# Integration of Building Information Modeling (BIM) and Process Mining for Design Authoring Processes

Sobhan Kouhestani

A Thesis

in

The Department

of

Building, Civil, and Environmental Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science in Building Engineering at

Concordia University

Montreal, Quebec, Canada

March 2019

© Sobhan Kouhestani, 2019

**Concordia University**

**School of Graduate Studies**

This is to certify that the thesis prepared

By: Sobhan Kouhestani

Entitled: Integration of Building Information Modeling (BIM) and Process Mining for Design Processes

and submitted in partial fulfillment of the requirements for the degree of

**Master of Science (Building Engineering)**

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final Examining Committee:

________________________________Chair

*Prof. Osama Moselhi*

________________________________Examiner

*Prof. Amin Hammad*

________________________________Examiner

*Dr. Fuzhan Nasiri*

________________________________Supervisor

*Dr. Mazdak Nik-Bakht*

Approved by ________________________________________________________

Chair of Department or Graduate Program Director

March 2019 ____________________________________________________

Dean of Faculty

# ABSTRACT


**Integration of Building Information Modeling (BIM) and Process Mining for Design Processes**

**Sobhan Kouhestani**

Building Information Modelling (BIM) corresponds to the generation and management of the digital representation for building products by wrapping building elements and their information in a unique source file. Open BIM, relying on platform-independent standards, such as IFC (Industry Foundation Classes), is supposed to increase the interoperability in the BIM environment. BIM, as a shared work platform in AEC (Architecture, Engineering and Construction) industry, can be upgraded to act as an Enterprise Resource Management (ERM) system and support data mining for the management of design and construction processes. ERM systems rely on transaction data, also known as "event logs". eXtensibile Event Stream (XES) is an XML (Extensible Markup Language) schema aiming to provide a format for supporting the interchange of event logs. XES-based Event logs commonly include some semantics (called extensions) regarding events.

This work aims to enable BIM to act as an ERM system. To realize this goal, four research objectives were defined and achieved. First, an 'IFC archiver algorithm' was developed to take snapshots, on a regular basis, from different stages of building modeling process (performed in Autodesk Revit), throughout the design phase from start to the end. Second, an 'IFC logger algorithm' was created to consecutively compare archived IFC files, detect design activities and save them in the CSV format event log. Then, XESame module is used to map the CSV format event log to the appropriate data format for Process Mining (i.e., XES format event logs). The activities were categorized in five classes: Addition, Removal, Rotation, Relocation of elements (e.g., a wall), and changes in their properties (e.g., the size, type or family of an element). Five attributes for each activity were stored in the database. Those included: Element ID, Designer, Element Name (Name of the Activity), Start and End time of each activity. Third, Process Mining

techniques were used to detect the as-happened processes. Last but not least, Process Mining helped to derive different types of design process information (analytics) such as social networks of actors, bottlenecks of processes and process deviations.

Two case studies were performed to validate and verify the research methodology. Around 300 and 30,000 events were captured respectively, during the design phase of our first and second case studies. Then, the activity log was fed to a Process Mining tool to mine the as-happened design processes. Two levels of process maps were discovered: As-happened level 2 and "level 3" BIM maps. As-happened maps were derived and represented in Petri net and process tree formats. Moreover, different types of animations of the as-happened design processes were derived for level 2 and "level 3" BIM maps from replaying the event logs on top of the captured processes. Those animations showed project paths, activities queue lengths and service times.

In a nutshell, the study successfully applied Process Mining on the foundation of BIM (as an ERM system) and accordingly made discovery, monitoring and optimizing BIM processes possible. The present study aims to assist BIM and project managers by enabling BIM as a management tool for design processes. These processes are important, because the design phase is at the early stage of every construction project.

# ACKNOWLEDGMENT

I would like to offer my special thanks to Dr. Mazdak Nik-Bakht, my research supervisor, for his patient guidance and enthusiastic encouragement during the planning and development of this research effort. Also, I highly appreciate his eagerness to give his time so kindly.

I am grateful to my mother and father Sousan and Hassan, who have supported me financially and emotionally in my life. I am also grateful to my brother and friends who have backed me up whenever I needed help.

I am also grateful to the members of my committee Prof. Osama Moselhi, Prof. Amin Hammad, and Prof. Fuzhan Nasiri for their valuable time and effort.

Moreover, I acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) and Concordia University, Gina Cody school which funded this study through RGPIN-2017-06697 and graduate research fund.

Last but by no means least, I am also thankful to everyone who had an impact in this project, specifically my colleagues in Compleccity research group Arash Hosseini and Abdelhady Ossama Hosny.

# LIST OF CONTENTS

## TABLE OF FIGURES

# TABLE OF TABLES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| Application Programming Interface | (API) |
| Architecture, Engineering, Construction, Owner Operator | (AECOO) |
| Architectural, Engineering, Construction | (AEC) |
| BIM Execution Plan | (BxP) |
| buildingSMART Data Dictionary | (bSDD) |
| Building Information Modeling | (BIM) |
| Business Intelligence | (BI) |
| Business Process Management | (BPM) |
| Business Process Model and Notation | (BPMN) |
| Comma-Separated Values | (CSV) |
| Construction Document phase | (CD) |
| Construction-Operations Building information exchange | (COBie) |
| Data Mining | (DM) |
| Design Development phase | (DD) |
| Enterprise Resource Management | (ERM) |
| Enterprise Resource Planning | (ERP) |
| Event-Driven Process Chains | (EPCs) |
| EXPRESS Evaluation System | (EvaSys) |
| Facility Management | (FM) |
| Information Delivery Manual | (IDM) |
| Information Technology | (IT) |
| Integrated Project Delivery | (IPD) |
| International Framework for Dictionaries | (IFD) |
| Research Objective | (RO) |
| Process Mining | (PM) |
| Return On Investment | (ROI) |
| Schematic Design phase | (SD) |
| Stochastic Activity Graph | (SAG) |
| Workflow Management | (WFM) |

# Chapter 1
# INTRODUCTION

This research study aims to develop a solution and implement it in practice to automatically detect and evaluate work processes in the design phase of construction projects. The solution captures the required data from the BIM environment and uses “Process Mining”, to discover and analyze processes. I developed algorithms to convert sequences of IFC (Industrial Foundation Classes) files to XES (the standard format used in Process Mining). This upgrades BIM to act as an ERM (Enterprise Resource Management) system for the design phase. The following sections of this chapter cover the motivation, problem statement, objectives, significance and impact, and layout of the thesis.

## 1.1. Motivation and Background

“The separation and lack of collaboration and communication among designer and builder teams (aka silo effect) is known as one of the main challenges in construction projects. This has roots in the industry including the contractual and organizational structures, which traditionally urges each party to optimize their operation processes regardless of the project performance [1]. Efforts to resolve this issue have resulted in modern project delivery methods (such as Integrated Project Delivery – IPD) and using Building Information Modeling (BIM). Over the past decade, BIM has evolved from a digital product to a “process” and a “culture” in the domain of AEC/FM (Architectural, Engineering, Construction/Facility Management), during the recent years. As a semantic archive of building components, BIM can accommodate a wide range of information, not only from the physical and conceptual elements but also regarding project phases, dependencies, impacts, actors, etc. These capabilities assist the management of ever-increasing complexity in construction projects by supporting integrated delivery methods (such as IPD).” [2]

“A process can be defined as a sequence of activities having a common objective (process objective). Execution of activities generate “events”; the start and end of activity are considered two events. Process Mining is the science of discovering the as-happened, end-to-end processes from the analysis of recorded data from events collected and stored in "event logs". An event log consists of "traces" (aka cases), each of which may contain one or more events for execution of activities from the beginning through the end. Event logs include instances (rows of data, i.e., events) and variables (columns of data, i.e. "features"). Variables (attributes) of executed events are also stored in event logs and could have numerical (e.g., timestamp) or categorical (ordinal or nominal) nature (e.g., resources associated with events) [10]. Figure 1 provides a schematic hierarchical view of the event log and the associated concepts introduced. Process Mining is, in fact, the analysis of event logs to learn the as happened process models (known as "process discovery"); detect where the as happened is different from as planned (known as "conformance checking"); and re-engineering and optimism business processes (known as "process enhancement")." [2]

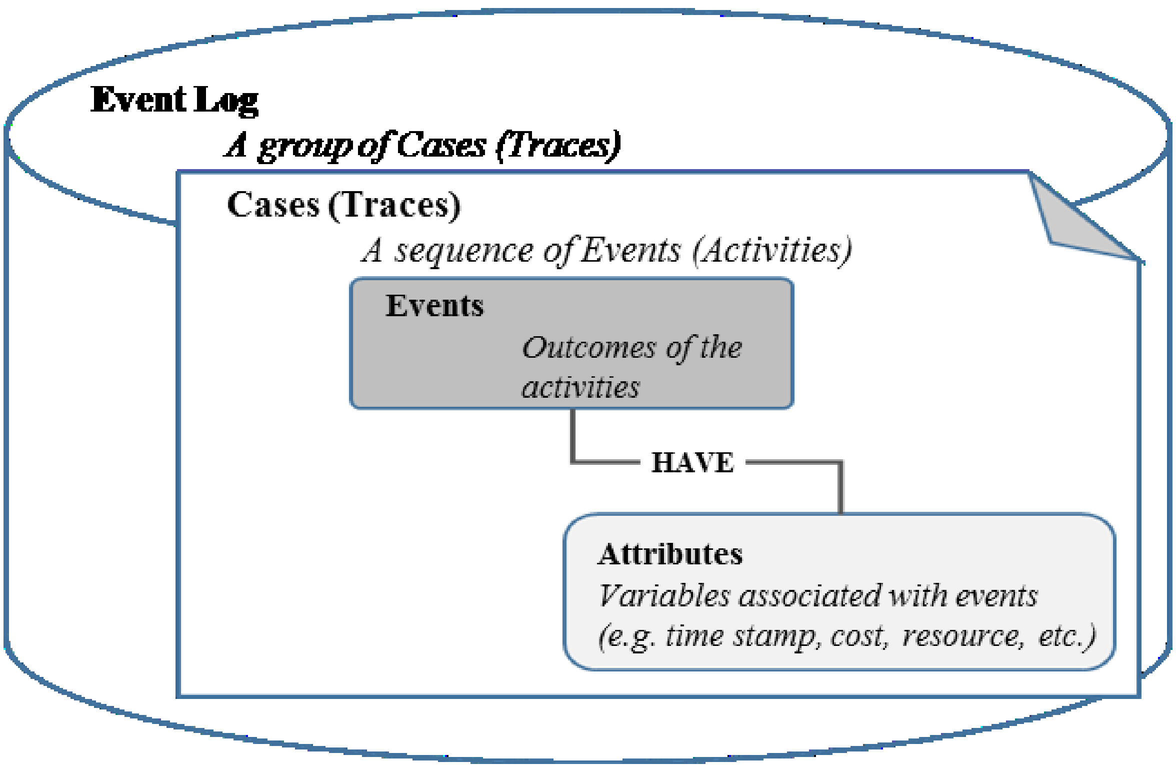



**Figure 1.** Event logs, Cases, Event and Attributes [2]

Although mathematical equations and models can predict and model machines or people behaviors in an assembly line accurately, it does not suffice to model people working in multiple processes with a diverse range of priorities [3]. “This has created a new discourse in AEC/FM, called BIM Management, with roles such as BIM manager or BIM coordinator, and tools such as BIM Execution Plan (BxP). The primary role of BxP is providing a protocol for successful BIM adoption in construction projects through defining the expectations (BIM goals and BIM uses); planning for achieving those expectations (BIM processes); and maintaining the timeliness, level of details required, and interoperability for the information flow among the involved parties. BxP templates, such as the CIC BIM Project Execution Planning guide – a buildingSmart alliance project, introduce two levels of template BIM processes to be designed in projects; level 1, as an overview of the entire project and its BIM uses, and level 2, as a default for each BIM use in the project. These process maps serve as the backbone of BIM implementation and are also critical from contractual and IT infrastructure points of view [4].” [2]

Planning is performed with information about the design process. Information is an outcome of experience from the planner (aka BIM manager) and the design company. Experience is gained by performing and doing the design. So far, no monitoring tool can effectively get insights about the processes happening in the design authoring BIM use. Design Authoring is “a process in which 3D software is used to develop a Building Information Model based on criteria that is important to the translation of the building's design. Two groups of applications are at the core of BIM-based design process are design authoring tools and audit and analysis tools. Authoring tools create models while audit and analysis tools study or add to the richness of information in a model. Most of audit and analysis tools can be used for Design Review and Engineering Analysis BIM Uses. Design authoring tools are a first step towards BIM and the key is connecting the 3D model with a powerful database of properties, quantities, means and methods, costs and schedules.” [4] As a result, they do not have the required information to know about the causes of project time-consuming steps or activities (aka bottlenecks) and procedure deviations. Quirijnen and van Schaijk (2013) stated that parties involved in construction projects might feel that something is wrong with the project and guess the reasons, and if they know the reason, they do not have any idea about how to improve defected processes. Hence, it leads to guess-based problem solving rather than information-based decisions. In 2013, Interviews with practitioners conducted by Quirijnen & van Schaijk revealed that there are monitoring systems in the construction firm capturing events based on human-written notes [5]. It was perceived that despite the importance, the current monitoring systems are costly, time-consuming and error-prone. Thus, practices of monitoring systems are rarely performed which makes it a necessity to move towards automation to decrease the errors in capturing data and managerial decisions. Nonetheless, monitoring performance research does not have the minimum requirements to empower control related studies [6].

Process mining studies aim to find an applicable solution for improving processes based on real-world data. It provides a method to gain knowledge and decrease flaws in processes in a diverse range of domains. Basically, the goal of Process Mining is to get event data, gain insights about the actual process and, finally, discover it. Often, organizations unearth process issues based on imaginations or rule of thumb rather than data-driven information. In 2011, Van der Aalst explained Process Mining as an evolving field of study contributing to several groups of tools to provide evidence-based information and to provide process improvements [7]. In contrast to other data mining techniques, such as Business Intelligence (BI) and Business Process Management (BPM), Process Mining delivers a comprehensive understanding of as-happened and end-to-end processes. BI emphasizes on documents and reports rather than transparent processes. BPM depends on the experts who are involved in modeling the as-planned processes. This type of processes does not help organizations to find out the real as-happened processes [7].

Process Mining highly relies on using appropriate event logs. Often it may be a challenge to extract and save the event logs from a diverse range of data sources including databases, message logs, flat files, ERP (Enterprise Resource Planning) systems, transaction logs, and document management systems. However, most of these data sources are utilized in the construction industry, every company has a unique and different software systems and data structure. Altogether, companies do not use these data sources for event logs creation, but for running their daily procedures. Thus, organizations do not understand that they are creating precious data worthy of investigation. On the one hand, the observing and monitoring data are not structured enough and thereby not proper for event logs generation. On the other hand, different companies and non-propriety organizations such as buildingSmart agree to develop a more structured and independent BIM format (aka IFC) to add, extract, store and re-use information. Thus far, IFC is under development and does not have the required semantics and syntax to store process information and event logs. Two main reasons may explain this limitation in IFC: (a) the fact that adding these process data to models is time-consuming and effortful, and (b) value of this type of data (event log) is still not understood and appreciated by the researchers and developers. These event logs may cause insights and creates significant opportunities and open an uncharted territory of valuable information for construction project management. Not having the required syntax in IFC explains the reasons why people do not accept the challenges to save and track data for monitoring purposes consistently. Despite the inability of IFC to store the required data for Process Mining, it is possible to use its existing BIM data to extract the right event logs for this purpose.

As discussed above and described in the background literature, there are three main reasons making this study and enabling BIM as a potential monitoring and management system for design phase legit (a) the attention in planning and management of construction projects and specifically in monitoring of design processes, (b) the inadequate quality of current planning and management tools and the unfeasible and imprecise manual process maps, and (c) the capacity to independently generate, extract and save design related data with new BIM technologies and tools. Accordingly, this research will explore the potentials of improving design procedures within the construction field with the aid of BIM and Process Mining. Furthermore, this study provides a solution consisting of three parts: (i) Archiver: an automatic IFC archiver to take snapshots of developing BIM models throughout the design phase (from the beginning to the end). (ii) IFC logger: a tool-independent event data logger which ran through the produced IFC archive and produce the right event log for Process Mining tools. (iii) perform design process analytics over the produced event logs with Process Mining algorithms and getting managerial-level insights about the design processes.

## 1.2. Problem Statement

"Successful applications of Process Mining in process discovery, conformance checking, and process enhancement can be seen in a broad range of industries (including healthcare, manufacturing, construction, oil and gas industry, silos management, delivery process, customer service, and finance [8]). Also, some applications are reported in infrastructure-related workflow management (such as airport management [9] or urban management services [10]). Most of such applications depend on the information exchange standards specifically developed to analyze event logs." [2]

The truth is that the AEC industry is in desperate need of Process Mining applications; time and cost deviation is among the most obvious challenges of the AEC industry. So far, there is no such information architecture and right feed for Process Mining defined within the AEC industry. Industry Foundation Classes (IFC) is the independent BIM data structure, which is used by most of the AEC software to exchange the information, cannot serve as a right database for data mining and Process Mining.

There are some central questions for this research which I used as a basis for this project:

Q1: How design processes can be monitored and improved?

A1: To shorten the duration of the design processes it is required to understand how they are characterized and predefined. To answer this question, literature for BIM process maps, process modeling and management, and Process Mining was surveyed.

Q2: What kind of data is required for analysis of design processes?

A2: The right type of data which is proper for capturing the design process is examined from two perspectives. First, it should be available naturally within the design process. Second, it should be converted to an appropriate data structure which can be used by Process Mining techniques.

Q3: How Process Mining methods help to detect and improve bottlenecks and planning deviations?

A3: I elaborate on the answer by stating how algorithms work.

Q4: How process-related data can be linked to BIM elements?

A4: Information within BIM standard data format is used to produce Process Mining (PM) standard data. In this process we extract the required data from BIM standard data format including element IDs.

Q5: What are the necessary tools to capture and work with the required data?

A5: To answer this question I will introduce and use the following tools:

- BIM design authoring software, Revit
- Sublime, Dynamo, and Python: data archive, analysis, and reconfiguration tool
- ProM: the main process analyzing software tool

## 1.3. Objectives

The goal of this research is "Enabling BIM to act as an ERM system for design authoring processes". Scope of this project is limited to the process discovery and analysis related to design authoring BIM use. BIM is widely used in design authoring phase of construction projects which makes it an excellent candidate to track real-life information about design processes. In this research, BIM-based data is archived and analyzed to gain insights and address the problems in design processes such as losing corporate memory from one project to another, time-consuming activities, and team collaboration issues.

To address the problems stated in the previous section, the Research Objectives (RO) of this article are categorized as:

Research Objective #1 (RO 1): Archiving snapshots (with IFC format) from different stages of the design authoring process on a regular basis;

Research Objective #2 (RO 2): Detecting changes by comparison of consecutive IFC files and extracting their differences. Then, mapping the discovered differences into the XES format as activities;

Research Objective #3 (RO 3): Discovery of the design authoring process using Process Mining techniques;

Research Objective #4 (RO 4): Gaining design related information using Process Mining techniques.

