core business function is closely related to the built assets, such as hospitals, laboratories and airports etc.. To define this approach, we used a mix of evidence-based research and empirical case-study-based research methodologies.

3.1. *Discrete Event Simulation*

DES is a key method for predicting the evolution of a model's state under a given set of conditions. In the case of this paper, the DES method is used to represent a healthcare process. DES relates to models in which the state variables change instantaneously at distinct points in time, known as events [43]. However, events may be used for other purposes as well with or without changing the state of the system. For example, in a decision-making process, a decision may be made to take no action for a given event. As state changes in discrete-time systems can only occur at these events, DES can jump from one event to the next without any relationship between the simulation's internal clock and the actual time required to run the simulation.

A key data structure in DES is the event list, a list of pending events sorted in chronological order. Events are generally added to the event list in one of the two following ways. In one case, a generator adds events (e.g., entity arrivals) to the event list at periodic or random intervals. A common and simple example of a generator is a Poisson arrival process, in which the times between arrival events are exponentially distributed. In the other case, events may spawn other events when triggered. For example, an event corresponding to the start of service on a given entity (i.e., the seizure of a system resource) may automatically spawn the matching end-of-service event.

The main function in a DES is therefore an event loop that: (i) finds and removes the next occurring event from the event list; (ii) performs any state changes or other tasks (e.g., statistics collection) associated with the event; and (iii) adds any events generated from the processed event to the event list, maintaining chronological order. In addition, many DES tools and software libraries contain additional constructs to assist in tracking a system's state.

In this paper, we used the Python library salabim [44] as a basis for our DES program. A benefit of salabim is the inclusion of built-in statistics collection for resources and other simulation components. To break the process logic into manageable code blocks, each step in the process is represented by an instance of `BaseProcess`, which includes the derived classes `Process`, `BatchingProcess`, `CollationProcess`, and `DeliveryProcess`, as shown in Fig. 2.

Each process instance defines an infinite loop:

- `Process` takes entities from its `in_queue` and launches the process defined by `fn` for each entity. To register a process, we make it a member function of the appropriate class using the following code:

  ```
  setattr(self.in_type, self.name(), fn)
  ```

  The process defined by `fn` is responsible for forwarding entities to the `in_queue` of the next process in the process chain (unless it is the last process in the chain).

- `BatchingProcess` takes `batch_size` entities from its `in_queue` and places a `Batch` entity in the `in_queue` of the next process in the process chain, as defined by the string `out_process`.

- `CollationProcess` takes entities from its `in_queue` and collates them according to their `parent` attribute. When all child entities of a parent entity are collated (as tracked by the specified counter), the parent entity is placed in the `in_queue` of the next process in the process chain, as defined by the string `out_process`.

- `DeliveryProcess` takes entities (possibly `Batch` entities) and delivers them to the specified `out_process`, using one of `resource` and requiring time as defined by `durations`. `Batch` entities are unbatched before being placed in the `in_queue` of the output process.

Note that some process classes shown in Fig. 2 are analogous to logic blocks available in some graphical trajectory-based DES software, e.g. Process and Batch in Arena. However, the advantage of our Python-based DES approach is easier integration with other components such as the BIM component described in the following subsection.

3.2. OpenBIM and Industry Foundation Classes

The use of IFC in construction is well-known — it is the main exchange standard in the BIM domain, which supports the generation, sharing, and query of spatial, system, and asset data. IFC is based on the EXPRESS data modelling language and is used to store a "snapshot" of the building's properties and features in a specific moment in time. In this paper, we use IFC4 ADD2 TC1 [45]. The goal is to test the usability of IFC as data input for the process simulation component of our BIM-DES methodology. Therefore, we use both the semantic and geometric information represented within the IFC schema.

On the other hand, DES has the capability to represent variability in system attributes such as task duration and some of these attributes are space-dependent, e.g. the time needed to reach room B in a building from room A or the waiting time in a lift to reach the desired floor. IFC data can be combined with the process logic to track where and when each event takes place and in what order. In particular, the location of any `IfcProduct` (e.g. a space, door, wall, stairwell, or lift) can be modelled in IFC and the schema can be parsed to extract all necessary geometric and semantic information required to feed the DES space-dependent variables. However, the geometric representation of the IFC classes derived from the `IfcProduct` instances can differ, thus requiring different data discovery mechanisms to retrieve the geometric and spatial representations of the BIM objects.

In this article, we use the spatial representation of doors and walls and internal partitions in a building to feed a DES model. As an example, consider an `IfcDoor`. This door are spatially contained within an `IfcBuildingStorey`, through the `IfcRelContainedInSpatialStructure` class which is in turn contained within an `IfcBuilding`; however, other hierarchies exist, e.g. involving the `IfcSpace` or `IfcSite` classes. The door may itself be represented using `IfcExtrudedAreaSolid`, `IfcBooleanClippingResult`, or `IfcAdvancedBrep` and each of these geometries are represented differently in the IFC data model. Therefore, a variety of methods are required to obtain the location of `IfcDoor` instances relative to the global coordinate system, making the manual parsing of the IFC schema very complicated. To solve this problem, in this article we use the ifcOpenShell Python library (https://ifcopenshell.org/), since our process simulation was also developed in Python. IfcOpenShell offers a set of methods to navigate the geometry definitions in an IFC model; this is shown in Listing 1.

The `get_level_name` function traverses the IFC object hierarchy to obtain a suitable human-readable name for each wall

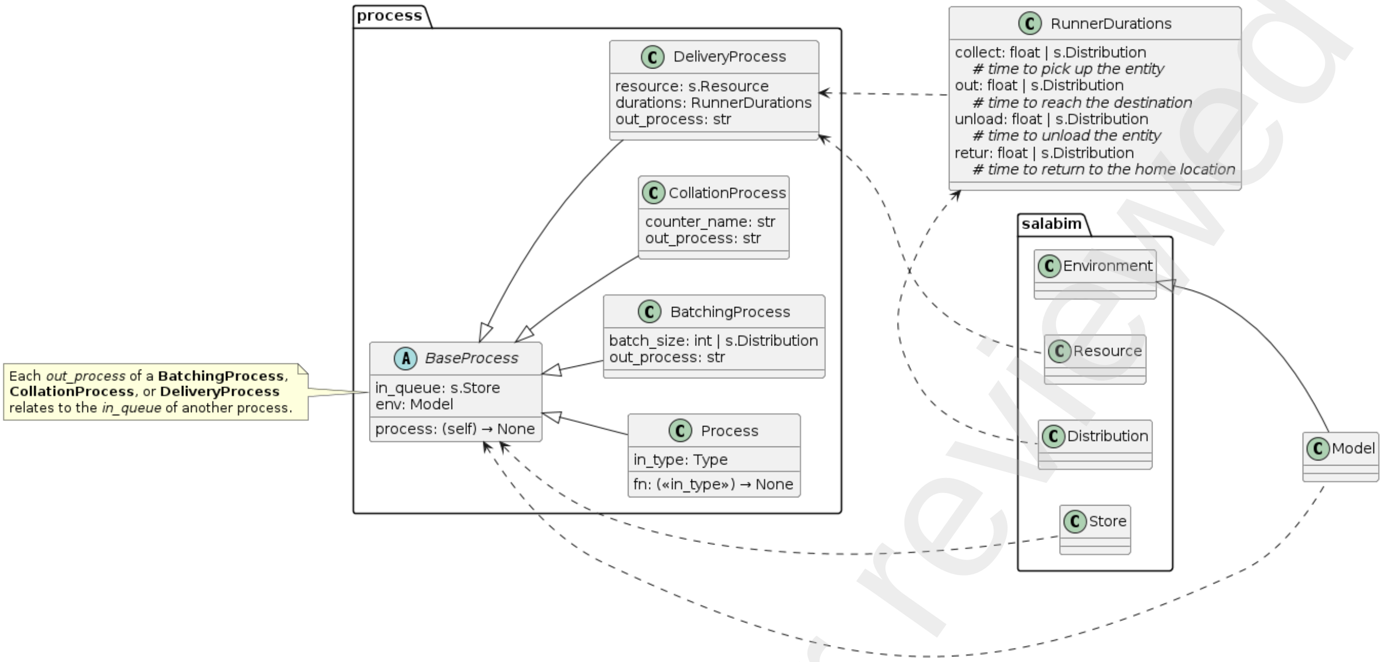

Figure 2: UML Class diagram for the `BaseProcess` and related classes in our DES framework ("s" is an alias for salabim). Lines with white arrowheads denote class inheritance while dashed lines denote a general association. Key attributes (middle box) and methods (bottom box) of selected classes are also shown.

or door. In particular, we have prepared the IFC model file such that all doors of interest have names of the form `d1`, `d2`, etc. Meanwhile, the `get_coords` function is used to obtain the bounding box of each wall or door. While the geometry definition of the IFC objects is much more complex than this, the current approach suffices for the model under consideration, in which all walls and doors are contained rectangular prisms aligned along the axes of the global coordinate system.

3.3. BIM-DES integration approach

The DES techniques are used in our proposed integration approach to model the core processes in a healthcare facility, while the adoption of the openBIM methods enable the calculation of the durations of space-dependent activities. Figure 3 represents the main phases of our proposed approach. Phase 1 (Experimental settings definition) involves defining the service requirements of the simulation with the key stakeholders. Since the focus is on the building's core processes, these stakeholders include Operations and Facilities Managers. This phase concerns the definitions of the desired capabilities of the process simulation, which will be used for decision-making in operation and process improvements. Phase 2 corresponds with process logic modelling, carried out via empirical research, collaboration with operations managers, and data extraction from Standard Operating Procedure (SOP) documents. This allows us to identify the main process stages, their inter-dependencies, the key process parameters, and the constraints/inputs/outputs of each stage, as described in [46]. Phase 2 also forms the basis of the mathematical modelling performed in Phase 3.