## 1.4. Significance and impact

"On the other hand, the rich source of information collected, archived, and contextualized in BIM, provides significant opportunities for distilling data-driven intelligence in projects. Among other benefits, such data analytics can reveal underlying patterns of collaboration among project teams and can provide BIM managers assistance with project planning and control. The present research aims to use the data organically generated (and in most cases wasted) during design and construction phases, for measuring the performance of project teams in the implementation of BIM uses. More specifically, this study reports on using "Process Mining" to discover and evaluate "as happened" processes (versus "as-planned" ones recorded in the BxP guide). We metaphorically refer to such processes as level 2 and "level 3" maps, since they will provide the BIM Manager with a closer, more in-depth, and more case-specific view on the BIM processes executed by the project team. In this regard, "Design Authoring", as a central BIM use in most BIM projects is selected to be studied more closely." [2] This BIM goal is also important because it is situated at the early stages of the project and improving even a small portion of it may result in a considerable saving of cost and time [2].

## 1.5. Layout of the thesis

"Despite extensive applications of Process Mining in different domains; AEC/FM has not taken advantage of this powerful family of techniques, which first and foremost has roots in the lack of availability of event logs (which generally come from the ERMs) in this domain. BIM can potentially archive the digital footprints of processes in design, construction and operation phases of building projects. These footprints can provide an opportunity for upgrading BIM to act as an ERM in the AEC/FM industry. The present study reports on early steps of an extensive attempt in this regard. The paper reports on acquiring event logs from BIM, and then using them in Process Mining to discover and analyze as-happened BIM processes." [2] After a literature review on the works done inside and outside AEC/FM for discovery, modeling, and management of processes in chapter 2 (literature review), a BIM-process mining integration model is introduced in chapter 3 (methods) through creation of IFC archiver algorithm for design authoring processes and IFC logger algorithm for logging event logs from analysis of the archived IFC files. These tools are validated and verified by two case studies for the discovery of design authoring processes, details of which are discussed in chapter 4 (implementation). Finally, the concluding remarks, as well as the ongoing and future research in this area are highlighted in chapter 5 (conclusion).

# Chapter 2:
# LITERATURE REVIEW

This chapter provides a literature study on Process Mining, BIM and Open BIM, sequentially. Process Mining techniques try to gain information by analyzing the repeatability in products and processes which can be insightful for decision makers in every industry. In the first section, I elaborated the concept of Process Mining, its applications, and samples of its industrial case studies. Since this field of study is new in Building Engineering, more details are provided for the concepts such as Process Mining tools, algorithms, and applications. The AEC industry has problems in capturing, understanding and replicating the existing end-to-end processes [67]. BIM implementation in the context of process modeling and management tries to improve workflow procedures by the elimination of wasteful practices within AEC firms. For the second part, a brief overview of BIM is provided and more emphasis is put on introducing BIM as a process and BIM management, i.e., managing BIM as a process. At the end of this chapter, the Open BIM standards, mainly IFC and its MVDs are explained.

## 2.1. Process Mining

This section first briefly introduces the concept of Process Mining. Then, a holistic view on its available techniques and their limitation would be provided which gives a conclusion that the inductive miner algorithm is the best option to mine processes for the current research scope. Moreover, it is followed by potentially available applications of Process Mining. In the end, some Process Mining case studies applied within different industries are explained.

### 2.1.1. Process modeling and management

A process is a sequence of activities (or steps) which are taken by the performers in order to satisfy a particular objective [12]. A business process is a type of process with an organizational goal such as the delivery of a product or service to a client. Petri net is a process modeling language which models the concurrency and controls the flow and does not elaborate on all other perspectives, such as data-flow, time, resources, costs, risks. Regardless of simplicity, many analysis techniques can use Petri nets to analyze them [13] [14]. Petri net is a foundational process notation. Its network is static and consists of places and transitions and tokens. Tokens could be held in places and be produced or consumed in transitions. States in Petri net are called marking. Markings can be reachable or not reachable. A transition is enabled when each of the input places contains tokens (the required number of tokens is the same as the number of arrows exits the place). Firing means consuming and then producing tokens to each output place based on the number of arrows going out of transition. Sometimes, Petri nets may have some limitations to fit the reality well.

BPMN is a discipline which applies combined knowledge from information technology and management sciences to operational business processes [15, 16, 17]. BPMN is considered as an extension of Workflow Management (WFM). WFM deals with business processes automation [18, 19] while BPMN focusses on the management of processes and organization of the work which include a broader scope. Workflow nets (WF-nets) are a subclass of Petri nets often used in the context of workflow management and business process management (is the linking pin between IT and management science). In the WF-nets should be free of apparent anomalies (should be sound). Soundness is a notation which is used for a model that is safe, live, have a proper completion, or an option to complete. Short-circuited is a transition in Petri net between end place and the start point of the Petri net. A WF-net is sound if and only if the corresponding "short-circuited" Petri net is live and bounded. The functions of models in Process Model (BPM)/WF Management (WFM) is that one can know about the reason of the processes, know where the bottlenecks are and how can make them cheaper; bottleneck is a sequence of steps in a chain of processes which limits the capacity of the whole chain. There are two main categories of model-based analysis in BPMN or WFM: verification (e.g., soundness checking) and performance analysis (e.g., simulation). However, there is some limitation in these types of analysis: they rely on the availability of high-quality models; they often have poor alignment between hand-made models and reality; and when the models and reality do not match each other, model-based analysis does not make much sense. Process models may be used to discuss responsibilities, analyze compliance, predict performance using simulation, and configure a WFM/BPM system.

There are different languages which can be used for representing models such as Business Process Model and Notation (BPMN), Event-Driven Process Chains (EPCs), Petri nets (often as a subclass of workflow nets). For the sake of this research, the selection between these languages should be based on two features; first, finding a model that illustrates different aspects of the process adequately such as decision points. BPMN is an appropriate model which is able to show decision points. Second, visualization must be understandable (not having extra information) and show the desired processes for the end users. The simplicity of Petri nets makes it a decent candidate for this project. The BPMN and Petri-Net models- which are used in ProM to present process models- are descriptive enough to satisfy the requirements for illustrating design authoring processes of the construction projects.

### 2.1.2. Process Mining rudiments

The critical part of having a process model, which is a necessity for business management, is to obtain a required business process data source. Every day in our daily life people produce data directly (e.g., Buying coffee or refueling your car) or indirectly (e.g., the sensors in the mobile). These raw data (event data) are related together via Internet which called Internet of Events [20]. There are four primary sources for Internet of event data: Internet of Content such as Google or Wikipedia, Internet of People such as Twitter or Facebook, Internet of Things (Devices which are connected to the internet), and the Internet of Places such as our phone which records the location of people. Examples of data sources are a Comma-Separated Values (CSV) file, a database system, a transaction log, a business suite/Enterprise Resource Planning (ERP) system, a message log, or an open API (Application Programming Interface).

In recent years, the exponential growth of producing data makes the internet of events concept more interesting. The concept of Big Data refers to this massive amount of data [21]. Big Data has four aspects: Volume, Velocity, Variety, and Veracity (Validation) of produced and producing data. While there are still difficulties in a small group of data, these four characteristics of Big Data result to even more problems. Hence, data scientists must know how to collect, analyze and interpret data either big or small. Therefore, data scientists typically have to answer four main questions regarding data: “what happened?”, “why did it happen?”, “what will happen?”, “what is the best that can happen?”. They use data mining techniques to answer these questions. Data mining is defined as “the analysis of (often large) data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner” [22]. For answering these types of questions, data scientists need to have a series of knowledge;

Data Mining (DM) and Process Mining (PM) are among them. In Process Mining, end-to-end processes are being investigated; law level data or isolated patterns are not in its scope. Process Mining techniques tend to interpolate event data and process models; it works like a glue between process model analysis and data-oriented analysis. Therefore, these techniques answer some unanswered question such as “what people actually do?”, “where are bottlenecks in the processes?”, and “when and where do deviations from the model happen”.

Moreover, Process Mining provides missing links between performance-oriented model analysis and data-oriented analysis [23]. Also, it aims to answer performance oriented and compliance analysis questions. Process Mining works as a glue to connect Data and processes, Business and Information Technology (IT), Business Intelligence (BI) and Business Process Management (BPM), and performance and compliance; it can be done in runtime or design time. In the BI tools, the problem is that they are just showing the spreadsheets and graphs and sets of numbers; it might be incorrect to capture reality based on just these numbers. Whereas, PM discovers the desired line and what people “really” do. Process discovery can be compared to learning a new language. If sentences are traces in an event log, the language would be the process model. Spell checking in the Office software is similar to the conformance checking; one will type real data, and the software will check it out with the earlier models, and for example, it can discover that it would be incorrect or correct based on the correct model.

*Data and Process Mining-* The starting point for Process Mining is event data. Characteristics of an instance in an event log are case id, activity name, timestamp, and resource. There are three types of relationship between event log and its model: Play out, Play in, and Replay. When the process model exists, and the generation of event log happen, it would be called a play-out action. For example, simulation, workflow system creation based on the process model, management games, model checking, and all the other behavior that generates data based on the process model are among play-out actions. For play-in, there is a reversed story; it starts with behavior (raw data) and then a model based on that would be generated. Process discovery based on real behavior is an appropriate example of this action. In the replay, the model and the behavior exist. This action is for aligning discovered and observed behavior. In other words, it is a confrontation between model and reality which is the most essential form of Process Mining [24].

In summary, Process Mining and Data Mining have some similarity and differences: they both start from data; DM is not process-centric; process discovery, conformance checking, and bottleneck analysis cannot be addressed by traditional DM techniques; end-to-end process models and concurrency are essential for Process Mining; event logs are different such as having timestamps that refer to cases. Also, one can combine both of them to answer advanced questions. Ideas for Data Mining techniques can be adopted in Process Mining. For example, if there is a choice in a discovered process model, by having the control flow, decision tree learning can be used to understand the causes for decisions in the process model.

*History of Process Mining-* The pioneers of the field of Process Mining are Cook et al. (1996) [25] [26]. Their contribution led to the creation of three algorithms: KTail, RNet, and Markov. Agrawal et al. [27] contributed to the field by finding the first Business Process Model and Notation (BPMN)-based Process Mining algorithm (1998). Hebst et al. developed a workflow called Stochastic Activity Graph (SAG), where it was a directed graph and nodes were referred to activities and edges where decorated with probabilities (1998). The idea of assigning two events (start and end) to each activity was introduced by S. Y. Hwang and W. S. Yang (2002) [28]. For the first time "trace" used by G. Schimm [29] [30] [31] as a set of events. Also, other definitions of events such as different life cycles of an activity where used by him (2002).

Van der Aalst et al. [32] [33] developed the "Alpha" algorithm to discover processes out of event logs and represent them in Petri nets format (2002). The limitation of this algorithm was the lack of activity frequency analysis; it was a huge error in real-world applications. Afterward, M. Golani and S. Pinter [34] [35] presented an approach that considered activities as non-instantaneous events and had start and end events in their lifecycles; their work was close to the work of Agrawaki et al. (2003). They also introduced the dependency of activities to each other which is the same as the Finish-to-Start concept in project management terms in construction. An extension of Van der Aalst work was introduced by Weijters et al. [36] which is called "heuristic miner" algorithm. In this approach, the authors tried to determine the sets of relations from the log and create a model based on them (2003). The novelty of the work was the statistical measures which were used in the identification of activity relations in event logs. G. Greco et al. [37] [38] tried to create a hierarchy of process models with different abstraction levels (2004). In this approach, model generation is happening for each log trace; then it would be aggregated as a final model. Ana Karla

Alves deMedeiros and Wil van der Aalst [39] [40] used genetic algorithms in which a wide range of processes where produced and then analyzed to check if it fits the related log or not (2005). Gunter et al. contributed to this topic by handling “spaghetti processes” (2007). Spaghetti processes are known based on a high level of flexibility and low level of structuredness.

### 2.1.3. Process mining Algorithms

This section, first, introduces different process models and mention the reasons why BPMN and petri-net are selected for the scope of this research. Then a review of five Process Mining algorithms and their advantages and disadvantages is conducted. In the end, the most appropriate algorithm is chosen to satisfy the requirements of this project.

*Alpha Algorithm-* Process discovery focuses on play-in action. Alpha algorithm is part of a process discovery family, called Direct Algorithmic Approaches, which captures footprints of events and uses it, automatically, to construct a process model; it is the first algorithm that can discover concurrency, loops, parallel parts, choices while guaranteeing specific properties [41]. Alpha algorithm only focuses on control flow; so, it just considers activities and ignores all other features of data such as frequencies. On the contrary, Fuzzy Miner [42] and Flexible Heuristic Miner (FHM) [43] algorithms, which are also considered Direct Algorithmic Approaches, consider data frequencies in their method. Alpha algorithm only cares about orders of activities within a particular case; the event log can be converted to traces (sequences of events for the activities in each run throughout the process is called a trace). In general, Alpha algorithm nicely discovers the critical ingredients of process discovery and visualizes the model as a Petri net.

Regarding the discovery of a process model, it is necessary to balance among all four forces: Fitness & Simplicity, Generalization & Precision. In the alpha algorithm, it is required to find relations among activities such as direct succession ($X>Y$), causality ($x \rightarrow y$: $x>y$ and not $y>x$), parallelism ($x||y$: $x>y$ and $y>x$), and choice ($x\#y$: not $x>y$ and not $y>x$) [7]. There are some basic ideas used by alpha algorithm to find patterns in a process including sequence pattern ($a \rightarrow b$), XOR-split pattern ($a \rightarrow b$, $a \rightarrow c$, and $b\#c$), XOR-join pattern ($b \rightarrow d$, $c \rightarrow d$, and $b\#c$), AND-split pattern ($a \rightarrow b$, $a \rightarrow c$, and $b||c$), and AND-join pattern ($b \rightarrow d$, $c \rightarrow d$, and $b||c$). XOR-split and AND-split represent decision points and cloning points, respectively. Join patterns correspond to merge points for different paths in the process. Every trace has a footprint table which is a unique table. It gathers all activities’ relations between every two activities. Finally, the model can be constructed based on this table and the patterns mentioned above via alpha algorithm. The limitations of Alpha algorithm are brought here [7]:

a- Implicit places: are the places if we leave them out the behavior does not change.
b- Loops of length one and two: it addresses the inability of Alpha algorithm for discovery of models having loops of length one and two via alpha algorithm technique; However, this limitation can be solved by data preprocessing.
c- Non-local dependencies: The inability to find a dependency between two activities and at the same time not happening one after another is not specific for the alpha algorithm, and other algorithms suffer the same problem. In this case, allowing the occurrence of some dummy events which does not exist in the main event log fix this issue.
d- Does not allow two analogous transitions: The Alpha algorithm will never create a WF-net with two transitions having the same label. Also, this could make another representational bias which one activity occurs two times:
e- There is no silent transition in the alpha algorithm: it happens when a transition is an optional one.
f- It cannot discover OR-split/join model in this algorithm. OR-split corresponds to a "wait and see" behavior. In other words, the token in the process can go left or right or both at the same time.
g- Another problem is that a discovered model via alpha algorithm might not be sound; for example, might have some deadlocks. Deadlock is a situation in which a trace of the event log starts a process and does not complete it.
h- Noise and incompleteness: To mine an illustrative process model, it is assumed that the log has a large sample behavior of events. However, there would be two challenges. First, event logs may contain infrequent data which are not representative for the whole events log and, finally, for the model. It means that alpha algorithm cannot deal with noise; in other words, it considers both infrequent and frequent data; However, noise is a challenge for other Process Mining methods, as well. Second, the log should include a comprehensive set of events to be able to mine some of the underlying control-flow structures. Lack of a comprehensive event log might drive a more general model than it is required. For example, the event log may result in a more general and underfit model than we might

need; fitness may not be satisfied well. Although, one is able to make a threshold for a minimum number of events needed for making a fit model.

*Two-phase discovery technique*- In Two-phase discovery technique, a transition system is produced based on the event log; it can be done in a way to result in a general or a specific model [44]. Subsequently, concurrency patterns are derived using the theory of regions; then in the second step, the Petri net can be discovered. This technique has some essential features that make it valuable. First, it can be used to discover complex process patterns which cannot be done well by another algorithm such as Alpha algorithm. Second, it provides insights into the essence of process discovery. On the other hand, the first limitation of this algorithm is its Inability to discover some special process constructs; however, it can be handled through extensions of the basic algorithm. For instance, it cannot deal with repetitive transitions and loops with length one. The second limitation is the inability to properly balance the four forces of fitness, precision, generalization, and simplicity which is a more foundational problem.

*Language-based regions algorithm*- The goal of state-based regions algorithm is to find the places in a Petri net. Language-based regions use "language" as input, instead of a transition system, to construct the model [45] [46] [47] [48]. It is, basically, solving a system of inequations of (1). A and A' are matrices representing the log, x and y correspond to the arcs in the Petri net, and c is the number of tokens in the place that we are trying to discover. At a simple definition, the model just says that when the event log is replayed on top of the model, places can never be negative; there can be no negative number of tokens. Any solution to this system of inequations is a region that we can add without removing any behavior from the log.

$$c.1 + A'.x - A.y \geq 0 \tag{1}$$

The first part corresponds to the initialization and represent the number of tokens at the beginning, second part is related to the production (it is based on A'), third part concerned with consumption (it is based on A); that is why the minus sign is behind it, and the last part says that the result should be positive. Finally, there is a set of inequations to be solved; any solution to these inequations is a feasible place in the model.

Another feature of this method is that it is easy to add some constraints (empty places at the end, limited fan-in (number of arcs enters) and fan-out (number of arcs exits). Moreover, there is another property called “Goal Function” which can be used to select the most interesting places. Also, one can solve the optimization problem, for example, using Integer Linear Programming (ILP) technique which is supported by ProM.

*Genetic Process Mining technique-* The general concept of genetic Process Mining technique is to use evolution to continuously improve the model and ending up to the best model that describes the event log [49]. This method is similar to the theory of evolution of life; based on the event log, one randomly creates the first generation of process models. The next step is the quality assessment of the model, and then finding the high-quality candidates to continue with. The best candidate goes to the next step, or a group of best candidates is selected to perform a crossover (mutator) operator to recombine them and create new models; basically, the mutated model is the one to continue with. Over a large number of these mutations, the process of random creation and mutation continues until the best model is reached. However, slowness and not being efficient is among its limitation, and flexibility in discovering the desired model is its strength.

*Inductive mining algorithm-* Inductive mining algorithm decomposes event logs into different parts to find existing types of separation (separation operators such as sequence, XOR-split, parallel, and loop operator) within the structure of processes [50]. After constructing a hierarchical tree, corresponding Petri net or BPMN model can be discovered. This technique is supported by ProM which is a robust Process Mining tool [51]. Also, it guarantees sound process models and mines quite a balanced model in terms of simplicity, generalization, and precision.

There are many more algorithms available to find process models. Regarding implementation, there are some consideration and factors in choosing an algorithm such as speed, memory usage, representational bias, and flexibility. Moreover, algorithm implementation and its approach are two essential issues that should be considered in the procedure of algorithm selection. However, for this research, there are some other criteria which make the inductive mining an appropriate choice compared to other Process Mining algorithms including the ability to discover a balanced process, the ability to handle instantaneous events, the capability to produce the animations of the process.

### 2.1.4. Bringing Event Data into the right format

The emphasis of this part of the report is about getting and preprocessing the event data and eliminating possible data quality problems. Event logs are the input of the Process Mining. Some of the primary sources which can be used to extract event data are ERP system, SAP (Systems, Applications and Products), middleware (such as IBM's WebSphere), pure sources of information such as CSV file, social media (open APIs), or by the tapping of message exchanges [52]. There are many sources of information, and the challenge is to convert them into an event log that can be used for Process Mining.

A conceptual model for event log definition has already illustrated in **Error! Reference source not found.** [2]. It shows that a process consists of cases and each case consists of events. Likewise, events may have attributes such as timestamps, resources, costs, and lifecycle transactions. Also, cases, themselves, can have attributes. This information architecture would be the right kind of data that can feed Process Mining, and there are rare use cases of such information architecture within the AEC industry [53] [54] [55] [2]. The challenge in the way of converting the raw data to our desired type of data is not the syntactical conversion, but are some others including: (1) Locating the relevant data (we need to understand the underlying data in order to identify what data we need.); (2) Scoping; (3) Identifying process instances and Flatten event data (i.e., convert the event data extracted from different resources to a unique one which can be used for Process Mining) and decide what the corresponding process instances are; (4) Data quality problems. There may be some data quality issue in data that make the Process Mining more challenging including missing data, incorrect data, imprecise data, or lots of irrelevant data.

One of the XML-based standard proposed by IEEE is eXtensible Event Stream (XES) [56] which aims to support the event log exchanges between data analysis and ERM systems [57]. XES identifies "extensions" schema to generate supportive semantics to assign attributes to events in XES-based event logs. For example, "Organizational" extension defines the actors of such activities, "Concept" extension provides a schema to store the events name, "Time" extension refers to the time of the event. These extensions are used for structure the event log in the right format to enable Process Mining and process analytics such as the bottleneck, deviation, and social network analysis. Basically, the key to the openness and extensibility of XES are extensions. OpenXES considered as a Java-based reference for implementation of XES standard [58]. It is proposed to be used by developers in multiple fields and aims to extend and enable XES to act as domain-specific solutions [59]. Latterly, a diverse range of commercial and research-based Process Mining tools (ProM [60], Celonis [61], Disco [62], myinvenio [63], and Lana [64] among others) have been industrialized. These application tools analyze event logs (in CSV or XES format) to mine processes, do compliance checking, process optimization analytics. XES format is the primary input of these process analysis tools. Also, an event log conversion from CSV to XES format is possible via a data mapping operation [65]. CSV characteristics of events (columns of the CSV log) can be defined through the extensions which are the mapping elements. Although there are a set of defined standard extensions, some semantics requires the creation of a new extension which can be developed via OpenXES.

### 2.1.5. Process Mining Applications

Different types of Process Mining applications are briefly introduced and reviewed in this section. Conformance checking, enhancement techniques, social network mining, organizational mining, and operational support are elaborated in each section, sequentially.

*Conformance checking-* Conformance Checking aims to compare an event log with its corresponding process model. Conformance checking can be done either offline or online (in runtime). Online one refers to cases when it is required to alert the deviation immediately after they happen. In order to balance four forces- fitness and simplicity, and generalization and precision- in a discovered process model, conformance checking was proposed [7, 24]; fitness is the main thing that needs to be checked in this task [24]. Checking the fitness is done by counting the number of deviations while replaying events on top of the model.

There are three types of diagnostics in conformance checking. The first diagnostic is at the level of an event log while real data does not fit in the model. The other diagnostic is at the level of a model; in other words, part of the model often deviates from the real behavior. The third type is the Global Metrics; for example, the fitness metric has a value between zero in which any number closer to zero corresponds to lower fitness. Hence, there are three use cases for Conformance Checking. The first use case is auditing and compliance. It is used when the model cannot be changed and corresponds to finding the event deviation and the reason for it at the business level (in this case the defected procedure should be modified to mimic the as-planned workflow). Next, evaluating process discovery algorithm regarding four forces which are mentioned earlier (Fitness

& Simplicity, Generalization & Precision). Last, conformance to specification is used when the goal is to refine or improve the model, and it is needed to refine the model to match the events more, for example, software improvement is the reasons of such conformance checking. To sum it up, if there is any deviation, it could be a deviation in the event log or the model. Also, deviations could be positive or negative; for example, breaking the glass of fire alarm system when a fire happens is a positive deviation. There are three main algorithms for conformance checking: Casual footprint, Token Based replay, and Alignment-based conformance checking [7].

*Enhancement*- Enhancement is different from conformance checking. In conformance checking, a log and a model are compared in order to quantify the differences between modeled and observed behavior. On the contrary, enhancement is about improving the existing model. Additional to control flow, enhancing the model, or extending the model with stochastic and performance information, it is possible to implement extra information in the mined model such as resources and times. Also, one can use techniques that are adopted from other areas such as data mining which is used for decision mining in processes. In conjunction with process discovery techniques, operation research techniques for performance analyses such as simulation or queuing theory can be utilized. In the following, a different form of enhancement is discussed to elaborate on the functionality of Process Mining.

Mining decision points are about answering the following question; "why certain cases take a particular route and why other cases take another route". In process models, there are many points that a case can go left or right, called decision points and Mining Decision Points is the act of finding these points [13], [17]. As an input for this act, it is needed to have an event log and process model. Through the notion of alignment, every real trace can be compared to the model. After aligning the model with reality, the goal would be finding the reasons and types of decisions have been made in the model [66]. Then, cases are categorized based on their attributes and decisions. Additionally, one can use decision point analysis in operational support to predict the future.

Discovering guards corresponds to take the output of a decision tree and translating it to a data-aware Petri net. This type of model enhancement can be done by looking at decision points. After decision points, there are a group of activities which are called response variables. Predictor variables would be attributes of the trace such as resource, customer and amount of previous activity [67]. The discovered guards describe what has happened rather than what should happen.

Data-aware Petri net also can be used for conformance checking. One can replay the event log on it and find out where there are deviations between reality and the model. Also, using Process Mining techniques decision tree and Petri net of an event log can be combined to make the data-aware Petri net. Then, Conformance checking can be done by using the data-aware Petri net.

To determine the bottlenecks and their causes, timestamps are required to be included in the event logs to analyze performance and suggest improvements. Basically, there is no need to analyze processes when there is no problem. On the other hand, Process Mining is crucial when there are problems and bottlenecks in a process. So, in this part, the objective is to find out where bottlenecks are and why they are there [11]. The event log can be enhanced and extended in different ways including timestamps; transactional information (start-time and end-time) can provide temporal analysis including the duration of activities (Service time), waiting time in a place, meantime, variation, routing probabilities, and the completion time of each trace. Moreover, we can whether use Gantt chart for view on the timed replay of the cases or for timed replay projected onto resources.

*Social Network Mining*- Different types of social networks can be derived from event logs which have data about resources. People leave lots of trails while involved in different systems. Internet of events has four main categories: internet of contents, people, things, and places. It is also possible to look at what people are doing socially, for example, on Facebook or Twitter. In the future, things will be connected to the internet more and capture human behavior more comprehensively. Hence, in the AEC industry, it is also viable to focus on actors' trails by extracting information from the already available data [2].

Event logs may refer to actual actors executing the activities. Using these resource attributes, one can do a fundamental form of analysis called "Resource-activity matrix". For creating this matrix, it is needed to know the number of times every resource executes each activity. Then, it is required to divide the execution number of a particular activity which a resource has performed by the number of all cases in the event log. In other words, resource activity matrix provides insights to "who is doing what".

Social network analysis is old. In the 1930s, Jacob Levy Moreno used such techniques to assign students to a residential cottage [11]. "Sociometry" is a notation which is used for presenting data on interpersonal relationships in a matrix or graphical form [68]. Another notation is "Arcs" which represent relationships in the social network and has features such as weights or inverted distances. Smaller weights (thickness) or longer arcs correspond to weaker relationships. “Nodes” or “Organizational entity” in these kinds of diagrams correspond to groups of people, resources, roles, and departments. Other notations such as “centrality”, “closeness”, “betweenness” can be used for, respectively, showing most central people, closeness in a group of people, and the extent of being in between nodes [11]. Also, “cliques” is used for people who have many relationships among them as a group and fewer with the rest of the network. Figure 2 is showing all the notations that stated earlier. Size of the oval (node) correlates with the frequency of activity which is done by a particular entity.

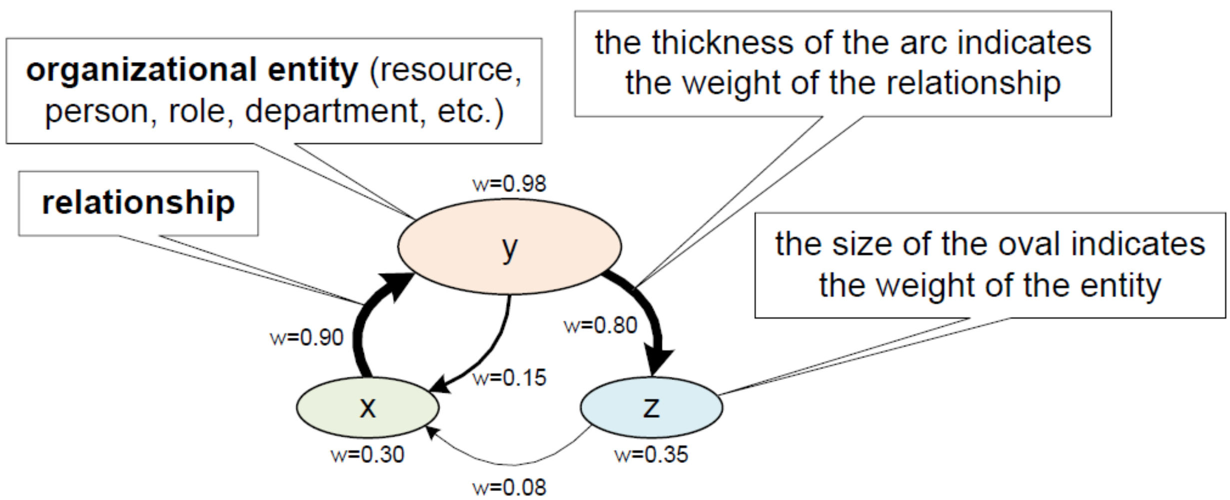



**Figure 2**. social network notations [11]

There are many ways to measure the importance of nodes [68]; here three notations are provided in this realm. First, “Degree centrality” which is the number of connections of a particular node with others. This notion is much localized. Second, “Closeness centrality” is based on the formula which is one divided by the sum of all shortest paths to a particular node. Third, “Betweenness centrality” which corresponds to the fraction of shortest paths between any two nodes going through the target node. These social networks can be created by the handover of work matrix [69]. For creating this type of matrix, the average number of times that the works are handed over from one resource to another resource (per case) are considered.

*Organizational Mining*- The focus of organizational mining is on learning organizational entities and mining corporation aspects of teams. Using the resource activity matrix, the roles can be learned automatically, and the process model can be improved with this information [70]. In the previous section, relationships between people, the handover of work between individuals, and social network information are learned. In this Process Mining application, event logs are used to derive information related to roles and refine the process model. For mining roles, first, the resource activity matrix should be constructed. Then, similar resources based on, for example, doing the same activities all over the event log will be discovered. To make the comparison easier, one needs to see every resource (row of the matrix) as a vector. Then, the "distances" between vectors using Euclidean distance, Manhattan distance, Minkowski distance, and Pearson's correlation coefficient will be calculated to know how far apart two resources are. Also, this comparison can be used to infer a social network based on the similarity of profiles. Learning role information is just an initial point. There are other information which might be learned about resources including: (1) "When are resources available?" (e.g., part-time, vacation, illness, and shared among different processes); (2) "Which resources do collaborate well with each other?"; (3) "Which resources do perform well on specific activities?"; (4) "What is the relationship between the workload of the people and how fast and efficient they work?" (Yerkes-Dodson law of arousal).

*Operational Support-* Operational support is a technique that helps its users to predict the remaining flow of a case [11]. It focuses on answering the "what will happen" and "what is the best that can happen" questions. Pre mortem data -data about cases that are still running- is the starting point for operational support. Three Process Mining activities can be used in this context: detection, prediction, and recommendation.

Detection is about online detection of bottlenecks or process deviations. In Figure 3, on the left side of the figure, a person is working on a case at a particular point in time. On the other side, the normative model (right side of the figure) has been made by a group of decision makers. Then, information system and operational support system interacting with each other. The moment that person does the activity, the partial trace will be sent to the operational support system. Any deviation between trace and the base model considered as a violation and the user or the manager is notified that such a deviation has taken place [11].

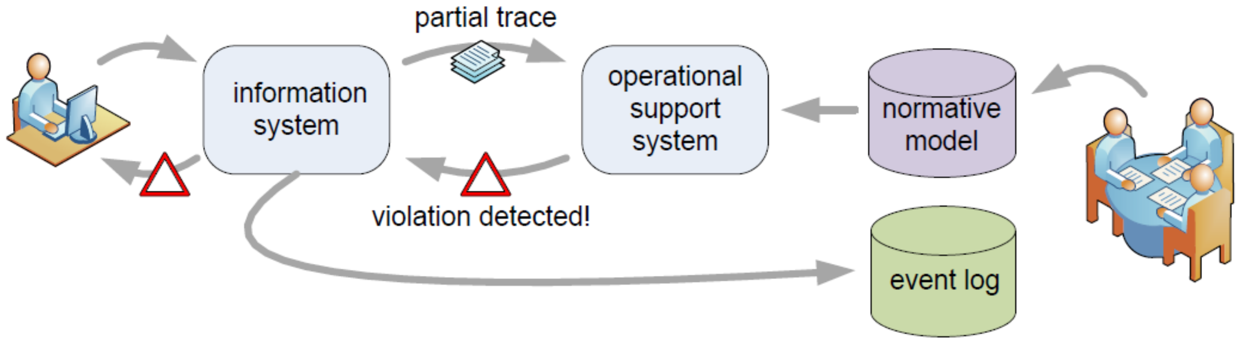



**Figure 3** Detection procedure [11]

Prediction is about predicting finishing time, possible scenarios, and probable deviations. In Figure 3, the person in the left would like to know the remaining time of the trace or the probable outcome of it. In this case, a predictive model, which is learned by historical information (event log), is used. The partial trace and predictive model are inputs for the operational support system to enable prediction in a way to capture the future events and the estimated completion time of the trace [71] [72].

The Recommendation phase is about finding the best way or ways to finish the case such as the best next activity, a suitable resource. Basically, Recommendation is a subtopic of Prediction [71] [72]. First, we need to predict all possible scenarios of the future of a trace. Then, one can compare them together to find the best possible solution based on the type of problems and maximize the project utility functions. In this procedure, a recommendation model is learned from the event logs. The recommendation is always given concerning a specific goal such as minimizing the remaining flow time, minimizing the total costs or minimizing resource usage.

Conformance Checking, Enhancement, Social Network Mining, Organizational Mining, and Operational Support are among the most useful categories of Process Mining applications which were reviewed in this section. The next section provides the real-life case studies of Process Mining in different industries.

### 2.1.6. Process Mining case studies

Successful applications of the Process Mining in different categories such as process discovery, conformance checking, and process enhancement can be seen in a wide range of industries (including manufacturing, healthcare, Construction, Oil and Gas industry, Silos Management, Delivery Process, Customer Service, and finance [8]). Also, some applications are reported in infrastructure-related workflow management (such as airport management [9] or urban management services [10]). Most of such applications depend on the information exchange standards specifically developed to analyze event logs. In AEC, openBIM standards such as IFC (Industry Foundation Classes) are the information exchange standards which makes them potential candidates for Process Mining and its applications.

The Urban Management Service case study deals with a log obtained from a local municipality of 90,000 citizens, located in the Netherlands [10]. Heretofore, Mining eXtensible Markup Language (MXML) was a standard to exchange and store events logs; because of its shortcomings, XES replaced the MXML standard in capturing event logs [57]. They used their workflow system, and in this case study the authors extracted process logs and stored them in the MXML format. The log contains 570 cases, 6616 events, 10 activities, and 110 employees. The work procedure includes invoice scanning, sending, registration, approval, and purchasing. The author used this event log as the input of ProM (a Process Mining tool) to mine a model for the workflow and extended it with performance and organizational characteristics. Then, they simulated the model with different waiting time configurations and generated an event log. The quality of the simulation models was investigated by comparing the real event log and generated the event log. The result is promising, and it showed that it is possible to construct simulation models based on real-life event logs.

Neste is a Finish Oil and Gas Company who used Celonis Process Mining technology to increase their business maturity and achieve a better process management capability [8]. Neste is a large company having 5000 employees and various business units with different supply chain issues. Considering the fact that management of these number of processes and their probable issues are laboursome, they tried to visualize and analyze their transactional data using Process Mining techniques to handle their logistics processes. The company made some changes to their business process management teams to increase the process management capabilities. They implemented a new database management system called “SAP HANA” to obtain the required raw data for their analytics. Neste chose Celonis Process Mining tool since it could handle SAP environment more efficiently. As a result, people from different process areas of the supply chain could use identified bottlenecks and possible improvement opportunities to increase the internal procedures efficiencies.

## 2.2. Building Information Modeling

BIM is a well-known concept in the realm of civil engineering which can be defined as a product (Building Information Model) or a process (Building Information Modeling). The second definition is the subject of this thesis. To approach BIM from a process management point of view, this section first engages the discussion about process modeling and management in AEC. Then, a literature review of BIM management is provided as an advanced form of process modeling and management in the construction industry. Later, a brief overview of BIM eXecution Planning (BxP) guide is provided as the most advanced document for BIM management. In the end, some case studies of Process Mining within BIM environments will be discussed.

### 2.2.1. BIM for process modeling and management

BIM is mainly used until the end of a construction phase; Goedert and Meadati added that it is essential and value adding to use BIM models for operational and maintenance procedures [73]. Azhar et al. mentioned that BIM encourages the integration of different models and roles in reaching a better and more accurate simulated model (which naturally leads to more efficiency) [74]. In 2010, Rezgui et al. talked about knowledge management in the AEC industry and how it needs some development to satisfy industrial needs [75]. In 2013, Bryde et al. also added that BIM helps not only with geometric modeling, but also it has positive results when it is used in the management of construction projects [76].

In 1998, Ballard and Koselka stated that design processes are conceived as conversion of input to output within AEC industry; Work Breakdown Structure (WBS) and critical path method (CPM) are used for managing these processes [77]. They reviewed the current practice and thinking of design management and tried to propose an agenda for design management research to eliminate waste from the flow process and reduce the rework. Unfortunately, the importance of information flow among designers has not gained enough attention in the industry and research, which causes defective and imperfect workflow practices. Conducted surveys within design teams predicted that negative reworks exhaust around 50% of design time [78]. These negative work iterations can be a result of inaccurate, obsolete, or missing information [79]. Often, team members work individually and are not aware of the organizational information they are withholding. Additionally, Austin et al. emphasized that capturing the processes with the help of analytical design planning technique results in waste removal, better decision making and reducing reworks.

Although BIM is used for better management of construction projects manually, there is no standard link and automated workflow between BIM and AEC process management. Because of the growing use of BIM and its enormous potential for producing valuable information, experts in AEC industry started to see BIM from various aspects and think of it as an opportunity which can minimize reworks, workflow issues, and collaboration problems among different parties in a project.