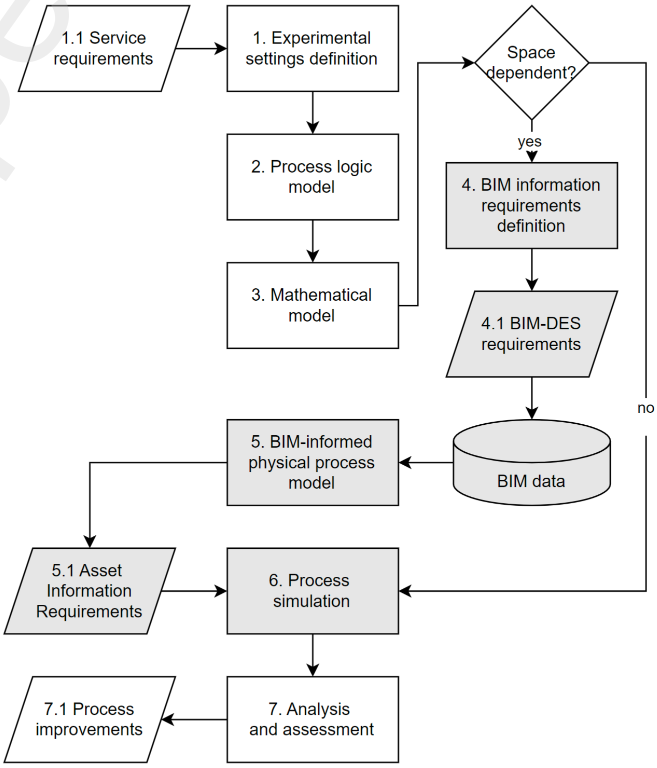

Figure 3: Phases of research for our proposed BIM-DES integration approach. Phases addressed in this paper are highlighted in grey.

At this point, the space dependencies of the core processes can be identified, such that the process schema in Fig. 3 branches into two. If the modelled process is not space-dependent, the process simulation (Phase 6) can be developed without using

Listing 1: Import ifcOpenShell packages and obtain the required spatial/semantic data.

```
from ifcopenshell import geom as ifc_geom
from ifcopenshell.util import shape as ifc_shape

# Get the name of an IFC object.
def get_level_name(obj: ifc.entity_instance) -> str:
    return (
        obj.ContainedInStructure[0].RelatingStructure.Name
    )

# Get the list of elevations for each IfcBuildingStorey
# Our IFC file is known to express elevation in mm,
# convert to m.
elevations: dict[str, float] = reduce(
    lambda d1, d2: d1 | d2,
    map(
        lambda s: {s.Name: s.Elevation/1000.0},
        ifc_file.by_type("ifcBuildingStorey")
    )
)

# Get the bounding box of an IFC object; for our IFC file,
# all walls and doors are aligned to the xyz axes.
def get_coords(
        obj: ifc.entity_instance) -> dict[str, float]:
    shape = ifc_geom.create_shape(settings, obj)
    grouped_verts = ifc_shape.get_vertices(shape.geometry)
    return {
        'x0': min(map(lambda xyz: xyz[0], grouped_verts)),
        'y0': min(map(lambda xyz: xyz[1], grouped_verts)),
        'z0': min(map(lambda xyz: xyz[2], grouped_verts)),
        'x1': max(map(lambda xyz: xyz[0], grouped_verts)),
        'y1': max(map(lambda xyz: xyz[1], grouped_verts)),
        # 'z1': max(map(lambda xyz: xyz[2], grouped_verts))
    }

# Extract door data, only for doors labelled d1, d2, d3...
# through the IfcDoor.Name property
doors: list[ifc.entity_instance] = list(
    filter(
        lambda door: bool(re.match(r'd\d+$', door.Name)),
        ifc_file.by_type("IfcDoor")
    )
)
doors_coords = [get_coords(door) for door in doors]

# Extract wall data
walls = ifc_file.by_type("IfcWall")
wall_coords = [get_coords(wall) for wall in walls]
```

BIM data; otherwise, BIM data is used to obtain the location, geometries, and semantics of the spaces and physical assets involved. However, since the BIM information is not usually created to support the simulation of core processes in buildings, the BIM-DES information requirements must be defined (Phase 4). In this this phase the geometric, alphanumerical Level of Information Needs (LOIN) [47] is defined (4.1 BIM-DES requirements) and mapped to the existing BIM data to ensure the BIM-DES integration. These information requirements are embedded into the BIM model (5.1: Asset Information Requirements – AIR) and can be also used as a reference to inform the design process of similar buildings where a BIM-DES integration is needed.

The DES-informed AIRs are then used in the process simulation (Phase 6) to investigate how space and location impact upon the process performance. Finally, Phase 7 corresponds to the analysis and assessment carried out to evaluate the simulated process performance against a set of Key Performance Indicators (KPIs) derived from the service requirements. The output of this phase is a set of process improvements in the form of recommendations, system automation, and notifications (7.1: Process improvements). The phases highlighted in grey in Fig. 3 are addressed this paper.

4. Implementation in the Histopathology Laboratory Digital Twin use case

The proposed BIM-DES approach has been implemented in the Histopathology laboratory at Addenbrooke's Hospital in Cambridge, UK, within the context of developing a laboratory digital twin (DT). The histopathology laboratory is a crucial function for the efficient treatment of patients. The main KPI used to asses the histopathology service is the turnaround time (TAT) distribution, measured as the percentage of cases processed end-to-end within a given time frame, from case creation (the sample is booked-in at the lab) to the issuing of a histopathologist report. Within the Histopathology department, the histological specimen (i.e. a section of human tissue), once taken from a patient, undergoes a series of stages until a pathological diagnosis is carried out and reported to the patient. These phases are enumerated in Table 3, which is re-elaborated from [46] and are located in the levels 3 and 4 of the building represented in Fig. 4, part of the biomedical campus.

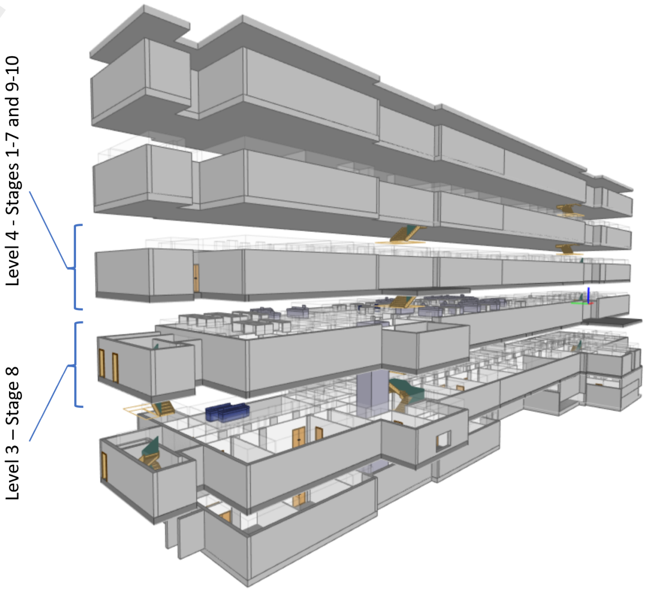

Figure 4: 3D model in IFC of the Biomedical building of Addenbrooke's hospital. The Histopathology laboratory is hosted in Levels 3 and 4, corresponding to the first and second floors above the ground.

However, within the laboratory setting, the reporting time cannot be completely controlled, since it depends on the pathologists which are external to the laboratory (Stage 12 in Table 3).

Therefore, we define the main laboratory process as Stages 1 to 11 in Table 3. The crucial KPI within the laboratory is thus the "lab TAT", which unlike overall TAT lies within the control of the laboratory's Operation Managers.

Table 3: Main histopathology process stages. Note that Levels 3 and 4 correspond to the first and second floors of the building, as illustrated in Fig. 4.

N	Stage	Level	Sample Processing
1	Reception	4	—
2	Cut-up	4	Single
3	Processing	4	Batch
4	Embedding	4	Single
5	Microtomy	4	Single
6	Staining	4	Batch
7	Cover-slipping	4	Batch
8	Digital scanning	3	Batch
9	Collation	4	Single
10	Block check & quality check	4	Single
11	Case allocation	4	Batch
12	Reporting	—	Single

Between laboratory stages 1 to 10 in Table 3, the specimens are transferred individually or in batches to the next space-function, e.g. from the reception to the cut-up rooms, all the way though to the block checks and quality checks (n. 10). Stages 11 and 12 of the process are not dependent on the physical movement of specimens. As a result, the laboratory TAT is affected by the space-time variables. In addition to this, not all the process stages are executed in functional areas located on the same floor of the Addenbrooke's Hospital's Biomedical building. In fact, in the analysed laboratory settings, the batches of specimens are transferred from Stage 7 to Stage 8 and from Stage 8 to Stage 9 using the lift. This ties the laboratory throughput to the maintenance condition of that equipment — if the lift fails and is out of order, whomever carries the batch of specimens between floors (the "runner") must instead use the stairs, significantly impacting the laboratory TAT (see Fig. 4). However, the laboratory operations managers can only guess from experience the effect of this dependency, and have no real control over the laboratory process' throughput under this scenario, nor decision-making power over the lift inspection/maintenance schedule. The quantification of this dependency is not possible without an integrated DT which inform the Operations and Facilities Managers of the current shortest path for specimen transfer based on the built assets' performance, and the predicted TAT after any change of the transfer pathways.