### 2.2.2. BIM management

In 2009, Kim and Grobler [80] proposed an ontological approach for consistency checking and its guideline resulting in decreasing of potential conflicts throughout the design coordination process. However, they mentioned that further works are needed to automatically extract constraints and requirements from IFC-BIM data into ontological representation. In 2010, Baoping et al. [81] mentioned that lifecycle management of construction projects could be facilitated via new information transmission methods in BIM. Although the author stated that BIM not only resulted to technological change but also it mandated some changes in the management of AEC industry by facilitation of information exchange, it is not mature enough when it is compared with Project Data Management (PDM) in the manufacturing industry. In 2011, Azhar mentioned that BIM has a multitude of benefits for AEC industry such as integration of the roles, visualization and an average of 634% return of interest [82]. However, there are some risks and challenges; lack of a well-defined transitional construction process model and need of a purposeful exchange of meaningful information are among them. Legal pitfalls such as data ownership and risk sharing are also considered as challenges of BIM practices in the projects. In 2012, Hartmann et al. drew attention to low empirical studies in BIM and did two case studies in cost estimation and risk management to fill the gap in this regard. The result of these two cases showed the possibility of expanding BIM-based tools in projects; it can be reached by technology and organization alignment via having a more thorough understanding of project management methods and aligning BIM-based tool functionality to these processes [83].

Moreover, BIM usage possibility in facility management is becoming more popular for the researchers [84] [85]. In addition to that, technologies such as cloud computing [86], passive RFID tags [87] and 3d scanning [88] facilitated the adoption of BIM in facility management and building operation area. Maintenance and operations are responsible for 60% of any building or facility lifecycle cost. Asset Management which includes facility management tries to maximize the efficiency of the project lifecycle. BIM act as a supporter for asset management filed by enabling effective implementation of its information system [86].

### 2.2.3. As-plan modeling using BxP guide

One of the contractual requirements for BIM implementation in projects is to define a BIM plan. Different countries developed and used different guidelines to do so. VA BIM guide, Guides Messner et al., Singapore BIM guide, AEC (UK) BIM Technology Protocol and BIM Project Execution Planning Guide (BxP Guide) are among them. In North America, BxP Guide is more commonly in use as the reference document. Throughout a BIM project, models change rapidly. For having a track of changes, it is essential and required to follow different versions of the model regularly. IFC standard is an appropriate data standard to keep tracks of BIM models [89]. It becomes more crucial, once multiple people are involved in the creation of the models. BIM coordinator of each discipline is responsible for doing this task [90]. Building Information Model (BIM) is a digital representation of physical and functional characteristics of a facility [4]. BIM and information exchanges are designed to increase efficiency in the Architecture, Engineering, Construction, Owner Operator (AECOO) industry. To improve the benefits of BIM in a project, it is required to have detailed and inclusive planning. It helps all parties involved in the project to know the tasks and prospects about the BIM applications. A sufficiently detailed BIM plan must define BIM uses on a project, process design for completion of BIM throughout the project lifecycle. Because every team and project have features, BIM execution strategy should be modified based on the team capabilities and the project characteristics. Collectively, there are four main steps in a BIM execution plan including Identifying BIM uses based on project goals and check whether these BIM uses have value or not, Designing the BIM execution process by making the maps, Defining the BIM deliverables as the required BIM exchanges, Development of infrastructures [4].

BIM would provide many benefits if it is implemented correctly. The benefits would be design quality, better prefabrication, field efficiency, and innovation in facility management. There might be inefficiencies and delays in projects using BIM if it is implemented in the wrong way. However, the BIM provides many benefits to projects; it is required to do feasibility studies regarding costs, technology, and practical knowledge. To implement BIM into the project process, having a detailed BIM Execution Plan is necessary. BIM plan provides insight and implementation details for team members in the project. This plan should be created in the early phases of a project and be developed and maintained throughout the project lifecycle. Additionally, it should define BIM implementation scope, the process flow for BIM tasks, information exchange, and required infrastructure for implementation. On the one hand, BIM can have many values as discussed above. On the other hand, it might have risks due to lack of knowledge and people unfamiliarity with new technologies. The best practice to decrease the risks is to increase knowledge about the unknowns.

There are twenty-five common BIM uses that are defined based on experts' opinions, case study analysis, and literature review [4]. BIM use is a unique task that can be done by implementing BIM in the project. These BIM uses should be evaluated and analyzed to check whether they are beneficial to the project or not. The process for evaluation and selection of these BIM uses will be discussed in the following chapter. After identifying the BIM uses, a high-level map showing relation and sequence of the BIM uses on the project should be developed. As a result, all team members can understand the interaction of their work with other team members. Then a more detailed BIM use should be designed by team members who are responsible for each BIM use. The information that needs to be exchanged between different responsible parties in the project should be defined precisely. Furthermore, the author and receiver of each file should understand the content of the exchanged information.

There are fourteen categories of information which should be included in a BIM execution plan [4]:

- BIM Project Execution Plan Overview,
- Project Information,

- Key Project Contacts,
- Project Goals / BIM Uses,
- Organizational Roles / Staffing,
- BIM Process Design,
- BIM Information Exchanges,
- BIM and Facility Data Requirements,
- Collaboration Procedures,
- Quality Control,
- Technological Infrastructure Needs,
- Model Structure,
- Project Deliverables,
- Delivery Strategy/Contract

The result of the information above will be summarized to BIM maps. BIM maps are currently are defined in two levels of details; level 1 and 2. The first level represents the high-level procedure of all the project BIM uses and information exchange requirements throughout the project lifecycle of the project. On the other hand, Level 2 BIM maps are defined at the programming phase for every BIM uses involved in the project, separately. Hence, a more detailed sequence of activities for each BIM use would be available.

BxP guide helps team members to develop detailed project plans including the goals, process, information exchanges and required infrastructure for BIM implementation. Hence, it will result in a successful BIM implementation. However, it might take time and cost to develop the BIM plan for an organization, the benefits of the BIM implementation outweigh the expenses. Often the need for a more detailed map cannot be ignored which help managers to have a more in-depth will look toward the project procedures. Also, the available BIM maps are as-planned, and no tool can track and map the real-world procedures of BIM uses. Having the as-happened processes can help managers to have analytical information about what is happening in projects such as bottlenecks, process deviations, human resources issues.

### 2.2.4 Case studies of BIM in Process Mining

“Process identification, modeling and evaluation has recently attracted attention in AEC/FM ( [91], [92], [93], [94] & [54] among others). BIM processes are usually planned either externally by the aid of BIM protocols (in the form of BIM frameworks, workflows or process maps [4], [95], [96]); or internally by the project team [91], [97]. Since BIM is value adding, only to the extent that it contributes to the accomplishment of the organization’s mission, the process of BIM implementation must be controlled and monitored from an organizational (rather than individual) point of view [98]. Some studies have focused on performance evaluation of BIM processes, taking into consideration different components such as performance, impact, Return On Investment (ROI), capabilities and maturity [99]. While the existing studies mostly focus on outcomes of the BIM processes, bottom-up approaches to evaluate the as-happened processes (when the actors execute activities) have been left understudied. This can be mostly due to the lack of access to detailed information regarding end-to-end processes executed in action. Process Mining, however, provides the tools for resolving this challenge and filling the gap.” [2]

“The feasibility of BIM and Process Mining integration in the construction phase has been recently studied [54]. The study created as-built 4D models by merging several point clouds captured during the construction phase. By comparing the as-built and as-planned 4D models over time using image processing techniques, the study captured delays as well as ahead-of-schedule activities. It also used Process Mining engines in combination with BIMserver to analyze the process variants, bottlenecks and the social networks of the project actors. The results of that study showed that the integration between BIM and Process Mining would not only benefit the ongoing project by detecting the process variations and the as-happened social networks but also can help to reuse the project data (which is usually disposed of), and systematically employ them, as lessons learned, in planning future projects. In another case study, the possibility of applying the Process Mining techniques with maintenance data has been investigated at a hospital [55]. As a result, problematic facility management processes were discovered and optimized. Another case study on the design phase used the data generated from a system engineering tool, called Relatics (which is employed to solve the project complexity). They applied Process Mining tools to the collected data to examine the possibility of detecting the underlying design process. Despite performing social network analysis and detecting the critical roles in the project, the lack of integration between BIM (as a potential ERM), and data analytics, hence the lack of automation was reported as the major limitation of the work [53].” [2]

“The recent studies provide a proof of concept for applicability of Process Mining to the AEC/FM via BIM. However, a successful and meaningful application to harvest the intelligence and achieve the breadth of advantages promised by Process Mining will require higher levels of automation. Such automation has at least two significant prerequisites; (i) access to event logs in AEC/FM, and (ii) availability of information exchange protocols compliant with both BIM needs and XES requirements. The present work reflects on the former need. We have used the IFC standard to develop a tool for extracting event logs from BIM. This tool and its application are explained in the following section.” [2]

## 2.3. OpenBIM

“OpenBIM is a universal approach to the collaborative design, realization, and operation of buildings based on open standards and workflows.” [100] OpenBIM is initiated by buildingSmart and some other leading software vendors. Considering that the AEC industry is moving towards BIM, it is essential for different analysis tools to be interoperable with a non-propriety openBIM schema such as IFC [101]. An IFC Model View Definition (MVD) is an element of IFC standardization in 2006. It defines a subset of IFC schema that is needed to satisfy one or many exchange requirements of the AEC industry. The method used and propagated by buildingSMART to define such Exchange Requirements is the Information Delivery Manual, IDM (also ISO 29481).” [100] I am also fond of the other definition indicating that MVD works like a filter for IFC; it filters out the extra information and keeps only the related ones. Sometimes MVD eliminates some information for the matter of security as well. This section provides a holistic view of one of the most important Open BIM standards, aka IFC. Then, some of the IFC MVDs will be reviewed.

### 2.3.1. IFC

Industry Foundation Classes (IFC) can be described as a data standard for building information which is developed and maintained by a non-profitable organization building smart. The purpose of this organization is to provide a basis to share information throughout the lifecycle of built environment assets for all the project participants. Considering that, IFC is a set of agreements responsible for building element (either physical or non-physical) representation in digital format. Non-physical elements are documents, costs, schedules, resources, and any other properties [102]. Being an open and neutral data schema enables IFC to a wide range of possibilities for the design, construction, and operation projects (BuildingSmart, 2016a). All in all, it brings more collaboration and decreases waste of money and time in the AEC industry. Also, it supports different configurations and level of details to increase the interoperability among different industry domains and software applications. Building elements are described as objects with attributes [103]. Another advantage of IFC is that any missing attributes for objects can be added if needed.

In 1996, Froese explained the different process model for information management throughout AEC [104]. Hence, he emphasized that there is a growing need of a unified standard for presenting this information including STEP or ISO standard 10303, ”Standard for the Exchange of Project Model Data”. In 1999, Froese presented models related to scheduling and estimating in IFCs and addressed different issues in costs, construction processes, products, and resources [102]. In 2001, Hassanian et al. proposed an extension of IFC for maintenance management information exchange between different resources [105].

IFC schema is constructed of entities (or classes), the relationship between entities, and attributes. A collection of these schema results in an IFC model. [106] In practice, this IFC model resembles a building information model. There are four IFC-related methodology standards which are needed for a better understanding of Industry Foundation Classes. The first standard (ISO 16739) is IFC which is described in the previous paragraph. The other three standards are International Framework for Dictionaries (IFD) ISO 12006-3, Information Delivery Manual (IDM) ISO 29481-1 ISO 29481-2, and Model View Definition (MVD) buildingSMART. IFD includes the buildingSMART Data Dictionary (bSDD) which contains the terms, attributes, and vocabularies for things. It eases the interoperability by defining a unique term for users. IDM is related to information regarding individual processes and deliverables with their order during a project; i.e., it helps to document information exchange requirements in a project. Each MVD is a subset of the

IFC schema defined for a particular usage (e.g., exchange requirements of the AEC industry). In other words, MVDs are just filters for IFC format data which is originated to satisfy different ends.

In 2006 IFC 2x3 was released to fill the gaps for IFC implementing and limitation of IFC [107]. In 2007, Kivienmi tried to improve the quality, consistency and real-world usability of IFC 2x3. 1n 2010, “IFC certification 2.0” was released by buildingSMART to bring major improvements [108]. These improvements were more specific and targeted MVDs as well. More specifically, it involved IFC interfaces and software vendors in getting certified for a specific MVDs for IFC2x3 data model; A Software would not be “IFC compatible” but “a specific MVD compatible”. Although IFC 2x3 was too conservative in adopting novel features, IFC4 is open to new features. There are two main reasons for the development of IFC4 other than the multiple improvements and extensions. The reasons are to put quality over speed and get the full ISO international standard status by having the final version in 2012 [109].

Since the whole IFC model is too big and had extra information that may not be useful for a particular purpose, two approaches were proposed for extracting the required information from an IFC file [110]: (1) parsing and running SQL query statements or using the available views specified in SQL or XML for data stored in an IFC server [111]; or (2) utilizing the Model View Definitions (MVDs) developed for particular information use cases [112] [113].

### 2.3.2. Model View Definitions (MVDs)

In 2014, Jeong et al. developed an MVD to, first, translate BIM models into a Building Energy Modeling (BEM) Software and, second, facilitate reuse of BIM data in building energy simulation [114]. They did it in four steps: (1) creating a process model to document the information mapping from BIM to BEM; (2) creating a class diagram to illustrate the object relationship and required information; (3) developing the software to do BIM to BEM translation; (4) verification and validation by case studies. In 2016, Pinheiro et al. due to the limitations on information exchange between BIM tools and Building Energy Performance Simulation (BEPS), authors developed an MVD to facilitate the standard information exchange between BIM and BEPS tools [115]. Hwang (2004) created an MVD to extract information from an IFC model for the use case of quantity take-off [112]. Chen et al. (2005) developed an MVD to extract the required information for structural analysis purpose [113].

The Construction-Operations Building information exchange (COBie) is a non-propriety data format to deliver the none-geometric BIM asset data as distinct from geometric data. COBie and FM handover MVD are among the first recognized IFC applications for none-geometric building information. Patacas et al. elaborated how IFC as an open BIM standard and COBie can support facility managers' information requirements [116]. CAD vendors have developed Coordination View MVD to bring consistency to their IFC geometry import and export implementations [117]. In the attempt to get a coordination view 1.0 certification, the vendors had to develop about 264 test cases and exchange them with each other back and forth to manually and visually check the results [118]. At the end of this error-prone qualitative process, they would be awarded a pairwise certification between two applications.

On the other hand, Coordination View 2.0 (CV 2.0) was the beginning of an effort to decrease the errors. CV 2.0 changed the certification process, and the committee process documented the complete CV 2.0 MVD and made it available on buildingSMART website as a certification requirement [119]. The process focused on checking the quality of IFC produced by an application. CV 2.0 is a model view definition. Its subschema is provided in three ways [120]: (1) as a formal IFC2x3 sub schema which includes List of all IFC2x3 entities in CV 2.0, (2) as an IFC2x3 EXPRESS schema for CV 2.0, and (3) as an IFC2x3 EXPRESS-G Diagram for CV 2.0 which is an A0-size document to show the graphical entity relationship for CV 2.0. For defining the property sets for the elements, there are excel sheets for each building element (e.g., a wall), which consider all of the possible scenarios of property assignment in an IFC step file, considering CV 2.0 MVD. Also, IFC2x3 EXPRESS-G Diagram for CV 2.0 is a valid subschema for IFC2x3 CV 2.0. It is used in this project as a guide for understanding the data structure of this MVD and, at the same time, to find our IFC signature-based comparison algorithms–a literature review for these algorithms is provided in the next section.

As discussed before, there is a need to have a BIM source file that can capture and handle all information through a Design Authoring process. With the review of entity list and property scope definition of different MVDs available on BuilidngSMART website, IFC2x3 CV 2.0 is the chosen MVD for this study.

### 2.3.3. IFC models comparison

Platform-neutral BIM facilitates collaboration among parties in a project using a model server [121]. A model server is a kind of database system enabling multiple users to download, upload, share, and coordinate models. An IFC based model server enables easier collaboration between the different applications used within a project lifecycle [122]. The idea of IFC-based collaboration is created around finding changes between two shared models and merging them with the shared data source. So, finding changes between different IFC models is a critical issue [123]. For comparison, there should be some Global Unified IDs (GUIDs) which are fixed and uniquely specified combination of characters for an object and are common throughout all the models. However there are already available GUIDs in IFC, but they are not always fixed and heavily criticized by researchers [124], [125], [126]; e.g., a GUID for a curtain wall might change in different IFC versions of a model. Since it is a vital part of comparing different model, a signature-based matching is proposed for IFC models as a complementary method for the GUID-based comparison algorithm. It investigates the possibility of using a combination of name, location, address, and specification with GUIDs as a more confident way of comparison [123].

Several studies address the issue of change management in IFC models [127] [128] [129]. Plain text-based comparison applications, as the first category of algorithms, are not good candidates for IFC comparison; since they do not have an organization of comparable files. GUID (globally unique identifier) is a way to uniquely identify object instances which follows the universally unified identifier standard (UUID). GUID-based comparison methods are the next appropriate strategy for comparing IFC files [130] [129]. In this method, two objects in two IFC files are considered equal if and only if their GUID match and vice versa. EvaSys (EXPRESS Evaluation system) is developed to determine the number of unique instances in two IFC models within their EXPRESS schema [129]. This method used GUIDs to determine the matching objects and then distinguish differences by comparing the values of object attributes. EvaSys method has four weaknesses: (1) it does not analyze IFC owner history; (2) it does not consider extra instances (3) it ignores the possibility of GUIDs' changes happening throughout data exchange (IFC exports/imports) between different systems [128] (4) it just counts the structural elements and ignores semantic elements. That said, the GUID-based method is one of the fastest comparison strategies.

The graph-based approach presented in [131] compares two oriented graphs resulted from two IFC files. It converts the IFC model to RFD-RDF signature to compare the differences between two IFC files. Being independent of GUIDs for comparison purposes is a limitation of this algorithm. Another method is a tree-based comparison approach using the hierarchical structure of IFC [127]. However, the tree-based method is a more acceptable comparison results than previously mentioned GUID-based approaches [123], it still depends on GUIDs in its comparison algorithm. The other approach [131] which is similar to the previous method uses data instances instead of GUIDs to determine the comparable items between two IFC files. Then, it utilizes three data instances (instance name, entity name and attribute values) and their relationships to make an IFC file hierarchy to do the comparison. However, this approach is faster and does not rely on GUIDs to match instances; different BIM tools have a different naming structure for instances. Thus, it is not a reliable way to find matching objects and instances to make IFC comparison.

Last but not least, the Signature Matching approach is proposed by [123]. Previous approaches relied on GUIDs or instance name to find matching items for comparison purposes. Those approaches were not reliable since GUIDs and names are subjects to change in data exchanges. Considering the dynamic identity of objects in IFC files, this approach combines a set of values to determine a set of values as a signature to detect the matching instances. Therefore, a signature does not depend on GUIDs and can be utilized for IFC models created by different tools.

In this chapter, a literature review on “process and Process Mining”, “BIM and process management”, and “openBIM standards” are provided. In the next chapter, I propose a methodology to archive BIM files in an openBIM standard format (IFC) and convert it to an event log (XES format). It enables BIM as a data source for Process Mining techniques. Then, I use Process Mining applications and analytics to provide insights and support management of BIM processes.

# Chapter 3
# RESEARCH METHODS APPLIED

The research focuses on the activities which involve BIM tools in the design authoring process. To illustrate the high-level process, the design authoring map, which is a level 2 BIM process map [4], is shown in Figure 4. The red box in this figure highlights the scope of this study (the activities, decision points, and information exchange instances); i.e., this part of the design authoring process is targeted to be discovered. This part of the design authoring process is performed in three different phase: (a) Schematic Design (SD): “The architect prepares a series of rough sketches, known as schematic design, which show the general arrangement of rooms and of the site. Some architects also prepare models to help visualize the project. The owner approves these sketches before proceeding to the next phase.”; (b) Design Development (DD): “The architect prepares more detailed drawings to illustrate other aspects of the proposed design. Floor plans show all the rooms in correct size and shape. Outline specifications are prepared listing the major materials and room finishes.”; (c) Construction Documentation: “Once the owner has approved the design, the architect prepares detailed drawings and specifications, which the contractor will use to establish actual construction cost and build the project. The drawings and specifications become part of the building contract.”. [132] The project aims to discover as-happened level 2 BIM process map. The sequence of these activities or the process will be discovered with Process Mining techniques. Also, it is possible to discover a more detailed process map. The discovered process can be considered as “level 3” BIM process map.