In use case, we propose a solution for this complex scenario, focusing on the application of the developed BIM-DES approach to provide insights on the core process operation. To address this challenge, the following service requirements have been identified through the co-operation with the laboratory operations team:

1. Support the team to identify where the bottlenecks are.
2. Predict what the impact of staff allocation is to TAT and service levels.
3. Determine what the staff-machine utilisation is.
4. Quantify the impact of equipment failure on the laboratory KPIs.
5. Quantify the effect of the layout organisation on flow and KPIs.

In this article, we describe how our proposed solution addresses Requirements 4 and 5 above.

4.1. *The BIM-DES Digital Twin architecture*

The paradigm of "DT with human-in-the-loop" [48] was adopted for this use case. The Operations and Facilities Managers are considered to be the direct beneficiaries and can both input data into the DT environment and use its outputs to implement actions on the physical process. The Histopathology digital environment and the DT frontend are out of the scope of this paper, therefore they have been greyed out in Fig. 5. Focusing on the DT backend environment instead, a set of independent modules have been developed to address the need for data availability, accessibility, and timeliness in responding the the Histopathology lab needs.

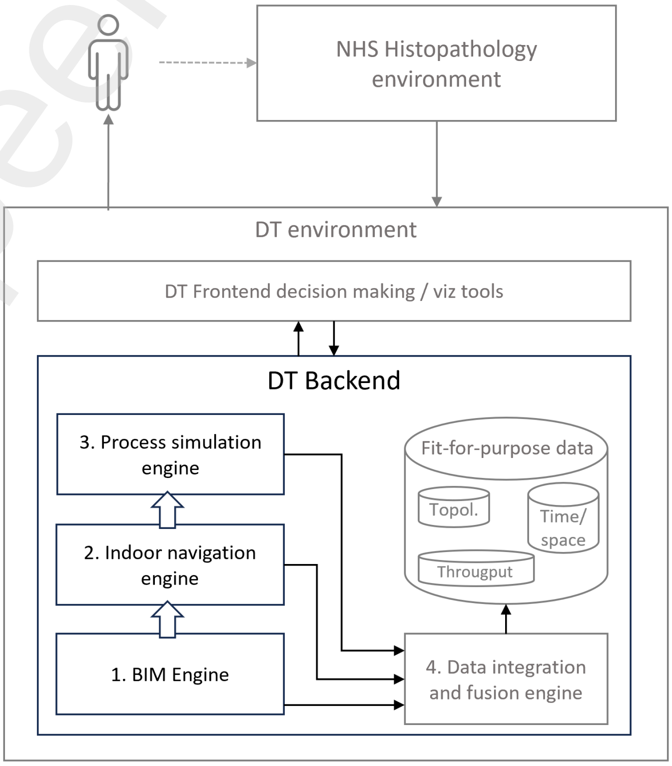

Figure 5: Histopathology Laboratory DT architecture. Plain arrows represent the data flow. Dashed arrows represent the actuation flow. Bold arrows represent optional data flows. Greyed out parts represents aspect out of the scope of this paper.

In this article, we focus on the functionalities of the BIM (1) and Indoor navigation (2) and Process simulation (3) engines, shown at the left hand side of the DT backend environment in Fig. 5. The three engines have been designed to be able to run independently, enabling the parallel computing of the results used in the Data integration and fusion engine (4).

This fourth module aggregates and transforms the data from the other engines based on the data requirements of the frontend tools, which access the fit-for-purpose data via a backend access layer. The simulation results can be obtained with or without the use of BIM data input, transferred after being processed through the indoor navigation engine. In fact, the Process simulation engine (3) only communicates with the Indoor navigation engine (2) when updated BIM information is needed and/or available. This can happens for example in case of major changes to the building layout. This simple and flexible architecture allows to decouple BIM and the core-process simulation and allows to extend to additional services engine (e.g., IoT, Asset maintenance etc.) without affecting the existing modules.

4.2. openBIM information requirements definition

The idea behind the definition of the BIM information requirements is that any spatial element which has an impact on the Histopathology laboratory throughput must be modelled in BIM with a certain level of detail. Each Histopathology process has a physical location and the samples follow the process logic being physically moved across the laboratory spaces. The time spent by the runner to transfer the samples (or the batches of samples) from one space-function — corresponding to the process stages — to another, is computed using BIM data. Since the resolution of the process simulation does not require to calculate the time for moving the materials within the same space, the time door-to-door is sufficient to inform the DES model. Thus the doors (i.e., IfcDoor elements) used as access to the subsequent functional areas of histopathology process have been identified and tagged in IFC, as described in Section 3.2 and shown in Table 4. The topology of the building forces to move samples through corridors and other connectivity spaces which do not have a clear characterisation form a histopathology process point of view, still must be considered for the path planning. These are d7 and d10 to d15.

This simple set of openBIM information requirements is sufficient to use the geometric definition of the assets and spaces modelled in BIM to enable the development of the BIM-informed physical model as described in Fig. 3.

4.3. Process simulation development

Based on the framework described in Section 3.1, we implemented a DES program in Python. A flowchart of the defined processes of the simulation program is given in Fig. 6. The colours of each process indicate the process type (green = `Process`, pink = `BatchingProcess`, orange = `CollationProcess`, blue = `DeliveryProcess`) as defined in Section 3.1. The arrow colours denote the type of entity being passed between processes (black = `Specimen`, red = `Batch[Specimen]`, blue = `Block`, green = `Batch[Block]`, pink = `Slide`, purple = `Batch[Slide]`).

To model the flow of entities through the histopathology lab, a class hierarchy of `Specimen`'s, `Block`'s, and `Slide`'s was defined. A generic `Batch` class was also defined to hold multiple specimens, blocks, or slides in a single entity, for machine processes and deliveries. Note that the output type of a process can

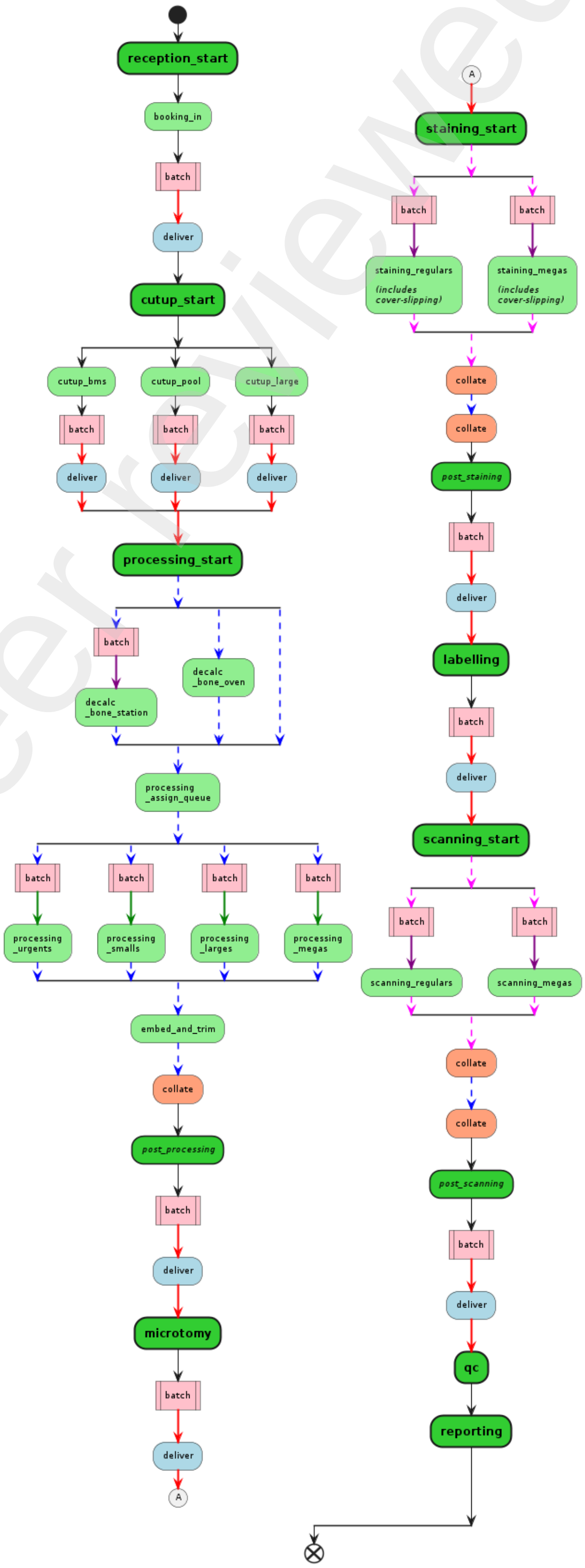

Figure 6: The defined processes of the discrete-event simulation model as a UML activity diagram. See Section 4.3 for a detailed explanation.

Table 4: BIM-DES Requirements - IfcDoor elements key for the OpenBIM to DES Integration

Level	IfcName	IfcDescription	Process Phase
Level 3 First floor	d16	Digital Pathology	Digital Scanning
Level 3 First floor	d15	Corridor	
Level 3 First floor	d14	Corridor	
Level 3 First floor	d13	Lift	
Level 3 First floor	d12	Landing	
Level 4 Second floor	d11	Lift	
Level 4 Second floor	d10	Landing	
Level 4 Second floor	d9	Staining Room	H&E Staining, Slide cover-slipping
Level 4 Second floor	d8	Main Lab	Microtomy/Slide printing, Case/slide collation, Block check and Quality Check, Case allocation
Level 4 Second floor	d7	Corridor	
Level 4 Second floor	d6	Processing Room (Embedding)	Cut-up
Level 4 Second floor	d5	Green Room (Cut-Up)	Cut-up
Level 4 Second floor	d4	Yellow Room (Cut-Up)	Cut-up
Level 4 Second floor	d3	White Room (Cut-Up)	Cut-up
Level 4 Second floor	d2	Lilac Room (Cut-Up)	Cut-up
Level 4 Second floor	d1	Specimen Reception	Reception

be different from its input type if splitting is performed within the process. Additionally, `Process` instances may have multiple outputs, with each entity sent to one of the outputs based on the internal rules of the process.