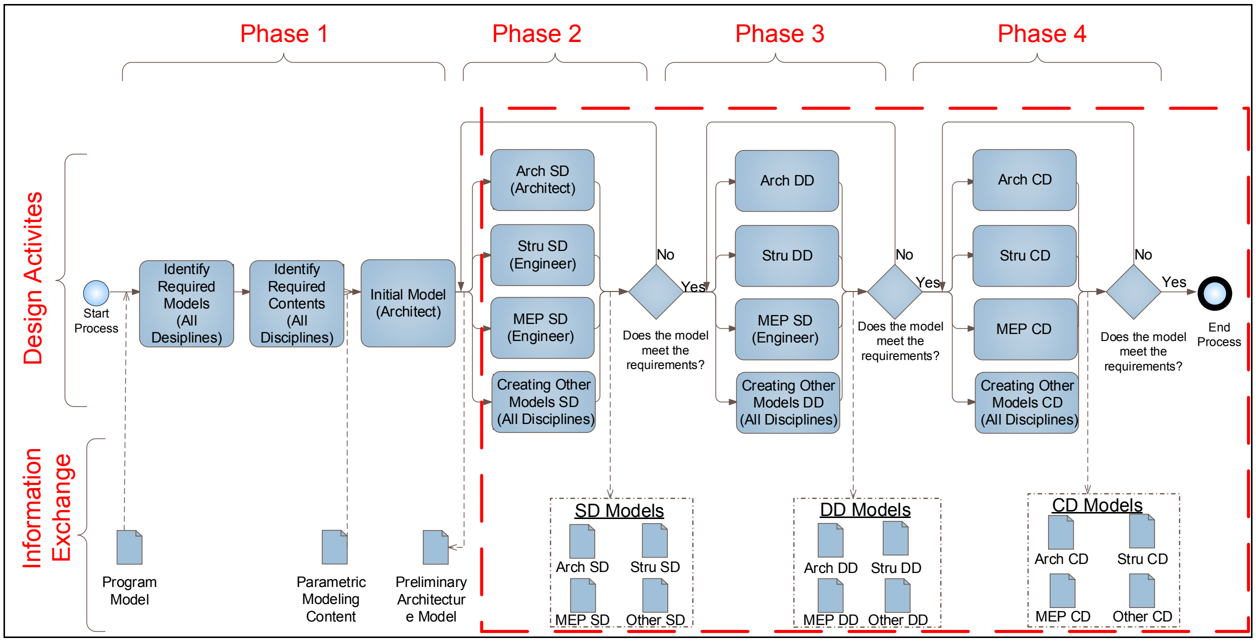



**Figure 4.** The scope of the methodology: based on level 2 BIM process map [4]

Two versions of workflows are proposed for this research. The first generation was applied in the first case study. Based on the result of the first case study, the workflow was improved; the improved version is called second-generation workflow. The second case study was conducted based on the second generation of the workflow. Figure 5 illustrates the first generation of the research workflow in a high-level. In parallel with modeling in Revit, database preparation or IFC archiving is performed using a dynamo code, automatically, and without any intervention of a human actor. After preparation of the IFC database and based on the BIM manager's decision, IFC file comparison and capturing event logs are performed using IFC Logging algorithm. It can be performed either in parallel with the design authoring phase or after its completion. Then the generated event log (which is in CSV format) is converted to XES format using the available XESame module. Finally, BIM manager can use ProM to unearth the as-happened design authoring processes, actors' social networks, and other process analytics.

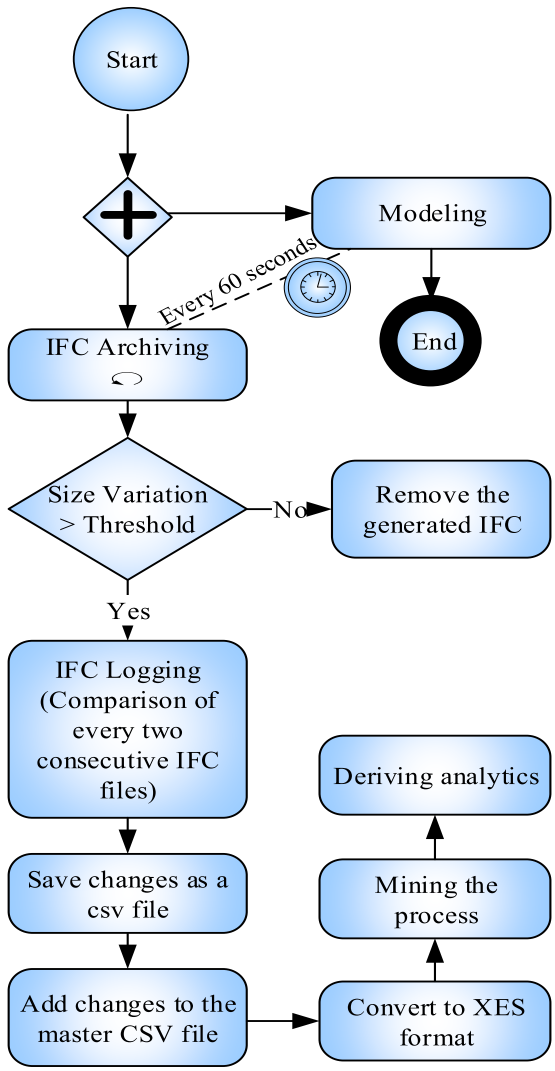



**Figure 5.** BIM – Process Mining integration model: first-generation activity diagram (for the first case study) [2]

Figure 6 shows the activity diagram for the second case study (second generation of the algorithm). There are four significant improvements compared to the previous algorithm:

(1) In the comparison of IFC files, more categories of differences can be detected. The first algorithm only explored the existence of elements (which includes adding or removing elements' activities). The second algorithm, however, can detect elements' attribute changes and element rotations/relocation activities as well;

(2) The second version of the activity diagram for the BIM-Process Mining algorithm is shown in Figure 6. First, the duration for automatic generation of IFC files was changed to a variable amount

(between 60 seconds and 300 seconds) based on the IFC file size. During the IFC generation, modelers cannot change anything in the model and have to wait until the end of this step. The reason for this change is that the IFC exporting might be time-consuming for the more detailed BIM models (Figure 7); this improvement reduces the designers' waiting time;

(3) During the design modeling task, the new version of the algorithm detects the IDs of the selected elements and stores it in a CSV file. Afterward, in the comparison phase, this CSV file would be used to narrow the search algorithm to the elements which were selected (at least once) within the generation timespan of the two consecutive IFC files. With this search algorithm, a lot of unwanted comparisons were removed, and the comparison time was decreased drastically.

(4) Finally, the logged activities (changes) would be saved directly to the master CSV file by doing the CSV merging automatically. This algorithm modification reduces some manual procedure and will improve the automation of the algorithm.

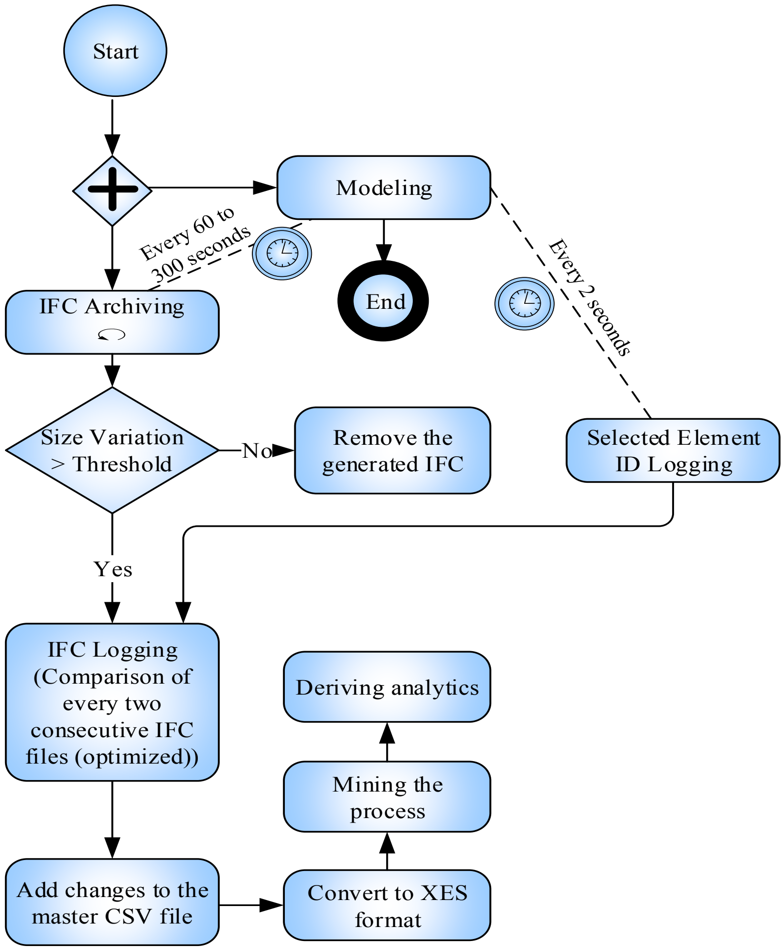



**Figure 6.** BIM – Process Mining integration model: second generation activity diagram (for the second case study)

Different tools are used during the methodology and implementation phases of this research. BIMVision, Sublime, Dynamo, Revit, and ProM are among them. Firstly, BIMVision is a freeware model viewer which visualizes BIM models captured in IFC format 2×3 [133]. It has a multitude of built-in features and provides a plug-in interface. Plug-ins can be found in its plugin stores and have different functionalities such as clash detection, IFC commenting, and reporting tools. Moreover, the BIM Vision API enables specific users for their needs. It also empowers the BIMVision to be integrated with other systems, therefore use their functionalities to fulfill their BIM needs such as accessing the model structure, element types, and properties. The API is available for C and C# .NET. At the early stage of the project, BIMVision was used to visually and manually compare two IFC files. Also, it was used for validating the code and testing during the development of IFC logger dynamo codes. Secondly, Sublime is an advanced text editing tool enabling the user to understand, manage, and compare complex codes. Sorting, searching, changing the syntax and indentation are among its capabilities. This tool was utilized for capturing the general rules for differences and use them in coding as algorithms in comparing two IFC files. Thirdly, Dynamo is an open source programming environment which extends BIM with algorithms and data. It can be run either stand-alone "Sandbox" mode or as a plug-in for Maya or Revit. Since Dynamo can be applied within Revit environment, it was selected for this project. Dynamo enables sequences of action by connecting elements and piece of codes together to achieve processing data or generating geometry. Also, it is supported by a community of specialist and avid users that help users to benefit from the discussions and work gallery in a vibrant forum. In this project, dynamo was the primary programming tool to develop IFC archiver and comparison algorithms. Fourth, Revit is used by designers during the case studies to develop BIM models. Last but not least, ProM is an extensible framework that supports a set of Process Mining techniques. It is designed to discover and analyze process models. Also, it is possible to use event data to discover a model; then replay the real data on top of the model to see what is happening in real-life or where the deviation occurs. In this project, it was used to conduct process discoveries and analyses.

## 3.1. Changes to be detected (RO 1)

Once two consecutive IFC files are compared, there are four categories of changes which can be translated into some "design activities" and should be detected:

(1) Addition or removal of elements: to know which elements are added/removed to/from the newer IFC file, the list of all elements' IDs for the newer model was compared to the older version. Consequently, if there are elements' IDs in the new IFC file and not in the old file, those elements are added to the model, and if there are missing element IDs in the new IFC file comparing to the old file, those elements are considered as removed. To find the list of IDs in each IFC file, regular expressions were used to find the elements.

(2) Relocation of elements: in comparing every two consecutive files, each element's position in the new IFC file is compared to the position of that element in the older version. While in the first version of IFC Logger algorithm (Figure 5) all elements were investigated, in the second version (Figure 6) just the elements which were touched (selected) during the design (and specifically between the exporting times of the two IFC files) were targeted. This change in the updated Logger algorithm resulted in the optimization of our Logger algorithm with the elimination of extra comparisons.

(3) Rotation of elements: for capturing this category of activities, horizontal and vertical direction of each element in two consecutive IFC files are compared with each other. Then, if there are any changes in the direction of each element, the IFC Logger algorithm (which is written in dynamo) captures this type of activity for that element. Similar to the previous step, the improved IFC Logger algorithm (Figure 6) just considered the touched (selected) elements within the timespans associated with the two IFC files (creation times).

(4) Property changes of elements: Each element in IFC may have multiple properties or attributes. For logging this type of activity, while the code is comparing each two IFC files, every attribute (property) of each element is compared with the same attribute of the same element in the newer IFC file. If there is a change, an activity with the name format of "[Attribute Name] of [Element Name in IFC file] is changed" would be captured. Moreover, the unneeded elements were not considered in the updated IFC Logger algorithm (Figure 6).

Each category may contain multiple numbers of activities such as "Column #723 is Added", "Beam #654 is Removed", "Wall #279 is Relocated" "Windows #152 is Rotated" or "Volume of Wall #279 is changed". Also, each activity will have information regarding end time, start time, actor, and GUID.

## 3.2. IFC archiving and data preparation (RO 1)

In this section, the required database (which is referred to IFC files archive) is produced and sorted based on the exporting time of IFC files to be utilized in the next step, which is the IFC Logging algorithm explained in the next section. Thus, a piece of algorithm code is developed in Dynamo

(1.3.2 version) to periodically generate IFC files during the design authoring phase in Revit. Based on the IFC file size, we set the IFC generation time to be within the range of 60 to 300 seconds, which presumed to be enough to capture BIM files upon every little change (Figure 7). IFC export considered as a time-consuming action especially when it comes to the complex and large BIM models. It might even disrupt the nature of the design authoring process and annoy the designers during the work. To minimize the interruption of IFC archiver algorithm, I categorized the time setting for IFC export based on the file size in the IFC archiving algorithm.

Moreover, this code may capture redundant IFC files during the idleness. Basically, redundant IFC files are the ones which are produced when the designer is idle and does not change anything in the model. To avoid having these extra files in the archive, the code is extended to capture IFC files periodically, if and only if there is a little change between the new file size and the previous one. The threshold for this, after trials and errors, is considered "0.0004*the newer IFC file size". The name of IFC files would be their generation time in "MM-DD-YYYY hh-mm-ss" format. In the end, we will have a bunch of IFC files, from different disciplines, upon every change throughout the design lifecycle.

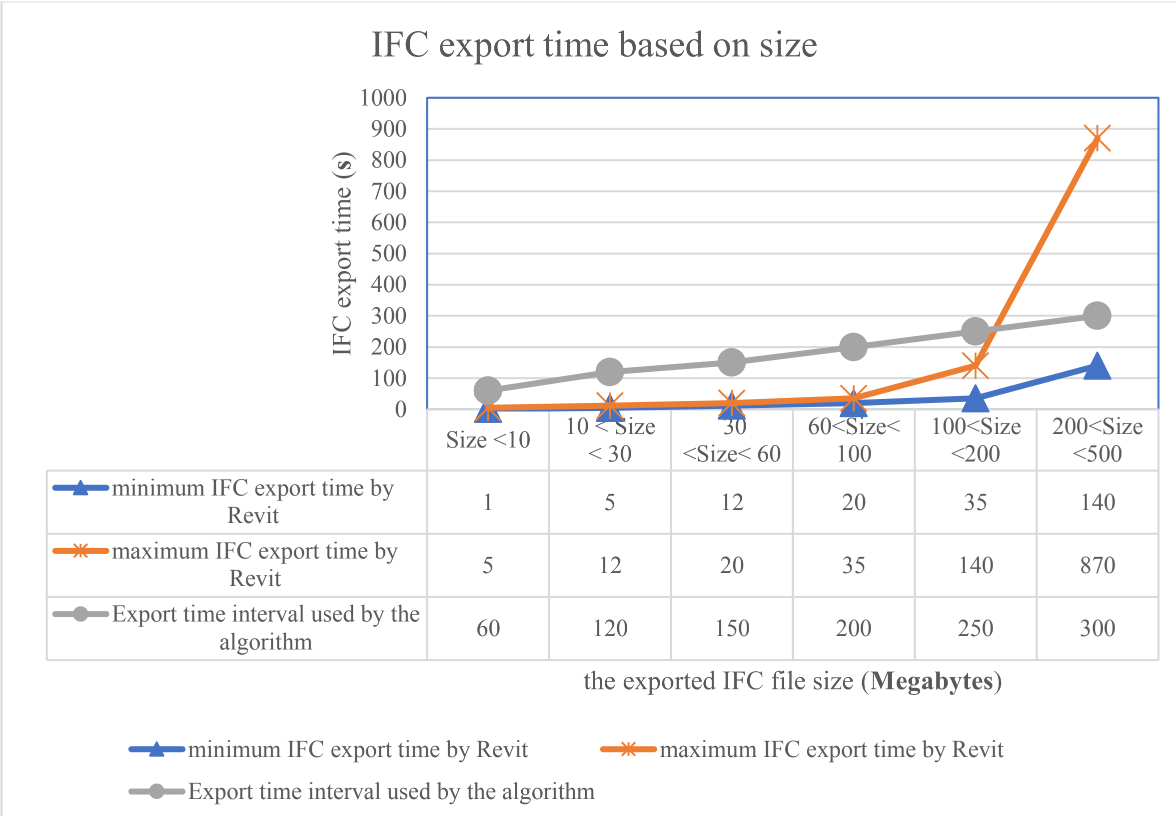



**Figure 7**. IFC export time using Revit 2018 based on the file size (The computer which was used in this project had an Intel(R) Xeon(R) CPU (E3-1270 v5 @ 3.60 GHz) and a 16 GB of RAM.)

In this part, I will explain the IFC Archiver algorithm with more detail which can be useful for future development. Figure 8 illustrates a snapshot of the created code in Dynamo (version 1.3.2). The algorithm has five main blocks. The relationships between blocks are shown in Figure 9. Each block satisfies a sub-objective of RO 1 which are explained as follows:

1) With this block, the time step for the generation of IFC files is set. Also, it provides the "current time" as an input for step 2, 3, and 5.
2) This block export elements' ID which was selected by designers and the time of selection by them. Elements' ID and selection time are exported into a CSV file on a real-time basis.
3) The third block is responsible for turning the date and time format to a string format ("MM-DD-YYYY hh-mm-ss" is the naming format used for the exported IFC files)
4) IFC exporter block helped in generation of IFC files. The inputs for this block is folder path (IFC files would be exported in this directory), name of the IFC file (which is the generation date and time), IFC file version (such as "IFC2x2", "IFC2x3", "IFC2x3CV2",

"IFC2x2", "IFC4", "IFCBCA", and "IFCCOBIE"), and two Booleans ("wall and column splitting" and "export base quantities"). For the implementation of our case studies, IFC file version and Booleans were set as "IFC2X3", "False" and "False", respectively.

5) The last step is created for the elimination of the extra IFC files which were generated. Extra files would be removed if at least one of the following criteria is met: a) When the designer is idle; i.e., when the size of newer IFC file is equal to the older IFC file; b) When the idleness time is less than one hour.

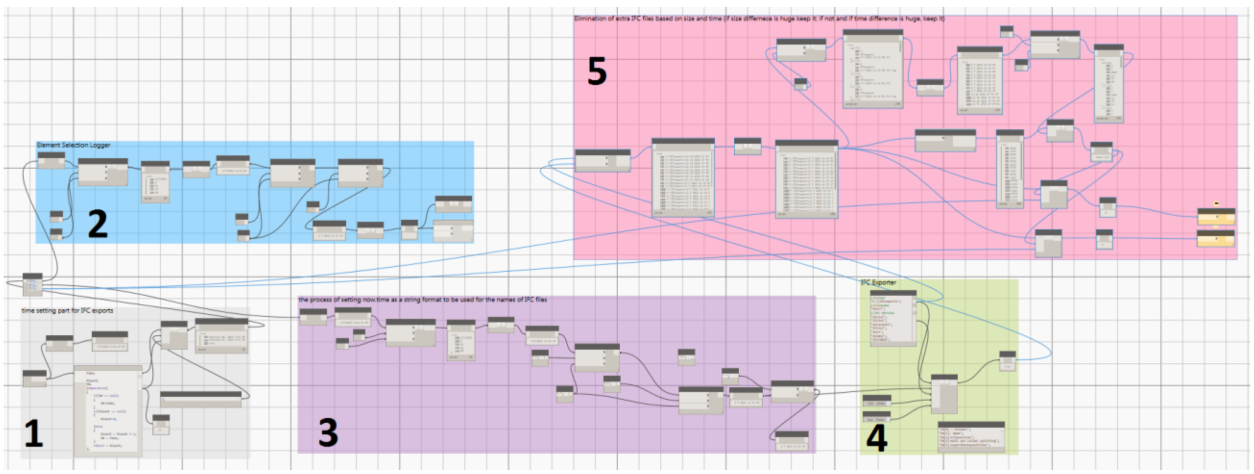



**Figure 8.** IFC archiver algorithm

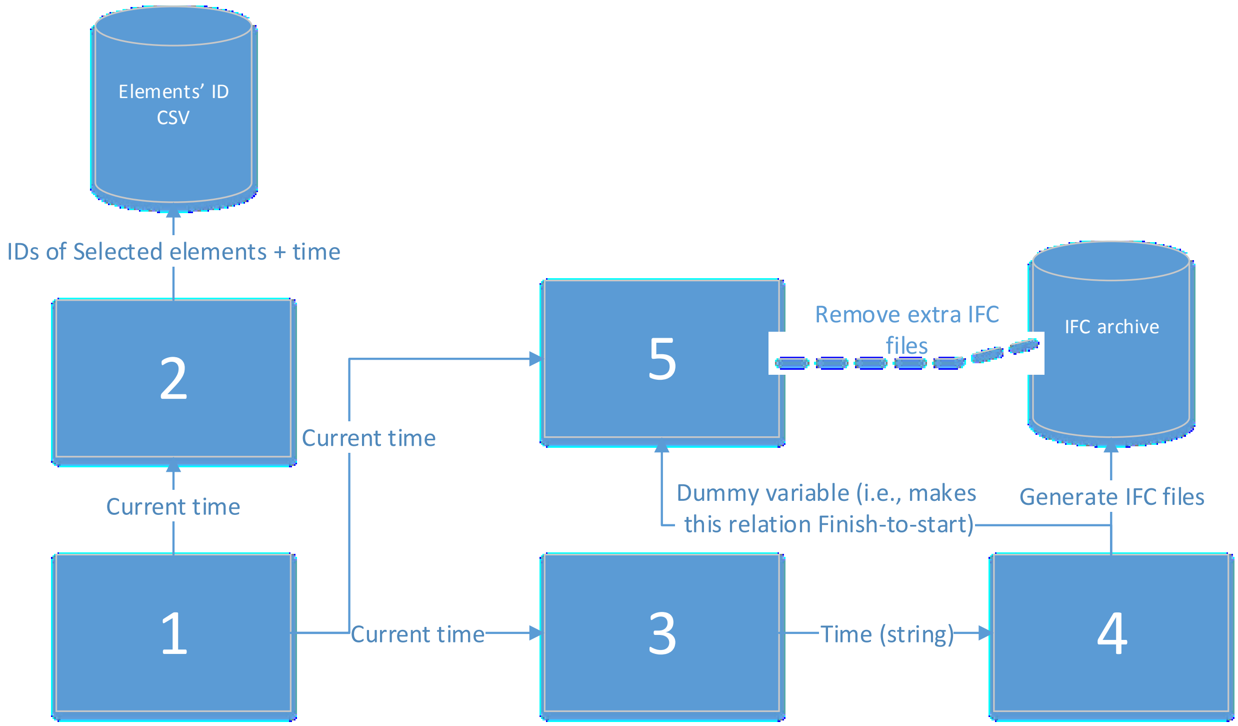



**Figure 9.** IFC archiver algorithm- the relationship between blocks

## 3.3. IFC logging (RO 2)

This part of the methodology is contributing to the second objective of this project (RO 2). In this step, which is the main contribution of our work, a piece of code is written to compare each two consecutive IFC files in our database; the IFC files archive is produced during the design phase using IFC archiver algorithm. Upon every change including adding or removing an IFC element (physical elements such as a wall, window, column, etc. and non-physical elements such as space, zone, etc.) one activity will be saved with the name of "element added" or "element removed" in CSV format. At any point in time and upon BIM manager request, CSV files will be combined using a piece of dynamo code and stored to another CSV file called "the master CSV event log". Then, the master CSV event log would be mapped via Extensions to the XES event log using a method presented by J.C.A.M. Buijs with XESame application.

Today's business process executions have different information systems. Each of them produces and store events data in different formats. To be able to do Process Mining it is required to have data in forms of event logs; which has information about cases, events, timestamps, and attributes. A standard format to capture event logs is XES which enables Process Mining tools to capture processes. The Table 1 shows an example of the CSV format event log.

**Table 1**. A small sample of the CSV event log

| Activity name | element GUID | Start time | End time | Phase | Designer |
|---|---|---|---|---|---|
| IFCCOLUMN Removed | 1zaq3g3QDCDRNzAQEfqMG x | 8-02-2018 18-52-04 | 8-02-2018 18-54-04 | Stru SD | Structural Engineer |
| IFCBEAM Added | 1iG7Z_y9X5Qhuha6Qwoo4C | 8-02-2018 19-12-09 | 8-02-2018 19-21-36 | Stru SD | Structural Engineer |
| Volume of "IFCSLAB" changed | 2E8ZvhLNP22viTYiWP$6u7 | 8-01-2018 13-27-24 | 8-01-2018 13-34-50 | Arch SD | Architect 1 |
| IFCFLOWFITTI NG relocated | 3KBhxio8LAL94uPyQizUCc | 8-02-2018 15-59-08 | 8-02-2018 16-02-10 | Mech DD | MEP |
| IFCOPENINGEL EMENT relocated | 3I87P1CqT4YAH$ZKpB4A9w | 8-07-2018 14-19-54 | 8-07-2018 14-20-35 | Arch CD | Architect 4 |
| IFCDOOR Added | 3I87P1CqT4YAH$ZL$B4Amq | 8-04-2018 10-04-29 | 8-04-2018 10-06-18 | Arch DD | Architect 2 |
| IFCSTAIR Removed | 2YjGjucsTDTBCmmGo9DnsD | 8-05-2018 11-29-31 | 8-05-2018 11-42-48 | Arch DD2 | Architect 3 |

This event log was mapped to XES format using XESame module. The resulted XES format event log is shown in Appendix 1.

### 3.3.1. IFC Logger Dynamo algorithm

In this part, I describe the IFC Logger algorithm in detail to pave the way for future improvements. Figure 10 illustrates a snapshot of the created code in Dynamo (version 1.3.2). This part of the algorithm was developed to satisfy the second objective of this research (RO 2). This code can be seen in Figure 10. The relationship between blocks is shown in Figure 11. Blocks are explained as follows:

1) This part is responsible for reading the elements' IDs and their selection time from the generated CSV file which was generated by block 2 by IFC archiver algorithm (Figure 8). Comparison of two IFC files. The core of this part is the "export IFC differences into excel2" which is numbered as 5.
2) It imports and reads all IFC files from the IFC archive.
3) It creates sets of consecutive IFC files and feeds step 4. Each set contains two contains two consecutive IFC files.
4) It works as a running loop for sets of consecutive IFC files to run block 5 (logging activities into the event log)

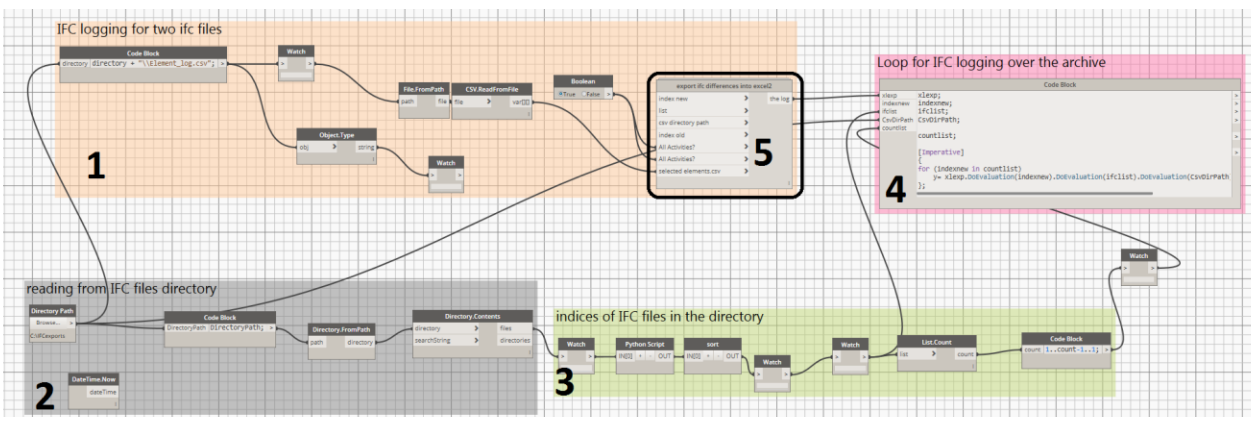



**Figure 10.** IFC Logger algorithm

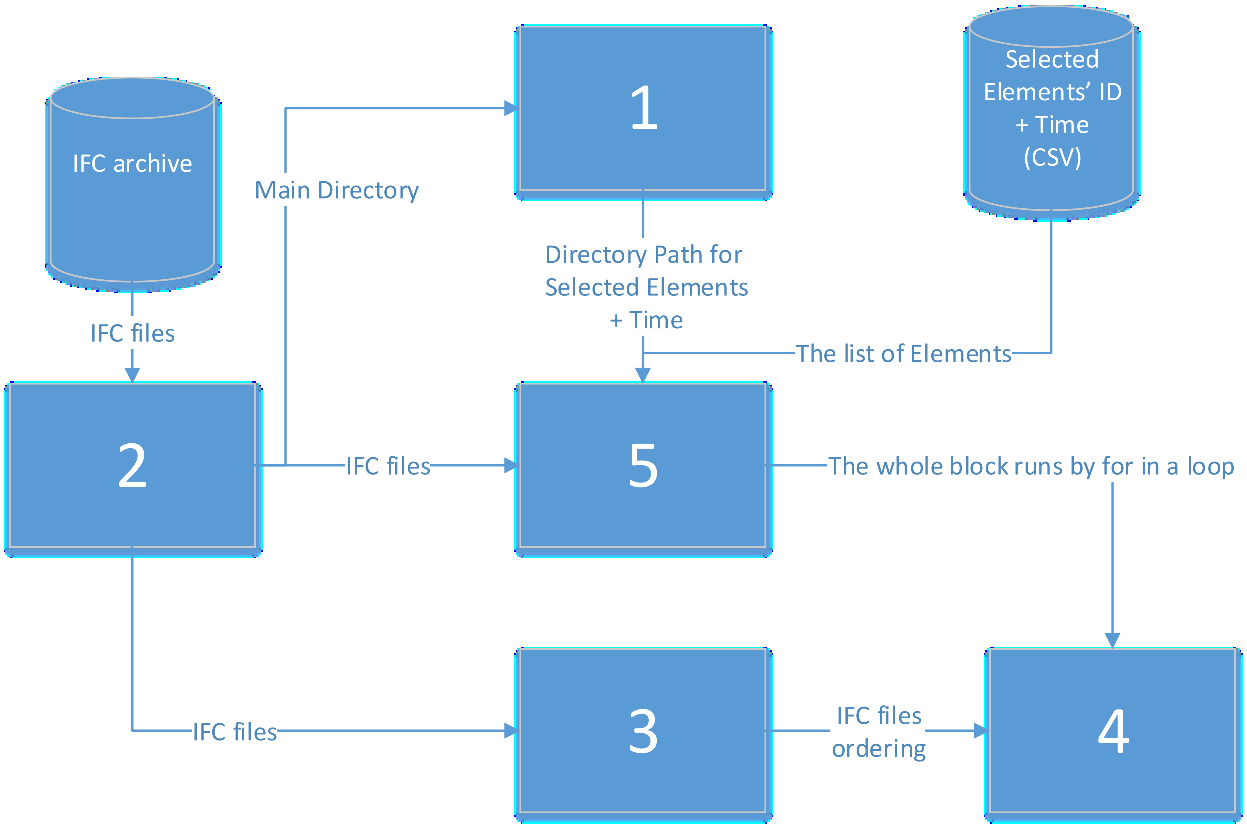



**Figure 11.** IFC Logger algorithm- the relationship between blocks

5) It is a custom node (a part of step 1) which itself includes different parts shown in Figure 12. In blocks I, f, g, j, the regex search algorithms (for "relocation and rotation" and "property changes") which were used for comparing IFC files are illustrated in Figure 47 and Figure 48 (Appendix 3). Also, the relationship among blocks is illustrated in Figure 13. In addition, a description for each block is provided as follows:
   a) This block is responsible for getting the two consecutive IFC files ("old IFC file" and "new IFC file") from the IFC archive.
   b) This block finds the elements in the IFC files (step a) (which is basically a text file) by using "\u0028\S{23}" regular expression in the GUID based search algorithm.
   c) This part is responsible for getting the curtain wall elements by using "\u0028.*?\n" regular expression in the search algorithm. Since GUID for IFCCURTAINWALL element in IFC tends to vary and is not stable within different IFC files of one BIM model, we used Revit ID (which is fixed) as a trustable signature for IFC comparison.

d) This part is responsible for comparing the list of IDs in two IFC files (output of step b and c) and export the "add and remove" type of activities into the event log.
e) It imports the selected elements and their selection time. It narrows the search query to only the selected elements and, henceforth, optimizes the code. (input for steps i, f, g, and j)
f) text mining pre-work for relocation and rotation detection in the old IFC file (the outputs are the direction and location of the elements in the newer IFC file which considered as inputs for step h and k)
g) text mining pre-work for relocation and rotation detection in the new IFC file (the outputs are the direction and location of the elements in the older IFC file which considered as inputs for step h and k)
h) It compares the direction of the elements in the old (output of step f) and new (output of step g). If they are not equal, it exports a rotation activity into the event log.
i) This combination of nodes gets the property information of the elements in the old IFC file. (the output of this step is one of the inputs of the step j)
j) This combination of nodes gets the property information of the elements in the new IFC file. Then it compares the property information of the element in the old and new IF
k) it compares the location of the element in the old (output of step f) and new (output of step g). If they are not equal, it exports a relocation activity into the event log.
l) It finds the designer information from the new IFC file and set it as "designer" attribute for every type of activity in the event log.

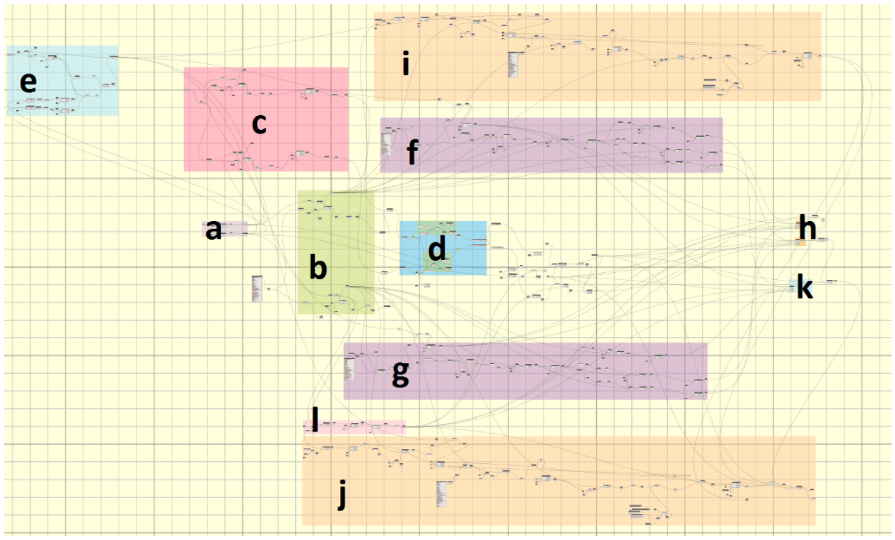



**Figure 12.** IFC files comparer node (node 5) to export the differences between two consecutive IFC files into the CSV format event log

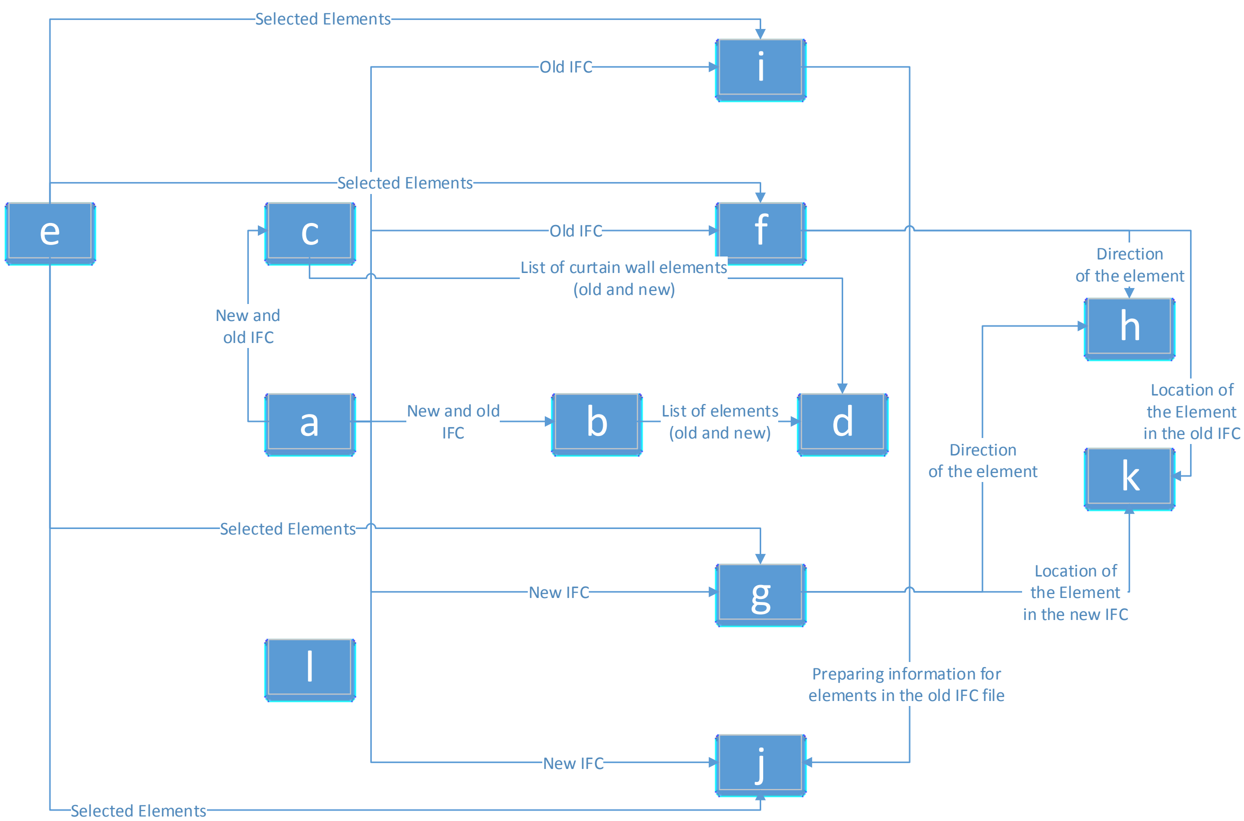



**Figure 13.** IFC comparer node- relationship among blocks of the custom node #5

### 3.3.2. Duration of activities (related to RO 2)

This part of the methodology explains an error which is detected and resolved in the IFC Logger (RO 2). The problem is associated with the calculation of activities' duration. Also, it was discovered during the implementation phase (second case study) and was fixed subsequently. Here I use two pairs of terms to explain the issue: the "New IFC Changed and Old IFC", and the "second and first" IFC file. The former couple is related to the IFC archiving stage, and the latter one refers to the IFC Logging phase. Figure 5 (a) illustrates that if the new generated IFC file has the same size as the previous IFC file, the "New" IFC would be erased from the database. After IFC archiving phase, two consecutive IFC files ("New and Old IFC files) were compared with each other, and if there are any differences in them, the differences would be detected as a set of activities such as "wall is added" or "the volume of the footing has changed". These activities had different attributes including start time and end time. The start time was considered as the creation time of the "first" one in "comparison stage" which is equal to the creation of the "Old" file in the archiving stage; since we removed the newest unchanged IFC file. As discussed, sometimes due to the idleness of the designers, the "Old" file was replaced with the "New" one.