Statistics collection for resources and queues in the simulation is enabled by default in the Salabim Python library. For example, the `Resources` class contains a number of `Monitor` objects to track the number of claimed resources, the total capacity of the resource, and the number of waiting requests over time. In addition, we attach a Python dictionary to the simulation model to store specimen attributes, particularly timestamps recording the start and end of each process stage or group of stages. For example, in Fig. 6, staining and coverslip application (Stages 6 and 7 in Table 3) have been combined, as further study of the histopathology process revealed that these two stages are completed by the same combination machine (except for mega slides which are coverslipped manually). Finally, the openpyxl Python library was used to extract simulation parameters from an Excel configuration file including:

- specimen arrival rates, hourly;
- task durations;
- batch sizes;
- staff allocation schedules;
- branching probabilities (e.g., cut-up type);
- process stage-to-door mapping for BIM-DES integration; and
- number of blocks and slides per specimen (set of random distribution parameters for each specimen and block type).

4.4. BIM-DES integration

The purpose of the BIM-DES integration is to inform the simulation model with the duration of the space-dependent specimen transfer times. The steps followed to develop the integration are as follows:

1. For each door, wall, and column obtain the local coordinates.
2. Overlay a rectangular grid to each floor in the study and mark each grid square as open space (no containment of any of the previous objects), obstacle (contains wall or column), or door (contains a door).
3. For each pair of doors on the same floor:
   (a) Filter out all grid nodes blocked by an obstacle or door (other than the source and target door).
   (b) Add diagonal edges to the grid, but only inside complete rectangles of four non-diagonal edges.
   (c) Compute the distance between the two doors by applying Dijkstra's[2] algorithm to the marked grid.
4. Use the distances computed in Step 3 to build a logical graph of the floors under study. Add edges corresponding to transfer modes between floors, depending on the state of each transfer mode (for example, if the lift is out of order, do not add the corresponding edge to the logical graph).
5. For each pair of consecutive stages, use Dijkstra's algorithm on the logical graph to compute the travel time between the doors corresponding to the two stages.
   (a) For the cut-up stage, which takes place in multiple cut-up rooms, all four room doors are treated as a single node and the average travel time used.

The Python code corresponding to the steps above is given in Listings A.1 and A.2, providing the `BimModel` and `ShapelyModel` classes, respectively. In particular, `BimModel` contains semantic and numerical coordinate data for each door and wall under study using a pair of Pandas dataframes, while `ShapelyModel` represents these doors and walls as `Polygon` objects using the Shapely library. Plotting functionality was also added to the `ShapelyModel` class, resulting the graphical output shown in Fig. 7. The grid (Fig. 8) and logical graph (Fig. 9) in Steps 2–4 above are represented using the networkX Python library, which is also used to execute Dijkstra's algorithm in Step 5. Note that the grid size of 0.5m is defined to be always smaller than the minimum standard width of a door (which is 0.9m). It can be seen that while doors `d1` to `d7` and `d10` form a complete graph (free travel between any two doors in this group), the remaining doors are more weakly connected

---

[2] Note that heuristic-based algorithms such as A* will still require exhausting *all* nodes reachable from the originating door if it is not connected to the destination door in our grid, thus performing no better than Dijkstra's algorithm.

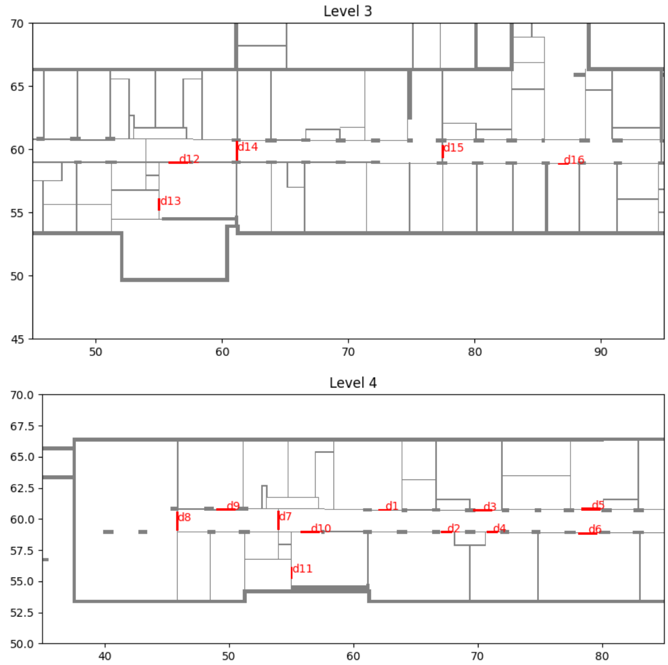

Figure 7: Result of plotting the two `ShapelyModel` instances representing Levels 3 and 4 of the histopathology laboratory, respectively.

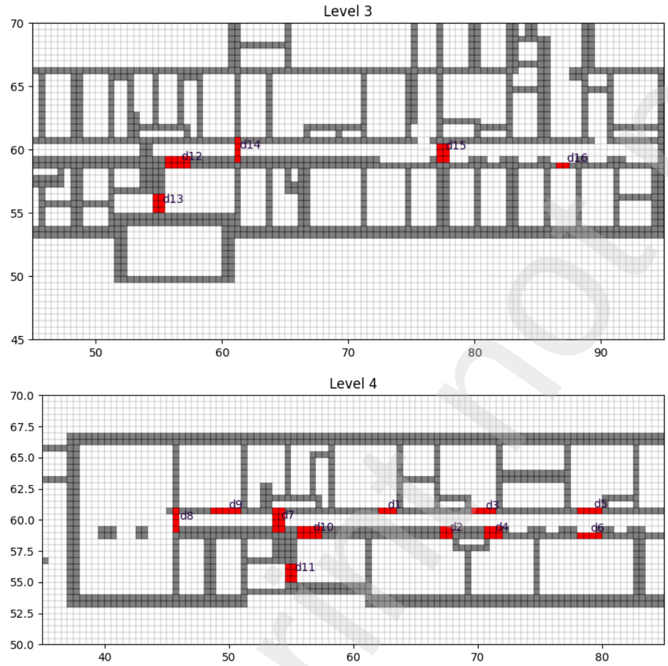

Figure 8: Grid approximation of Levels 3 and 4 of the histopathology laboratory, respectively.

to the core of the graph, with `d7` and `d10` forming bottlenecks that all specimens must pass through.

Note the addition of diagonal edges in Step 3b. This helps to find edge lengths closer to the shortest possible path in free space. The reason we do not allow diagonal edges except within complete rectangles is illustrated in Fig. 10, where the diagonal line touches the corner of the obstructed grid square, leaving no space between the straight-line path and the obstruction.

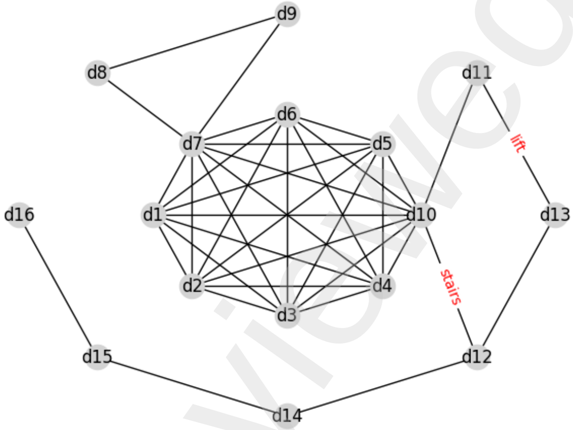

Figure 9: Logical graph of the histopathology laboratory. A lift failure is modelled by removing the edge between doors `d11` and `d13`.

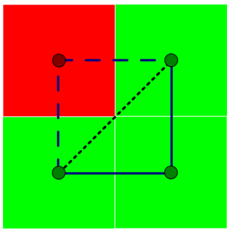

Figure 10: A grid with three open (green) and one obstructed (red) square. Since the edge between the centres of the two diagonal open squares (black dashed) touches the obstructed square, only the two solid lines are considered as part of the final grid in Step 3b of the BIM-DES integration process.

4.5. Impact of space on the process laboratory performance

Figure 11 shows the runner times between each pair of doors in the histopathology lab (Step 4 output) under normal operation, i.e., lift is working. This results in the total runner times between stages (Step 5 output) as shown in Table 5. It can be seen that the runner times to and from the Scanning stage are much larger than between any of the other process stages, due to the digital scanning room being located on a different floor (level 3 in Fig. 4) from the rest of the histopathology lab (level 4 in Fig. 4).

To test the capabilities of the integrated BIM-DES approach and verify the impact of the space variables on the process performance, we consider the scenario where the lift used to carry the slides from the main lab to the digital scanning and back (process phases 7-9 in Table 3) is out of order. Thus the resultant runner times between stages must be computed considering delay due to the use of the stairs instead. The glass slides are very fragile objects and the runner needs to pay extra attention when they are moved through the stairs and when crossing doors, which are always closed for safety reasons in the lab environment. Also, the dimension of the batch can be large in some cases and this may require to break it down in smaller assemblies to be carried the digital scanning one by one as opposed to using a trolley and carry all of them to the next stage using the lift. For these reasons the runner times to and from the Scanning stage is estimated to be almost 80% higher than in the base scenario where the lift is working.

To gauge the impact of this increase, we ran the process simulation model for both sets of runner times, with results shown in Fig. 12. It is demonstrated that the increase in runner times under the lift-down scenario results in a lower service level of the laboratory. In particular, the difference in the proportion of specimens completed — Reception to Block & Quality Check stages as defined in Table 3, i.e., used to calculate the laboratory TAT — is statistically significant at the 7- and 10-day marks. The result here can seem counter intuitive at first, as the mean lab TATs for the two scenarios are 8.6 and 9.4 days, respectively, corresponding to a TAT increase of 9.3% when the lift is out of order relative to the base scenario. On the other hand, the transfer of specimens between floors only accounts for a small percentage of this increase. This large performance loss relative to the small increase in runner times can be explained by the knock-on effects of increased staff utilisation on the queuing times of other tasks in the overall histopathology process, some of which may be pushed to the next day due to shift scheduling caused by the longer time in moving samples (slides in this case) between floors.

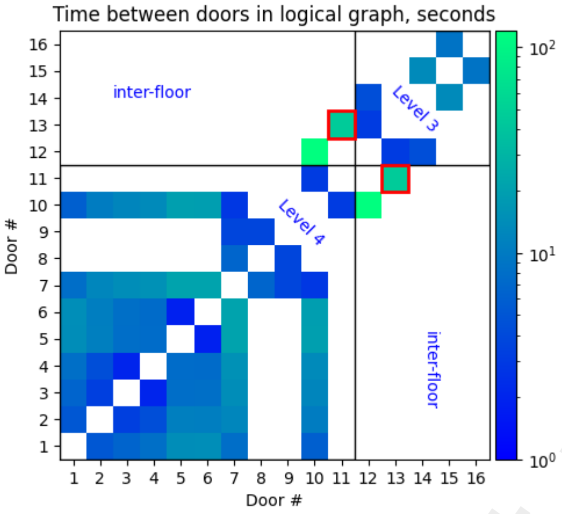

Figure 11: Runner times between doors for the base scenario, using a speed of 1.2m/s within each floor. For the lift-down scenario, the highlighted (red outline) entries, corresponding to the graph edge marked “lift” in Fig. 9, are removed.

Table 5: Runner times computed from the logical graph shown in Fig. 9

Runner Journey	Duration, seconds Base scenario	Lift down
(reception, cutup)	8.452	8.452
(cutup, processing)	6.969	6.969
(processing, microtomy)	28.089	28.089
(microtomy, staining)	3.679	3.679
(staining, labelling)	3.679	3.679
(labelling, scanning)	86.898	155.863
(scanning, qc)	86.898	155.863

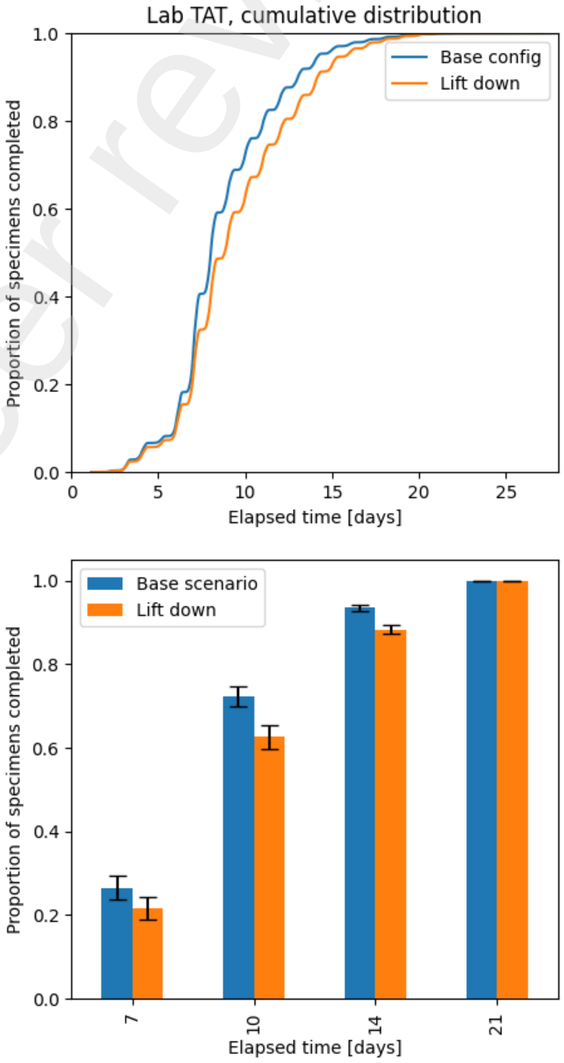

Figure 12: Comparison of lab TAT for two scenarios with the lift working and out-of-order, respectively. The runner speed is set at 1.2m/s within each floor for horizontal transfers. Error bars denote 95% confidence intervals based on 30 simulation runs.

5. Discussion

In this paper, we described a BIM-DES integrated approach, and showed how it can be integrated into a DT architecture for a histopathology laboratory (Fig. 5). The research on DTs has flourished in the past few years and considerable advancements have been done in the application of digital technologies to support built assets’ maintenance and operation. However, the majority of the research on the integration of BIM and M&S (see Fig. 1 and Table 1) has focused on non-core processes e.g., Asset Operation and Energy performance management, with little attention to the impact on the core business processes, namely those implemented to deliver the main business outcomes, which is in our use case the histopathology laboratory service. This paper has addressed this knowledge gap and contributed to the understanding of how an integrated proces-space DT based on BIM and M&S can have an impact on the core processes operation. The benefits are described below.

The proposed approach links the core-process throughput to the efficiency of the built assets. So that the Operations Managers have clarity on the parts of the building which directly impact the core business. The use case has demonstrated that if the spatial layout is not appropriate or a critical asset fails (e.g., the lift), the core process performance is disrupted, with a direct impact on the clinical cases' reporting capability and so on the patients' health. Implementing our proposed approach, the Operations Managers have solid evidence to make decisions on the resources that highly impact the laboratory TAT and the re-organisation of the laboratory layout.

On the other hand Facility Managers can use the results of the integrated BIM-DES to define a lab service-based asset maintenance prioritisation strategy which identifies and allocate more investments to those parts of the building which may cause major disruptions. This gives clear evidence of why Estates (healthcare facilities in this case) and the Clinical services should not be siloed and thereby make integrated decisions based on the interdisciplinary insights provided by the space-process DT.

Overall, the proposed approach increase the laboratory resilience and helps finding viable solutions in the case of disruptions to the laboratory process. This could happen in the case of a major building failure which results in the need to interrupt part of the laboratory operation. This could happen, for example, in major rehabilitation for the presence of highly degraded Reinforced Autoclaved Aerated Concrete (RAAC). RAAC was widely used in mid-20th century in the UK and its service life is now coming to an end [49], posing a real problem for the NHS Estates. The integrated DT can help answering the question: when should the histopathology process interrupted in the case of a temporary decamp or when the whole function needs to be moved to a new building? The impact of having clarity on this issue can save time critical resources and helps controlling the cases' backlog.

5.1. *Improved BIM-based indoor navigation*

Several challenges remain too. The indoor navigation algorithm used in this paper only allows travel in the eight ordinal directions. A more advanced pathfinding algorithm may allow travel in a straight line between any two grid points as long as the path between them is unobstructed (any-angle path planning) [50]. Moreover, we used a weighted graph method to compute the travel time in 3D (through the stairs and the lift). Although, more advanced navigation methods exist in literature [51], the current version of the BIM-DES approach does not require such complexity, which is instead an issue to consider in the case of a more detailed process simulation. In fact, we have used the distance between doors as a proxy for the runner time between process stages, but have ignored the internal geometries of each room involved in the histopathology process, e.g. workbenches and machines.

The spatial variables have been identified as described in [46] and modelled adopting a low level of geometric detail in the case of the use case building. The LOIN in this case depends on the resolution of the process simulation which has been developed to support the decision on the whole laboratory. However, the BIM model offers already a higher level of granularity, which could be used to compute the travel time within each space/function. Although this data is available and summarised in Table 6, it has not been used in the current version of the BIM-DES approach, offering the opportunity for a further development of the research.

Table 6: Assets classification and description.

Asset / Equipment	**Type**	**Description**
PC		Each workstation is equipped with a PC connected to the LIMS.
Barcode Scanner		Part of the equipment of each workstation.
Reception Desk		Desk used at the specimen reception.
Labels Printer		Part of the equipment of the reception desk.
Batching/Storage Desk	Sorting	Used for sorting the specimens received.
Batching/Storage Desk	Store Booked-In	Used to store the specimens booked-in.
Cassette Printer		Special machine used to label the cassettes where the specimens are placed once cut-up.
AFOS Bench		Special bench used for the specimens cut-up.
Microscope		Used for quality slides quality check.
Para Trimmer		Special equipment used to refine the cut-up specimens embedded in paraffin wax.
Oven	Mega Oven	Special temporary storage for embedded blocks.
Tissue Processor		Special equipment used to prepare the specimens for embedding.
Embedding Station		Workstation for embedding in paraffin wax.
Cold Plate – Mega		Cold surface used to solidify mega sections embedded in wax.
Ventilated Store		Special storage for chemicals and chemically treated specimens.
Fridge		Temporary storage for specimens.
Chemical Store		Store for chemicals.
Staining Machine		Special Equipment for staining and slides coverslipping.
Microtomes		Special equipment to produce thin sections of embedded specimens which are subsequently stained.

5.2. *Real-time data communication*

The simulation model in this paper can, with some modification, be used for process performance forecasting by reading in a fresh building state (i.e., whether the lift is working) and process state (i.e., the status of all work-in-progress specimens) at the beginning of each simulation round. This will require a data serialisation format for reading and writing the full simulation state from/to a file, e.g., JSON (ISO 21778), MessagePack or HDF5. A method of estimating residual task durations based on time elapsed is also required, i.e., generating random variates from the truncated probability distribution (i.e., the distribution of a task's duration conditioned on the time already elapsed). Rejection sampling is suitable for this but requires further study to minimise the rejection rate for computational efficiency.