Sojourn time corresponds to the time a token stays in a place during one visit. In an ideal case, sojourn time would be "execution" time of the activities. However, In the case studies, often we had the "idleness + execution" time as sojourn time because of the mentioned error in the previous paragraph. These case studies helped us to determine this error. The error is fixed, and the corresponding dynamo algorithm (aka IFC logger algorithm) is improved.

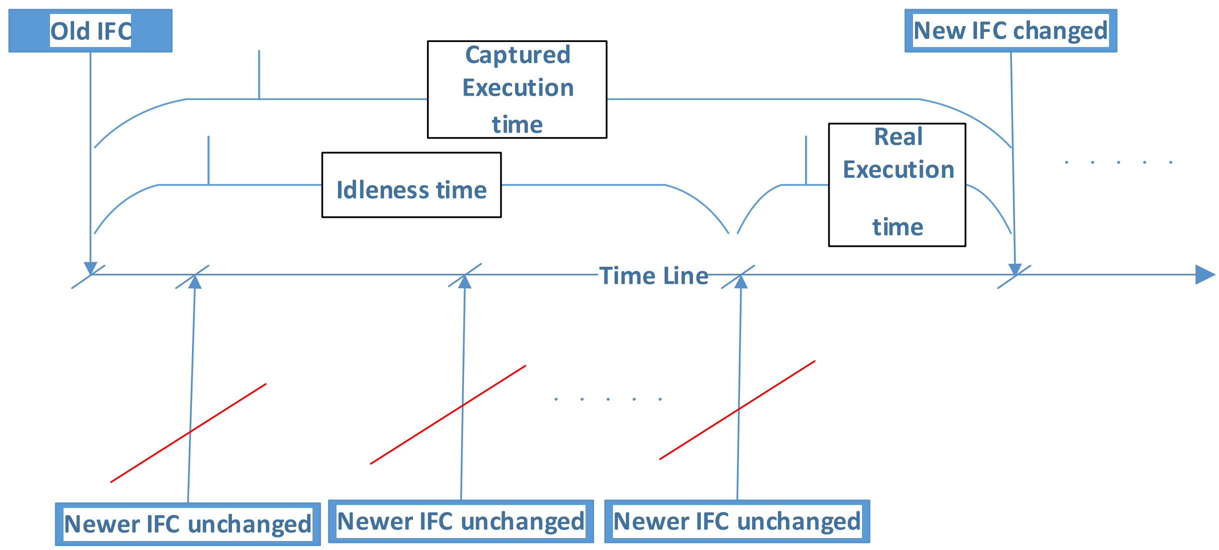



**Figure 14**. Error in duration calculation of activities

## 3.4. Discovery of design authoring process (RO 3 and 4)

The XES format Event log was used as an input for Process Mining techniques to discover the underlying processes happening in the design phase. The Process Mining algorithm which is applied in this phase is the inductive mining algorithm. In the literature review, this algorithm compared with other methods and selected as the most appropriate method for the scope of this project.

Additionally, social network analysis algorithms would be applied to the event log to capture the collaboration issues in the project. The techniques which are applied for this section was the handover of works matrix, the subcontracting social network mining and working together. These methods are available through specific modules in ProM.

Moreover, deviations, bottlenecks, sojourn time, service time and queue length in the process can be detected by replaying the event log on the mined process in ProM. Then, the result of this action would be shown as animations. These animations illustrate the project timeline at a smaller scale. For instance, a project which is performed within one month would be shown in one minute in the animations. Hence, the animation can show which activities are taking longer than usual. These longer-than-usual activities, aka bottlenecks, are the ones which require more investigation to find their causes. Later, the causes may be fixed to remove the bottlenecks.

Process deviations have two definitions. Once the as-planned process model is available, and once it is not. Regarding the former definition, if there were a predefined level 3 BIM maps, then after the discovery of the process model with the proposed solution in this research, it would be possible to compare them to find different types of deviations between the planned and happened processes. However due to unavailability of the level 3 BIM map for design authoring, this definition is not the case here, it can be investigated by the future researchers. The second definition is about the granularity settings of the model. The granularity is set by the BIM manager in ProM to eliminate the less frequent activities and paths in the model. The difference between the filtered model and the primary model is defined as the second definition of process deviations. This, in fact, shows how the actual as-happened process is different from the process model identified and visualized. Only with having both the process deviations and the filtered model, BIM manager will be able to minimize the risk of her/his decisions.

## 3.5. Use-case diagram of the system

This section aims to provide guidelines for the implementation of the research methodology and its products. The methodology of this research is divided into four main sections (Figure 15): (1) Design Authoring, (2) IFC Archiving, (3) IFC Logging and (4) Process Mining. The use case diagram shows that in design authoring stage, Designers (architects, structural engineers, and mechanical engineers) create the BIM model using Revit. In parallel with the design phase, the BIM manager runs the IFC Archiver algorithm within the Dynamo environment (described in section 3.2). This algorithm captures snapshots of different stages of the model (in the format of IFC files) through the gradual development of the design (from A to Z). The time setting for creating these IFC files can be provided by BIM manager within the algorithm using block #1 in Figure 10. Consequently, it provides archives of different stages of the design authoring process for each designer at the end of the design phase (or at any point in the design phase). Also, IFC archiver algorithm produces a list of selected elements in another CSV file for each directory (described in section 3.2). Then, BIM manager uses the IFC archive and the selected elements CSV file to feed the IFC Logger algorithm in order to capture the gradual evolution of the BIM model (described in section 3.3). Subsequently, the outcome of the IFC Logger algorithm results in one CSV format event logs for each archive. Afterward, BIM manager merges the CSV files together to construct a master CSV file including all event logs. At the end of the IFC Logging phase, CSV files are used as an input to XESame to map the event log into XES. Then, XES event log is used by ProM to capture the as-happened process and other types of analytics such as social network analysis, bottleneck detection and finding the process deviations (described in section 3.3)

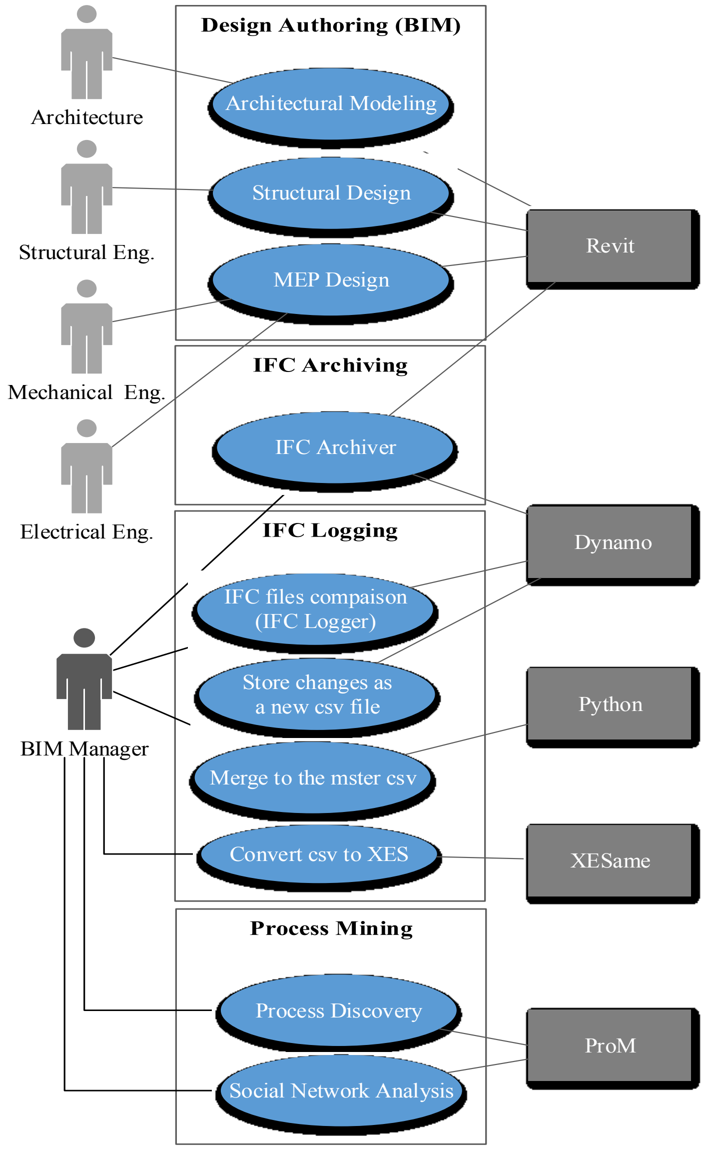



**Figure 15.** Use case diagram: it works as a manual for the user to implement the research methodology (BIM-based Process Mining for design phase) [2]

In this chapter two generations of the high-level workflow for this research scope were proposed: first-generation and the second-generation BIM-Process Mining integration model. Also, the dynamo algorithms which were created during the methodology were explained in detail. In the end, the use case diagram is provided as a manual for implementation of the workflows for the end users (BIM managers). At the beginning of the next chapter, I brought an earth moving operation simulation example to validate the bi-similarity of a simulated construction process and the discovered process. Next, I implemented the workflows in two case studies to verify and validate them.

# Chapter 4
# IMPLEMENTATION

Validation and verification were conducted through the implementation of the validation example and case studies. Verification is a process ensuring that the deliverables and the products satisfy the specified requirements which are done through technical review methods by conducting two case studies. Validation process attempts to check if the system deliverables meet the client's requirements [134]. Validation example was performed to check the bi-similarity of a simulated process and discovered process; it shows that the discovered process has similar features and capabilities as the simulated process. Next, the proposed workflow was tested by two BIM design authoring case studies. For the purpose of validation, we asked the designers, in both cases studies, to keep track of their work steps and timelines. The first implementation of our algorithm is shown in the case study #1. This case study was useful for verification (debugging the IFC archiver and logger algorithm) and validation (by comparing the analytics and observation records during the case study). The second case study was more complex and involved the improved workflow (second-generation) and updated dynamo algorithms which were capable of capturing more types of activities more efficiently. The outcomes of this case study were more informative and matched the real-life scenario more accurately.

In the proposed case studies (Case #1 and Case#2), each element considered as a trace. In the first approach, elements' changes are set as activities ("elements' changes as activities approach"). In the "designers/phases as activities approach", on the other hand, the designers/phases (Case #1/Case #2) are considered as activities. Each trace may consist of one or more events. For both scenarios, each event consists of four main attributes (extensions). Those included: trace ID, Event ID or name, Timestamp, and Resource. Traces would represent the different executions of the same process; in other words, each trace equals one instance of the process run. In the discovered processes, the Case ID would be named after the designer's role (e.g., Architect in Case #1) or the phase of the design (e.g., Arch SD in Case #2). Each step of the process is an activity, and it may consist of the end or start event (moment) of those activities. The timestamp extension saves temporal information related to the start or finish of the activities. The timestamps would be helpful for identification of the events order, detection of bottlenecks, and animating the event log over the generated process model. Resource extension is used for organizational and social network analysis.

## 4.1. Validation

In order to test that if the Process Mining can capture all the capabilities and features defined in the scope of projects, I designed and simulated a familiar process for professionals in the AEC industry, using EZstrobe and WebCyclone. The example is an earth moving operation of a construction project. It involves 21 trucks, 4 shovels. 2 scales, and one hopper. Using WebCyclone, I simulated and generated an event log for about 82 process runs of the model. The model in EZstrobe for more clarification is shown in Figure 16.

The event log is generated in WebCyclone website and modified in excel. A part of the final CSV file of the event log is shown in Table 2. It has 39,127 events, 82 traces, 28 actors (21 trucks, 4 shovels. 2 scales, and one hopper), and 5 attributes (Trace ID, Event ID, Actor name, Event Name, and Completion time). The process associated with the event log was captured using Process Mining techniques. The Petri net of the mined process was discovered and illustrated in Figure 17 to be compared with the first model which is shown in Figure 16. Theoretically, they should match each other perfectly.

Petri Net is a foundational process notation. It is a static network and consists of places, activity and tokens. Tokens can be in places and be produced or consumed in activities. An activity is enabled when each one of its input places contain at least one token (the required number of tokens is equal to the number of arrows comes to the activity). Firing means consuming and then producing tokens based on the number of arrows going out of an activity. Activities are shown as rectangles; the empty ones are dummy activities which are meant to satisfy one of the main rules of petri net structure; the rule is “two places cannot be connected directly with an arrow”. [67]

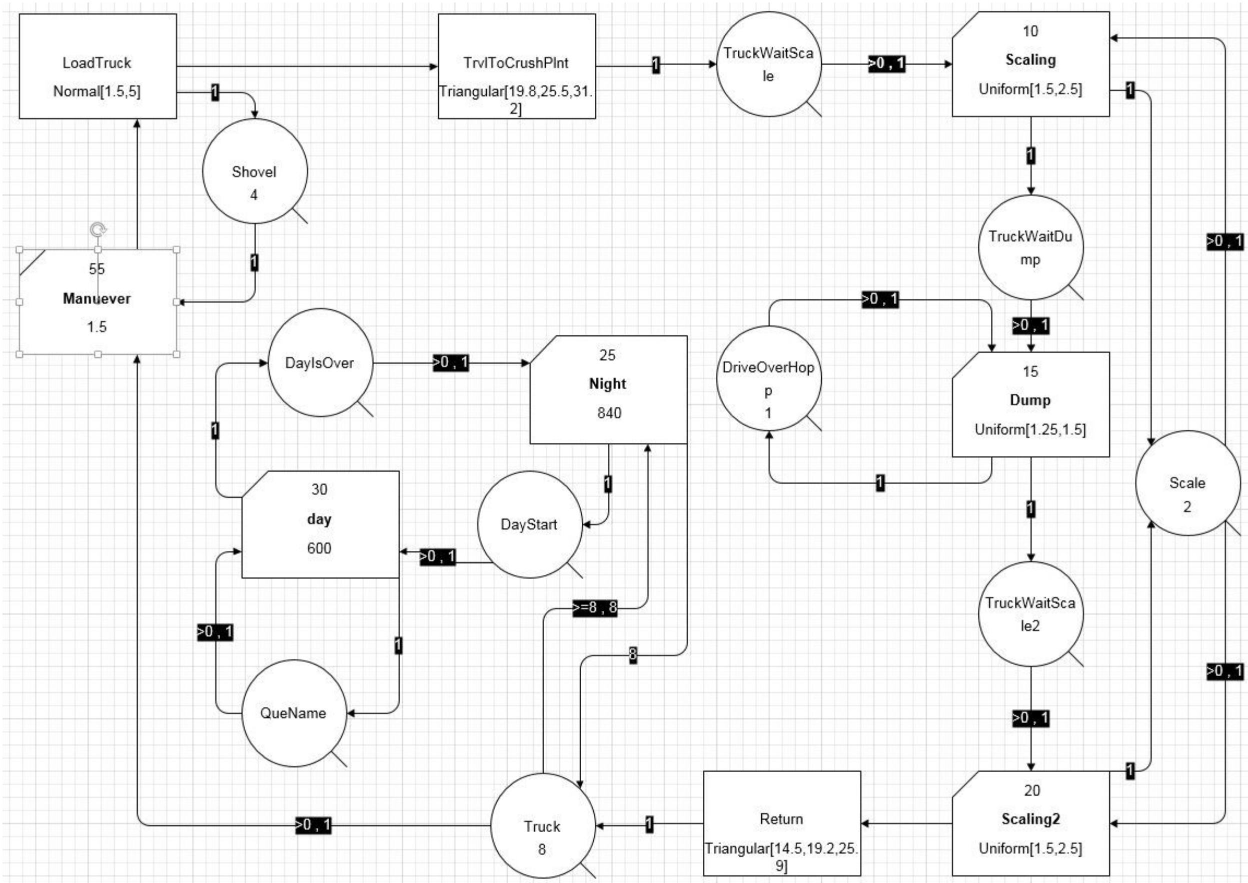



**Figure 16.** Petri net of the example- an earth moving operation (modeled by EZstrobe stencil )

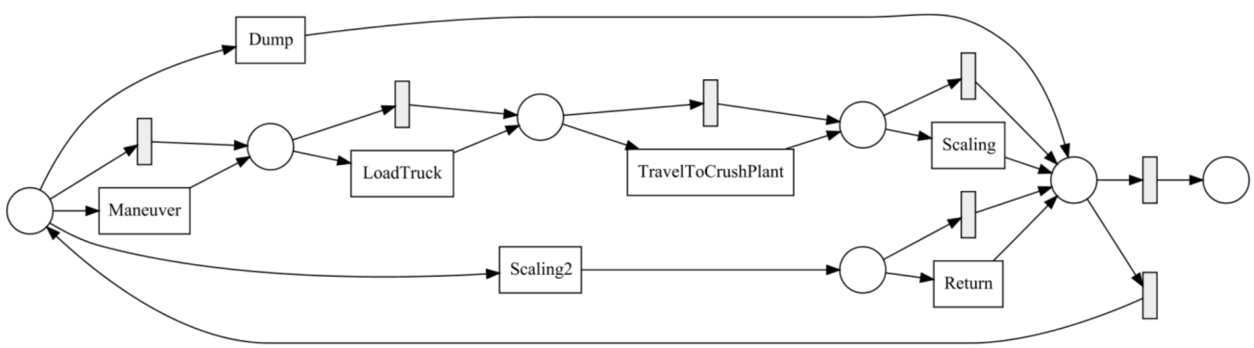



**Figure 17.** The mined process of the example using ProM

**Table 2.** Event log for the validation example- each event has Trace, Event ID, Actor, Event Name, and Time attributes.

| Trace | Event ID | Actor1 | Event Name | Time |
|---|---|---|---|---|
| 1 | 1 | Truck1 | Maneuver | 1.5 |
| 1 | 2 | Truck1 | LoadTruck | 1.5 |
| 1 | 2 | Shovel1 | LoadTruck | 1.5 |
| 1 | 3 | Truck1 | TravelToCrushPlant | 25.1 |

I compared each node and activity to show that it is the same as in our simulation model. M, L, T, S, S2, D, and R are used as abbreviations for Maneuver, LoadTruck, TravelToCrushPlant, Scaling, Scaling2, Dump, and Return activities which are shown in Figure 16 and Figure 17. Seven scenarios are investigated to illustrate how the mined process is the same as our simulation model (bi-similarity checking). Each scenario explains the situation where an actor or a set of actors involved in one particular activity execute the process.