Alternatively, the process flow logic of the simulation model can be used to track the progress of the specimens within the laboratory in real-time, with the simulation component removed. However, such fine-grained progress tracking of specimens requires additional real-time data collection. Location tracking using bar code scanners or sensors may also be added to ensure that the specimens are where they are supposed to be according to the process logic. Further discussion with operations managers is required to determine the LOIN for real-time process tracking and/or simulation-based forecasting, as well as designing methods to reach this LOIN.

5.3. Integration of Asset and Facilities Management data

Under the FM perspective, the performance of the built assets have been considered as an input for the scenario analysis presented in Section 4.5. However this is a simplification of the asset performance in an extreme high-risk situation (i.e., a faulty lift), needed to demonstrate the impact of a critical asset on the overall process performance. However, the built asset performance can be modelled with higher level of detail and the condition of both the built assets and spaces can be defined using a stochastic approach or data driven methods e.g., using environmental and asset monitoring sensors. This requires to develop new engines, respectively a Facilities Management and an IoT engine which require further research work.

Similarly, the performance of the clinical equipment has been considered in the process simulation model, though this not does not represent a limitation for the overall process throughput since the skill mix and the staffing allocation do not saturate the equipment capacity. However, the maintenance of these assets represent a criticality and, similarly to the built assets, any disruption to clinical equipment could heavily impact on the core process. The integration of this data requires additional work and a closer relationships with the Clinical Engineering team to define an additional Asset Management engine.

5.4. An ontology for the integrated Digital Twin

To realise the full potential of the the proposed DT architecture able to handle BIM, M&S, AM, FM and IoT information must be structure in a consistent and interoperable manner. For example, within the BIM domain, the IFC schema is an application agnostic and non-proprietary schema which offers the opportunity to access and use building information using a variety of software and libraries (such as IfcOpenShell in this paper). However, IFC does not contain enough classes to represent the clinical assets in the histopathology laboratory, other than the generic `IFCProduct`. On the other hand, since IFC is specified using the EXPRESS (ISO 10303-11) data modelling language, it is possible to create a superset of IFC to include one's specific use cases.

IFC also contains classes for process modelling, including `IFCProcess` corresponding to the processes described in Fig. 2 and `IfcWorkCalendar` to define the allocation of staff resources. Therefore, it may be possible to describe the operations of the histopathology laboratory within the IFC schema, extended as described above to express histopathology-specific asset types. Nevertheless, it may be preferable to describe these processes using other modelling schemas, such as UML (particularly activity and sequence diagrams) or BPMN (ISO 19510). To describe the events (corresponding to `IfcEvent`) of these processes and corresponding changes to the laboratory state, formal specifications include the Discrete Event System Specification (DEVS) and its extensions [52], as well as stochastic Petri nets [53]. A benefit of DEVS and Petri nets is that simulation tools already exist for models defined using these specifications, e.g., PythonPDEVS [52] and DEVSimPy [54][3] for DEVS; and Timenet [55] and ORIS [56] for stochastic Petri nets.

Also, the research on ontologies and knowledge graphs the built environment domain has demonstrated the possibility of enriching the existing schemes and standards with a wide variety of data models [57], which increase capability of representing any asset, spatial, or process object and physical/virtual agents. The scope is potentially much broader and creates the opportunity of developing an full-hospital DT ontology, including clinical services, logistics and remote patient monitoring.

6. Conclusions

This paper has addressed Research Question 1 (defined in the Introduction) through the definition of an openBIM method to define the information requirements, parse the IFC data, and process the geometric information to inform the space-dependent variables of a DES. The method is developed using open source technologies and can be applied to a variety of buildings. The use case of an histopathology laboratory facility has demonstrated the effectiveness of the proposed method. The BIM-DES integration approach improves the decision making capabilities of the Operations and Facility Managers (Research Question 2) of the laboratory who can:

- simulate failures to critical built asset and their impact to the core process,
- build a scenario analysis and compare the current situation with the performance of the laboratory under the new condition; and
- have a strong business case to allocate additional investments on spaces and built assets which allows to optimally operate the core process and anticipate any disruptions to the workflow.

The research question 3 has been partially addressed through proposing the space-process DT architecture, though additional

---

[3] Not related to the other DES Python library, Simpy

research is required to enable the effective curation of the cyber-physical infrastructure, develop ethical data pipelines and maintain the functionalities, mirroring and influencing the operation of the physical counterparts.

Acknowledgements

This research is funded by the Cambridge University Hospitals NHS Foundation Trust. We deeply thank Zahrah Rosun and Colin Carr from the Department of Histopathology, Addenbrooke's Hospital for giving us access to the laboratory and sharing their knowledge. We also thank the Lifecycle Management Group Ltd team for the fruitful conversations on Histopathology process modelling.

Data accessibility statement

The key code used in this paper is available in Appendix A. The full Python code base for this paper can be found at https://github.com/yinchi/histopath-bim-des.

Appendix A. Program Listings

Program listings for Section 4.4 (BIM-DES integration) are given in Listings A.1 and A.2.

Listing A.1: `BIMModel` class definition for the BIM-DES integration.

```
from dataclasses import dataclass
from functools import reduce
from itertools import islice, product

import ifcopenshell as ifc, ifcopenshell.geom as ifc_geom, ifcopenshell.util.shape as ifc_shape
import matplotlib.axes, natsort, networkx as ntx, numpy as np, pandas as pd, re, shapely as shp

from os import PathLike
from typing import Self
from shapely.plotting import plot_polygon

import sys
if '..' not in sys.path: sys.path.append('..')

from histopath_bim_des.config.runners import RunnerConfig

(settings := ifc_geom.settings()).set(settings.USE_WORLD_COORDS, True) # Find global coordinates

@dataclass
class BimModel:
    elevations: dict[str, float]
    doors: pd.DataFrame
    walls: pd.DataFrame

    @staticmethod
    def from_ifc(path, door_filter = r'd\d+$'):
        ifc_file = ifc.open(path)
        elevations: dict[str, float] = reduce(
            lambda d1, d2: d1 | d2,
            map(lambda s: {s.Name: s.Elevation/1000.0}, ifc_file.by_type("ifcBuildingStorey")))

        def get_level_name(obj): return obj.ContainedInStructure[0].RelatingStructure.Name

        def get_coords(obj):
            shape = ifc_geom.create_shape(settings, obj)
            v = ifc_shape.get_vertices(shape.geometry)
            return {'x0': min(map(lambda xyz: xyz[0], v)), 'x1': max(map(lambda xyz: xyz[0], v)),
                    'y0': min(map(lambda xyz: xyz[1], v)), 'y1': max(map(lambda xyz: xyz[1], v)),
                    'z0': min(map(lambda xyz: xyz[2], v))}

        # Extract door data
        doors = list(filter(lambda door: bool(re.match(door_filter, door.Name)),
                            ifc_file.by_type("IfcDoor")))
        doors_coords = [get_coords(door) for door in doors]
        doors_df = pd.DataFrame({
            'door_name': [door.Name for door in doors], 'floor': [get_level_name(door) for door in doors],
            'x0': [box['x0'] for box in doors_coords], 'x1': [box['x1'] for box in doors_coords],
            'y0': [box['y0'] for box in doors_coords], 'y1': [box['y1'] for box in doors_coords],
            'z0': [box['z0'] for box in doors_coords],
        })\
        .sort_values(by='door_name',key=natsort.natsort_keygen())\
        .reset_index(drop=True)

        # Extract wall data
        walls = ifc_file.by_type("IfcWall")
        wall_coords = [get_coords(wall) for wall in walls]
        walls_df = pd.DataFrame({
            'wall_name': [wall.Name for wall in walls], 'floor': [get_level_name(wall) for wall in walls],
            'x0': [box['x0'] for box in wall_coords], 'x1': [box['x1'] for box in wall_coords],
            'y0': [box['y0'] for box in wall_coords], 'y1': [box['y1'] for box in wall_coords],
            'z0': [box['z0'] for box in wall_coords]})
        return BimModel(elevations,doors_df,walls_df)

    def to_shapely(self, level: int) −> 'ShapelyModel':
        """Returns a Shapely representation of a floor in the `BimModel`."""
        wall_shapes = [
            shp.box(wall.x0, wall.y0, wall.x1, wall.y1, ccw=False)
            for wall in self.walls.loc[self.walls.floor.str.contains(f'Level {level}')].itertuples()]
        door_shapes = {
            door.door_name: shp.box(door.x0, door.y0, door.x1, door.y1, ccw=False)
            for door in self.doors.loc[self.doors.floor.str.contains(f'Level {level}')].itertuples()}
        for s in wall_shapes + door_shapes.values(): shp.prepare(s)
        return ShapelyModel(wall_shapes, door_shapes)
```

References

[1] V. V. Tuhaise, J. H. M. Tah, F. H. Abanda, Technologies for digital twin applications in construction, Automation in Construction 152 (2023) 104931. `doi:10.1016/j.autcon.2023.104931`.

[2] AIAA Digital Engineering Integration Committee, DIGITAL TWIN: DEFINITION & VALUE An AIAA and AIA Position Paper (2020).
URL `https://www.aiaa.org/docs/default-source/uploadedfiles/issues-and-advocacy/policy-papers/digital-twin-institute-position-paper-(december-2020).pdf`

[3] UK Government Office for Science, RTA: Digital twins - GOV.UK (2023).
URL `https://www.gov.uk/government/publications/rapid-technology-assessment-digital-twins/rta-digital-twins`

[4] W. AlBalkhy, D. Karmaoui, L. Ducoulombier, Z. Lafhaj, T. Linner, Digital twins in the built environment: Definition, applications, and challenges, Automation in Construction 162 (2024) 105368. `doi:10.1016/j.autcon.2024.105368`.

[5] National Infrastructure Commission, Data for the public good, Tech. rep. (2017).
URL `https://nic.org.uk/app/uploads/Data-for-the-Public-Good-NIC-Report.pdf`

[6] D. T. Hub, Driving cyber-physical innovation in the UK (2024).
URL `https://digitaltwinhub.co.uk/download/national-cyber-physical-infrastructure-ecosystem-programme-report-march-2024/`

[7] Singapore Smart Nation, Transforming Singapore trough technology (2021).
URL `https://www.smartnation.gov.sg/`

[8] C. Boje, A. Guerriero, S. Kubicki, Y. Rezgui, Towards a semantic Construction Digital Twin: Directions for future research, Automation in Construction 114 (November 2019) (2020) 103179. `doi:10.1016/j.autcon.2020.103179`.

[9] ISO, 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. Part 1: Concepts and principles (2018).
URL `https://www.iso.org/committee/49180.html`

[10] J. Patacas, N. Dawood, M. Kassem, BIM for facilities management: A framework and a common data environment using open standards, Automation in Construction 120 (June) (2020) 103366. `doi:10.1016/j.autcon.2020.103366`.

[11] S. Lavy, S. Jawadekar, A Case Study of Using BIM and COBie for Facility Management, International Journal of Facility Management 5 (2) (2014) 13–27.

[12] K. Schwabe, M. Dichtl, M. König, C. Koch, COBie: A Specification for the Construction Operations Building Information Exchange, Springer International Publishing, 2018, pp. 167–180. `doi:10.1007/978-3-319-92862-3_9`.

[13] K. A. Awwad, A. Shibani, M. Ghostin, Exploring the critical success factors influencing BIM level 2 implementation in the UK construction industry: the case of SMEs, International Journal of Construction Management 22 (10) (2022) 1894–1901. `doi:10.1080/15623599.2020.1744213`.

[14] X. Gao, P. Pishdad-Bozorgi, BIM-enabled facilities operation and maintenance: A review, Advanced Engineering Informatics 39 (2019) 227–247. `doi:10.1016/J.AEI.2019.01.005`.

Listing A.2: `ShapelyModel` class definition for the BIM-DES integration. See Listing A.1 for required `import` statements.

```
@dataclass
class ShapelyModel:
    """Shapely representation of a single floor in the histopathology lab."""
    wall_shapes: list[shp.Polygon]
    door_shapes: dict[str, shp.Polygon]

    def is_valid_box(self, box: shp.Polygon, ok_doors: list[str]):
        """Determines if box intersects with a wall or door except for `ok_doors`."""
        ok_door_shapes = [self.door_shapes[x] for x in ok_doors]
        shp.prepare(box)
        return (
            (True, 'ok_door') if any(box.intersects(ok_door_shapes))
            else (False, 'wall') if any(box.intersects(self.wall_shapes))
            else (True, 'empty'))

    def shortest_path(self, from_door: str, to_door: str, grid_size=0.5,bottom_left=(30, 45), top_right=(90, 70)):
        """Find the shortest path between two doors in the model."""
        x_min, y_min = bottom_left
        x_max, y_max = top_right
        n_x, n_y = len(np.arange(x_min, x_max, grid_size)), len(np.arange(y_min, y_max, grid_size))

        # Create base grid
        grid = ntx.grid_2d_graph(n_x, n_y)
        for i, j in grid.nodes:
            grid.nodes[(i, j)]['box'] = shp.box(
                x0 := x_min + i∗grid_size, y0 := y_min + j∗grid_size, x0+grid_size, y0+grid_size, ccw=False)
            shp.prepare(grid.nodes[(i, j)]['box'])
            grid.nodes[(i, j)]['pos'] = ((centroid := grid.nodes[(i, j)]['box'].centroid).x, centroid.y)

        selected_nodes = [n for n, v in grid.nodes(data=True)
                          if self.is_valid_box(v['box'], ok_doors=[from_door, to_door])[0]]
        grid2 = ntx.Graph(grid.subgraph(selected_nodes))

        # Add diagonals to grid2 within each complete "box" of 4 edges
        for _, _, v in grid2.edges(data=True): v['weight'] = 1.0
        for x, y in grid2.nodes:
            if ((x+1, y) in grid2.nodes and (x, y+1) in grid2.nodes and (x+1, y+1) in grid2.nodes):
                grid2.add_edge((x, y), (x+1, y+1), weight=2∗∗0.5) # northeast direction
            if ((x+1, y) in grid2.nodesand (x, y−1) in grid2.nodes and (x+1, y−1) in grid2.nodes):
                grid2.add_edge((x, y), (x+1, y−1), weight=2∗∗0.5) # southeast direction

        # Get node indexes for from_door and to_door
        from_node = [n for n, v in grid.nodes(data=True) if v['box'].intersects(self.door_shapes[from_door].centroid)][0]
        to_node = [n for n, v in grid.nodes(data=True) if v['box'].intersects(self.door_shapes[to_door].centroid)][0]

        path_nodes = ntx.shortest_path(grid2, from_node, to_node, weight='weight')
        path_edges = list(zip(path_nodes[:−1], path_nodes[1:]))
        path_graph = ntx.Graph()
        for i, n in enumerate(path_nodes): path_graph.add_node(i, pos=grid2.nodes(data=True)[n]['pos'])
        for i, e in enumerate(path_edges): path_graph.add_edge(i, i+1, weight=grid2.edges[e]['weight'])
        path_length = ntx.shortest_path_length(grid2, from_node, to_node, weight='weight') ∗ grid_size
        return path_length, path_graph

    def plot_floor(self, ax: matplotlib.axes.Axes, title: str, bottom_left=(30, 45), top_right=(100, 80)):
        """Plots the floor model using Matplotlib."""
        for p in self.wall_shapes: plot_polygon(p, ax, facecolor='gray', add_points=False, linewidth=0)
        for n, p in self.door_shapes.items():
            plot_polygon(p, ax, facecolor='red',add_points=False, linewidth=0)
            ax.text(p.centroid.x, p.centroid.y, n, color='red')
        ax.axis('square')
        x0, y0 = bottom_left
        x1, y1 = top_right
        ax.set(xlim=(x0, x1), ylim=(y0, y1), title=title)

def logical_graph(model: ShapelyModel, speed: float):
    """Construct a logical graph representation of a floor."""
    graph = ntx.Graph().add_nodes_from(keys := list(model.door_shapes.keys()))
    for i, k1 in enumerate(keys):
        for _, k2 in islice(enumerate(keys), i+1, None):
            try:
                path_len, _ = model.shortest_path(k1, k2)
                graph.add_edge(k1, k2, weight=path_len/speed)
            except ntx.NetworkXNoPath:
                continue
    return graph

def runner_times(model: BimModel, cfg: RunnerConfig) −> dict[tuple, float]:
    """Compute runner times between process stages in the histopathology model."""
    target_levels = [re.match(r'Level (\d+)', s).group(1) for s in model.doors.floor.unique()]
    logical_graphs = {level: logical_graph(model.to_shapely(level=level), speed=1.5)
                      for level in target_levels}
    full_logical_graph = ntx.compose_all(logical_graphs.values())
    for path in cfg.extra_paths: full_logical_graph.add_edge(∗path.path, weight=path.duration_seconds)

    d = cfg.door_map.model_dump()
    pairs = list(zip(k := list(d.keys()), k[1:]))

    ret = { }
    for u, v in pairs:
        if 'cutup' in (u, v):
            du, dv = d[u] if u == 'cutup' else [d[u]], d[v] if v == 'cutup' else [d[v]]

            # compute the average runner time for all cutup rooms
            ret[(u, v)] = np.average(
                [
                    ntx.shortest_path_length(full_logical_graph, d1, d2, weight='weight')
                    for d1, d2 in product(du, dv)
                ], weights=cfg.cutup_dist
            )
        else:
            ret[(u, v)] = ntx.shortest_path_length(full_logical_graph, d[u], d[v], weight='weight')
    return ret
```

[15] buildingSMART International, Activity proposal. ISO41000 FMBIM Protocol, IDS and CDE AKA "Facilities Management and openBIM (2020).
URL https://app.box.com/s/q4gy4lzw1kkuxynt0d5e13u98qmy4t12

[16] R. Charef, H. Alaka, S. Emmitt, Beyond the third dimension of BIM: A systematic review of literature and assessment of professional views, Journal of Building Engineering 19 (2018) 242–257. doi:10.1016/j.jobe.2018.04.028.

[17] C. Boton, Supporting constructability analysis meetings with immersive virtual reality-based collaborative BIM 4D simulation, Automation in Construction 96 (2018) 1–15.

[18] S. Kouhestani, M. Nik-Bakht, IFC-based process mining for design authoring, Automation in Construction 112 (2020). doi:10.1016/j.autcon.2019.103069.

[19] C.-S. Park, D.-Y. Lee, O.-S. Kwon, X. Wang, A framework for proactive construction defect management using BIM, augmented reality and ontology-based data collection template, Automation in Construction 33 (2013) 61–71. doi:10.1016/j.autcon.2012.09.010.

[20] J. Jia, S. Dou, S. Yang, Y. Wu, F. Cao, B. Li, H. Cui, Study on the application framework of BIM in the life cycle management of offshore wind farms, in: Proceedings of the International Offshore and Polar Engineering Conference, Vol. 1, 2019, p. 513 – 520. doi:10.1201/b12516-145.
URL https://www.researchgate.net/publication/337672936

[21] H.-J. Shin, H.-S. Cha, Proposing a Quality Inspection Process Model Using Advanced Technologies for the Transition to Smart Building Construction, Sustainability (Switzerland) 15 (1) (2023). doi:10.3390/su15010815.

[22] A. D. Chasey, S. Pindukuri, Information exchange requirements for capital equipment and facility infrastructure for semiconductor facilities, in: Construction Research Congress 2012: Construction Challenges in a Flat World, Proceedings of the 2012 Construction Research Congress, 2012, p. 437 – 446. doi:10.1061/9780784412329.044.

[23] W. Lu, T. Olofsson, Building information modeling and discrete event simulation: Towards an integrated framework, Automation in Construction 44 (2014) 73–83. doi:10.1016/J.AUTCON.2014.04.001.

[24] E. Alreshidi, M. Mourshed, Y. Rezgui, Cloud-based BIM governance platform requirements and specifications: Software engineering approach using BPMN and UML, Journal of Computing in Civil Engineering 30 (4) (2016). doi:10.1061/(ASCE)CP.1943-5487.0000539.

[25] F. R. Correa, Cyber-physical systems for construction industry, in: Proceedings - 2018 IEEE Industrial Cyber-Physical Systems, ICPS 2018, 2018, p. 392 – 397. doi:10.1109/ICPHYS.2018.8387690.

[26] L. Gebken, P. Drews, I. Schirmer, Enhancing the building information modeling lifecycle of complex structures with IOT: Phases, capabilities and use cases, in: Proceedings of the Annual Hawaii International Conference on System Sciences, Vol. 2019-January, 2019, p. 5929 – 5938. doi:10.24251/HICSS.2019.715.

[27] H. Liu, M. Al-Hussein, M. Lu, BIM-based integrated approach for detailed construction scheduling under resource constraints, Automation in Construction 53 (2015) 29–43. doi:10.1016/j.autcon.2015.03.008.

[28] H. Guo, X. Ye, Q. Ren, L. Zhubang, Automatic generation of construction schedules based on BIM and rule reasoning, Journal of Tsinghua University(Science and Technology) 62 (2) (2022) 189. doi:10.16511/j.cnki.qhdxxb.2022.22.001.
URL http://jst.tsinghuajournals.com/article/2022/4306/20220201.htm

[29] Y. Dong, Y. Zhang, F. Sang, Construction process simulation of building decoration based on bim and computer-aided technology, Computer-Aided Design and Applications (2023). doi:10.14733/cadaps.2023.S15.294-308.

[30] B. Liu, B. Yang, B. Zhang, M. Dong, S. Jiang, J. Xiao, Spatio-temporal deduction of floor construction based on the agent modeling of construction actors, Automation in Construction 142 (10 2022). doi:10.1016/j.autcon.2022.104487.

[31] D. Ioannidis, D. Tzovaras, C. Malavazos, Occupancy and business modeling, in: eWork and eBusiness in Architecture, Engineering and Construction - Proceedings of the European Conference on Product and Process Modelling 2012, ECPPM 2012, 2012, p. 919 – 930.
URL https://www.researchgate.net/publication/235412763

[32] R. Heikkilä, M. Marttinen, Development of BIM based rehabilitation and maintenance process for roads, in: ISARC 2013 - 30th International Symposium on Automation and Robotics in Construction and Mining, Held in Conjunction with the 23rd World Mining Congress, 2013, p. 1216 – 1222. doi:10.22260/isarc2013/0136.

[33] J. Clarke, J. Cockroft, A. Cowie, J. Hand, N. Kelly, A. Samuel, D. Tang, On the High-Resolution simulation of building/plant systems, in: 14th International Conference of IBPSA - Building Simulation 2015, BS 2015, Conference Proceedings, 2015, p. 90 – 97. doi:10.26868/25222708.2015.2327.

[34] G. Gourlis, I. Kovacic, Energy efficient operation of industrial facilities: The role of the building in simulation-based optimization, in: IOP Conference Series: Earth and Environmental Science, Vol. 410, 2020. doi:10.1088/1755-1315/410/1/012019.

[35] G. Gourlis, I. Kovacic, A holistic digital twin simulation framework for industrial facilities: BIM-based data acquisition for building energy modeling, Frontiers in Built Environment 8, cited by: 6; All Open Access, Gold Open Access (2022). doi:10.3389/fbuil.2022.918821.

[36] F. Meng, Q. Wang, Y. Liu, Y. Zhang, Design and BIM-based technology on a new cross-sea bridge in Xiamen, China, Proceedings of the Institution of Civil Engineers: Civil Engineering 176 (3) (2023) 121 – 127. doi:10.1680/jcien.22.00229.

[37] M. Eguaras-Martínez, C. Martín-Gómez, M. Vidaurre-Arbizu, T. Brennan, S. Krinidis, D. Ioannidis, D. Tzovaras, Architectural simulation of the integration of Building Information Modelling (BIM) & Business Process Modelling (BPM), in: Proceedings of the 10th European Conference on Product and Process Modelling, ECPPM 2014, 2015, p. 805 – 811. doi:10.1201/b17396-130.

[38] Z. Wang, T. Bulbul, J. Lucas, A case study of BIM-based model adaptation for healthcare facility management - Information needs analysis, in: Congress on Computing in Civil Engineering, Proceedings, Vol. 2015-January, 2015, p. 395 – 402. doi:10.1061/9780784479247.049.

[39] M. Xu, D. Peng, Pyrosim-based numerical simulation of fire safety and evacuation behaviour of college buildings, International Journal of Safety and Security Engineering 10 (2) (2020) 293 – 299. doi:10.18280/ijsse.100218.

[40] V. Popov, V. Juocevicius, D. Migilinskas, S. Mikalauskas, Application of building information modelling and construction process simulation ensuring Virtual Project Development concept in 5D environment, in: ISARC 2008 - Proceedings from the 25th International Symposium on Automation and Robotics in Construction, 2008, p. 616 – 624. doi:10.3846/isarc.20080626.616.

[41] M. Patel, V. S. Rajus, G. Wainer, Software Architecture For Integrating Devs Simulation Into BIM, Proceedings of the 2022 Annual Modeling and Simulation Conference, ANNSIM 2022 (2022) 670–681doi:10.23919/ANNSIM55834.2022.9859473.

[42] A. H. Behzadan, C. C. Menassa, J. L. Tishman, A. R. Pradhan, Enabling real time simulation of architecture, engineering, construction, and facility management (AEC/FM) systems: a review of formalism, model architecture, and data representation., ITcon Vol. 20, pg. 1-23, http://www.itcon.org/2015/1 20 (1) (2015) 1–23.
URL http://www.itcon.org/paper/2015/1

[43] A. M. Law, Simulation Modeling and Analysis, 5th Edition, McGraw-Hill Education, New York, 2015.

[44] R. van der Ham, salabim: discrete event simulation and animation in Python, Journal of Open Source Software 3 (27) (2018) 767. doi:10.21105/joss.00767.

[45] ISO, ISO 16739-1:2018 - Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries — Part 1: Data schema (2018).
URL https://www.iso.org/standard/70303.html

[46] A. Mukherjee, N. Moretti, Y.-C. Chan, G. Yilmaz, M. Sasidharan, J. Merino, Z. Rosun, C. Carr, D. McFarlane, A. Kumar Parlikad, Healthcare Process and Variable Mapping For Identifying Digitalisation Opportunities and Building a Hospital Digital Twin, in: EC3 CIBW78 Joint Conference 2023, 2023. doi:10.35490/EC3.2023.270.

[47] 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 the assets (2020).
URL https://www.iso.org/standard/75109.html

[48] W. Kritzinger, M. Karner, G. Traar, J. Henjes, W. Sihn, Digital Twin in manufacturing: A categorical literature review and classification, IFAC-PapersOnLine 51 (11) (2018) 1016–1022. doi:10.1016/j.ifacol.2018.08.474.

[49] A. Mafi, NHS trusts are told to prepare to evacuate hospitals with crumbling concrete (2023). doi:10.1136/bmj.p2045.

[50] T. Uras, S. Koenig, An empirical comparison of any-angle path-planning algorithms, Proceedings of the International Symposium on Combinatorial Search 6 (1) (2021) 206–210. doi:10.1609/socs.v6i1.18382.

[51] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, P. Wang, Path planning techniques for mobile robots: Review and prospect, Expert Systems with Applications (2023) 120254.

[52] Y. Van Tendeloo, H. Vangheluwe, DEVS: Discrete-Event Modelling and Simulation for Performance Analysis of Resource-Constrained Systems, Springer International Publishing, 2020, pp. 127–153. doi:10.1007/978-3-030-43946-0_5.

[53] P. J. Haas, Colored Stochastic Petri Nets, Springer New York, 2002, pp. 385–445. doi:10.1007/0-387-21552-2_9.

[54] L. Capocchi, J. Santucci, B. Poggi, C. Nicolai, DEVSimPy: A collaborative Python software for modeling and simulation of DEVS systems, in: 2011 IEEE 20th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, IEEE, 2011. doi:10.1109/wetice.2011.31.

[55] A. Zimmermann, Modelling and Performance Evaluation with TimeNET 4.4, Springer International Publishing, 2017, pp. 300–303. doi:10.1007/978-3-319-66335-7_19.

[56] M. Paolieri, M. Biagi, L. Carnevali, E. Vicario, The ORIS tool: Quantitative evaluation of non-Markovian systems, IEEE Transactions on Software Engineering 47 (6) (2021) 1211–1225. doi:10.1109/tse.2019.2917202.

[57] C. Ramonell, R. Chacón, H. Posada, Knowledge graph-based data integration system for digital twins of built assets, Automation in Construction 156 (2023) 105109. doi:10.1016/j.autcon.2023.105109.