Maneuvering of a Truck from the waiting line to an appropriate place adjacent to the shovel is the nature of activity 'M'. This activity can only be done when one shovel is available and free, and this only happens after completion of a LoadTruck (L) activity. The activity 'L' requires the presence of a truck and a shovel at the same place. To check the bi-similarity of the two processes, the event log was animated again, on the mined process (Figure 17). By selecting activities 'M' and 'L' during the animation, the involved resources (trucks and shovels) mimicked the same behavior on the as-happened model. The same type of investigation was conducted on the remained activities including TravelToCrushPlant (T), Scaling (S), Scaling2 (S2), Dump (D), and Return (R).

From the comparison of the as-happened and as-planned processes in this example, it can be concluded that:

a) Testing the Process Mining inductive miner algorithm with the simulated construction process proved the inductive miner to be accurate, trustable and verified
b) The process can be mined and shown in any format (e.g., Petri net and BPMN). The point is that one should choose the right format based on the project requirements.

c) Although the purpose of Process Mining is to capture the process behavior out of real-world data, this small experiment tried to describe the right type of event log that should be captured for Process Mining on the realm of AEC. Consequently, the lessons learned from this example were used as a guideline for developing the logging algorithm (Figure 10) and conducting the case studies #1 and #2 for capturing the right event log.

Next, we tested the system implementation by performing two real case studies (Case studies #1 and #2) for verification and further validation of our BIM – Process Mining integration model. While performing the first case study, verification of the algorithm was done through capturing some internal errors of the workflow (Figure 5). As a result, the errors in the workflow were debugged, and the algorithm was improved. Next, the second case study was done based on the improved workflow which is illustrated in Figure 6. The second case was more complex and involved more actors to mimic actual requirements of a design project for validating the system. The implementation results of our methods are presented in the following sections. Mainly, process discovery and social network analytics are highlighted for each case. Validation and debugging some errors for IFC logging algorithm was done through performing the case studies. For validating the system, we asked the designers to track the record of their steps and elements they create; and at the end, we checked whether the system was able to capture the same sequence of steps.

## 4.2. Case study #1

In order to apply and test the system, we studied the process of designing a 2-story building using Revit, involving one actor (myself) who played the role of five fictitious modelers including two architects, two structural engineers, and one MEP engineer. Within the explanations in this case study, "designers" refers to the fictitious designers. The general purpose of the first case study was to test different aspects of the workflow, debug the IFC archiver and logger algorithms, and mainly perform the verification of the system. During the design phase, the IFC archiver was running periodically (every 10 seconds), checking if there were changes in the size of IFC files. After completion of the design, every two consecutive IFC files generated were compared, and the differences were saved into a master CSV file. Then, the generated CSV event log was imported to ProM to produce the XES file using the XESame module. Finally, the process discovery and analytics resulted in finding a social network of the project and process deviations.

### 4.2.1. Process discovery

Process discovery for the first case study is performed in two different ways. In the first approach ("Elements' changes as activities approach"), the elements were set as traces of the process; in other words, each token in the discovered Petri net (or each dot in the process animation) represents one of the elements. In this approach, detailed activities are modeled in the process as "element name + type of change applied". With these settings, the discovered process can be considered the as-happened "level 3" BIM map for design authoring. The second approach ("designers as activities approach") is proposed to discover the as-happened level 2 BIM map. These types of maps are considered as BIM maps with higher-level of abstraction and fewer details compared to the "Elements' changes as activities approach" ("level 3"). The activities correspond to design authoring phases.

*Elements' changes as activities approach: "level 3" as-happened BIM map discovery*- In this approach, the event log contains 299 different traces (each trace corresponds to one element) and 310 events (40 unique events). In this case, each event corresponds to the end point of an activity (since we only captured the activities' end time). Also, each event contains 4 variables including event name, event ID, end time and resource. Discovery of the underlying process ("level 3") through the inductive miner module resulted in what is depicted in Figure 18. Then, the event log was played on top of the discovered model to construct the process animation; a snapshot of the animation is depicted in Figure 19. It involves every event that occurred during the design authoring, regardless of their occurrence frequency. The number in each box shows the frequency of changes for a specific type of elements which are passing through each box (activity) within the project timeline. The discovered process is unfiltered and can be considered an overfit process (too case-specific); in other words, the discovered model just represent itself. Therefore, in order to make the process a good representative of the future processes, BIM manager may apply some levels of filtering to reach the model he has in mind by eliminating the deviated paths and activities. While "paths filter" is responsible for filtering the infrequent process sequences, the "activity filter” removes the infrequent activities; a lower number corresponds to a more level of filtering. Activity filter and path filter were set at 0.6 and 0; in other words, eliminating the infrequent paths while trying to keep the activities with at least 40 percentile frequency). Then, the deviations analysis was conducted to show the deviated process on top of the filtered model; the reason for deviations is a simplification of the discovered process model. An example of the deviation in the process model can be found as red dashed lines in Figure 20. For instance, it shows that the task “IFC member removed” was not executed for 65 times (out of 71).

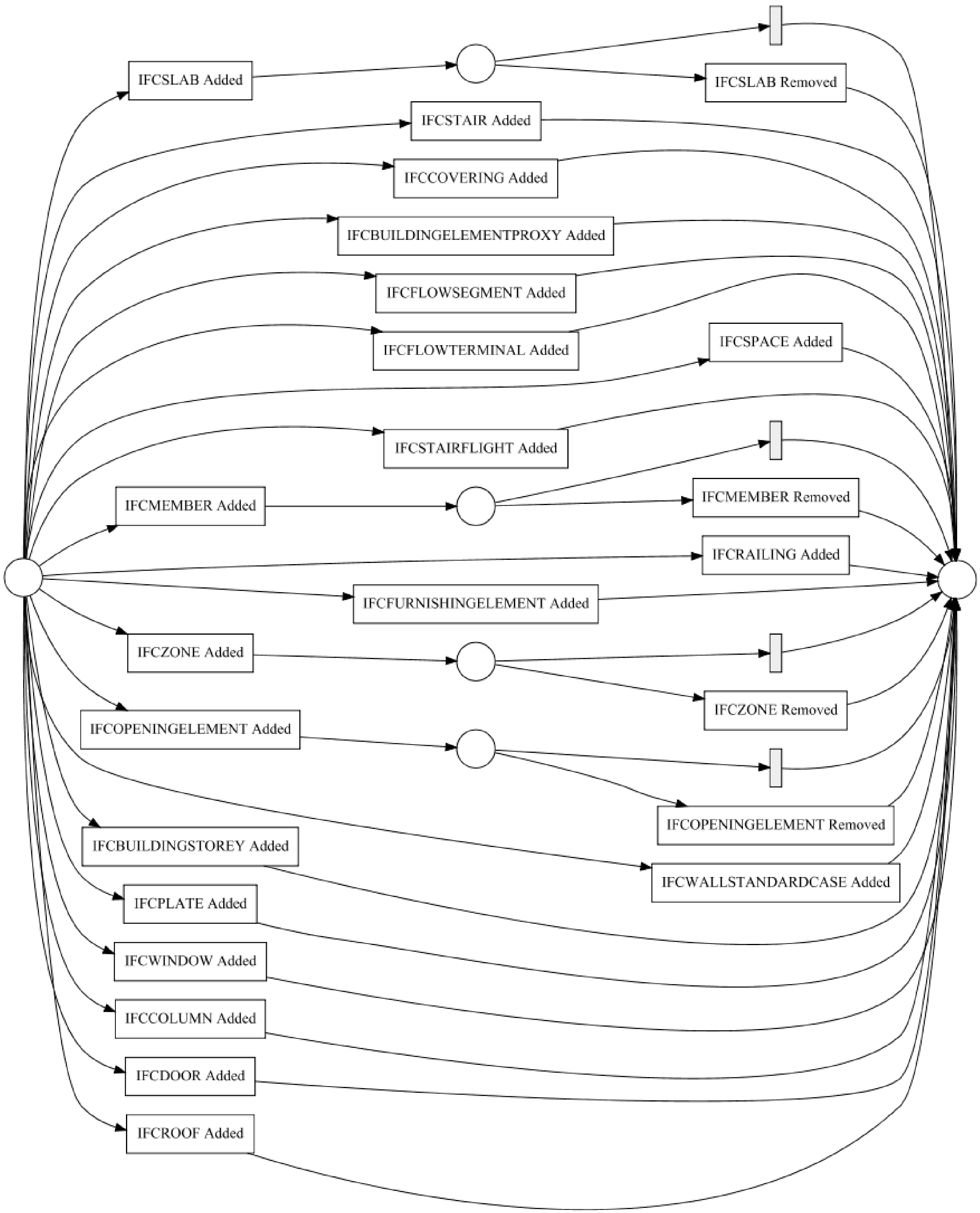



**Figure 18.** The “level 3” as-happened Petri-net of the process of the case study #1

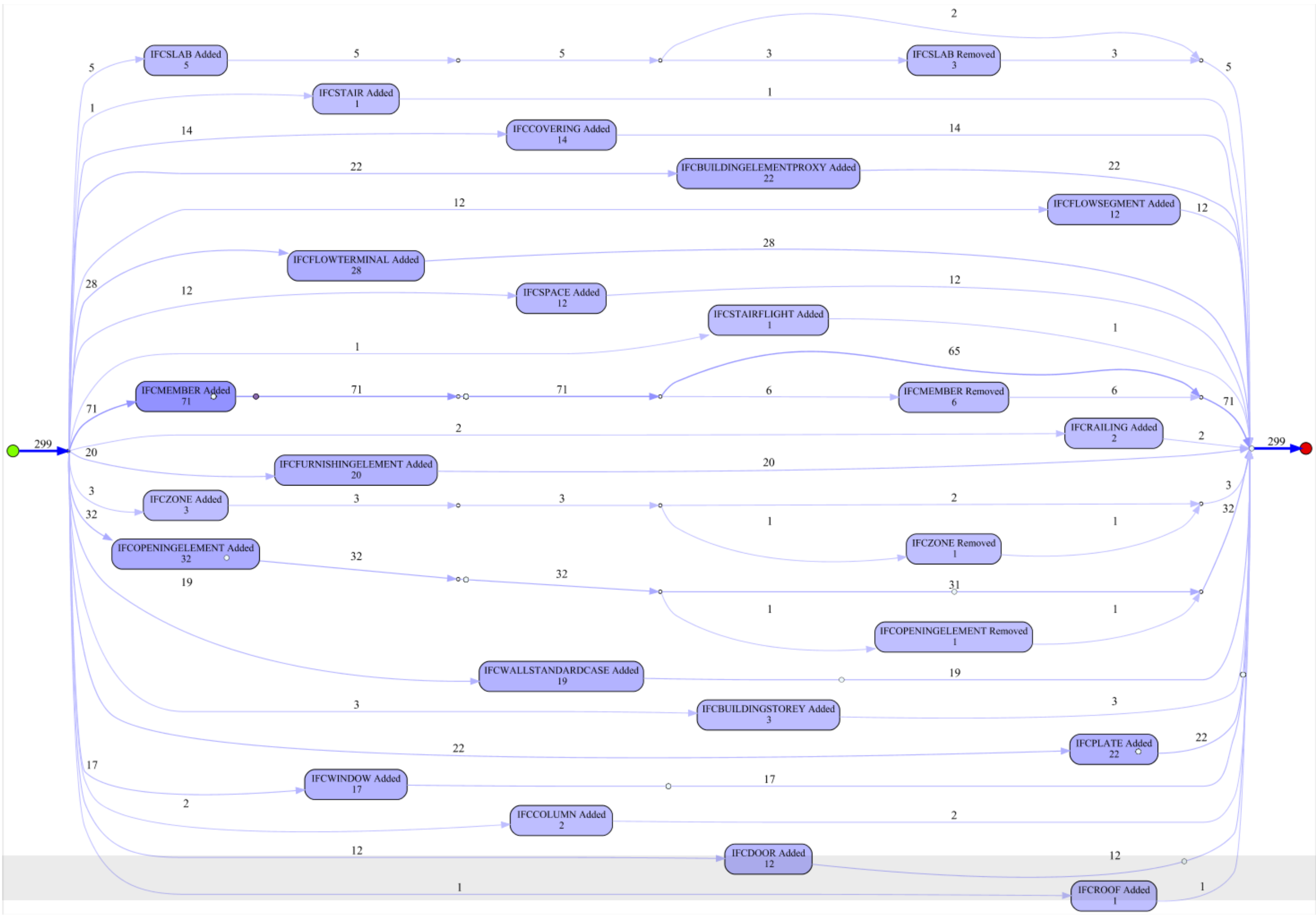



**Figure 19.** “level 3” as-“level 3” Process animation (as-happened): the number in each box shows the frequency of changes for the elements which are passing through each box (activity) within the project timeline- Case study #1

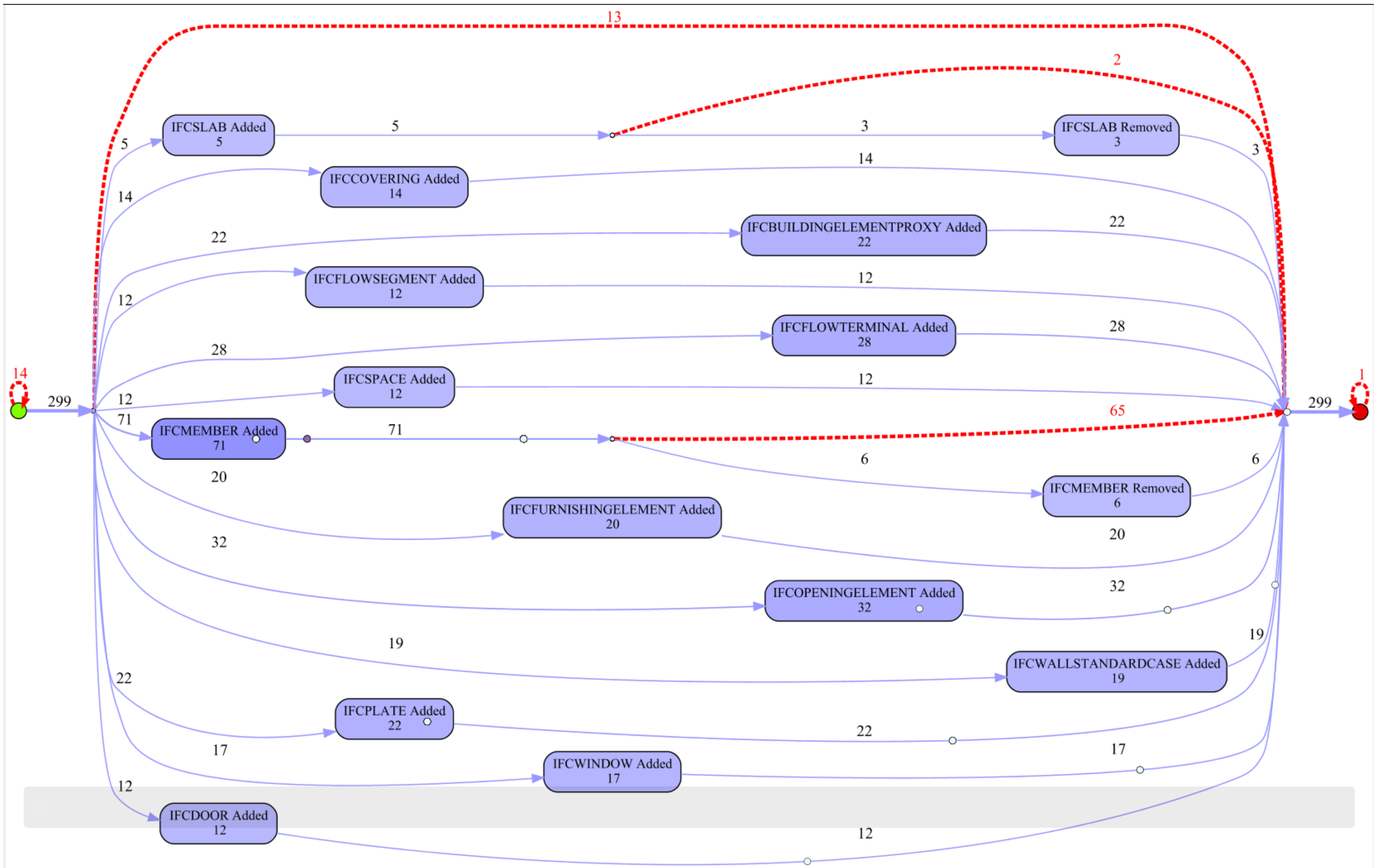



**Figure 20.** An example of process deviations existing in the process of the case study #1

*Designers as activities approach: level 2 as-happened BIM map discovery-* The event log for the first case study has 299 cases (which correspond to unique elements) and 310 events. The events are the end moments of activities such as "IFC wall added". The "Event per case" window classifies the number of events per each case; i.e., it classifies the elements (cases) based on the number of events which has happened for each element (case). For example, most of the elements have been changed once (1 event). However, there are a few elements in the log which are associated with two activities (2 events). Figure 21 is a graph categorizing activities based on different designers. For example, activities "IFC COLUMN Added" were performed by architect #2 and structural engineer #1. The project path can be seen in Figure 22. It shows that the project was mainly started with the involvement of architects #1 and #2, at the beginning. Next, MEP joined them to complete his own part and then the design continued by architect #1 and structural engineer #1. Then, the work was handed over to architect #2 followed by the involvement of MEP modeler and structural engineer #2. In the end stage, the work was finished with some minor contribution of architect #1, architect #2, MEP, and structural engineer #2.

The level 2 as-planned BIM process map for design authoring was provided in Figure 4 (chapter 4). Figure 23 is the as-happened BIM process map of the same procedure; the only difference is that, instead of phases, the designers are considered as steps (or activities) in this figure; the designer's names represent activities completed by them. It was mined by the proposed workflow of this research (BIM-based Process Mining workflow). The as-happened map shows the real procedure more accurate compared to the as-planned process map. The process tree for elements in the process is shown in Figure 24. It merely illustrates the logic of the elements (traces) in the process. For instance, there are elements which were changed just by structural engineer #2 (the middle branch of the tree). The left "xor" in this figure shows that some elements were either changed by structural engineer #1 in a loop (several modifications by him) or architect #2 in a loop (several modifications by him). Next, the reality (event log) was played on top of the discovered model; a snapshot of the animations is illustrated in Figure 25. The number in each box shows the number of the step execution (each execution corresponds to change a characteristic of an element which passes through that step) by each designer throughout the project.

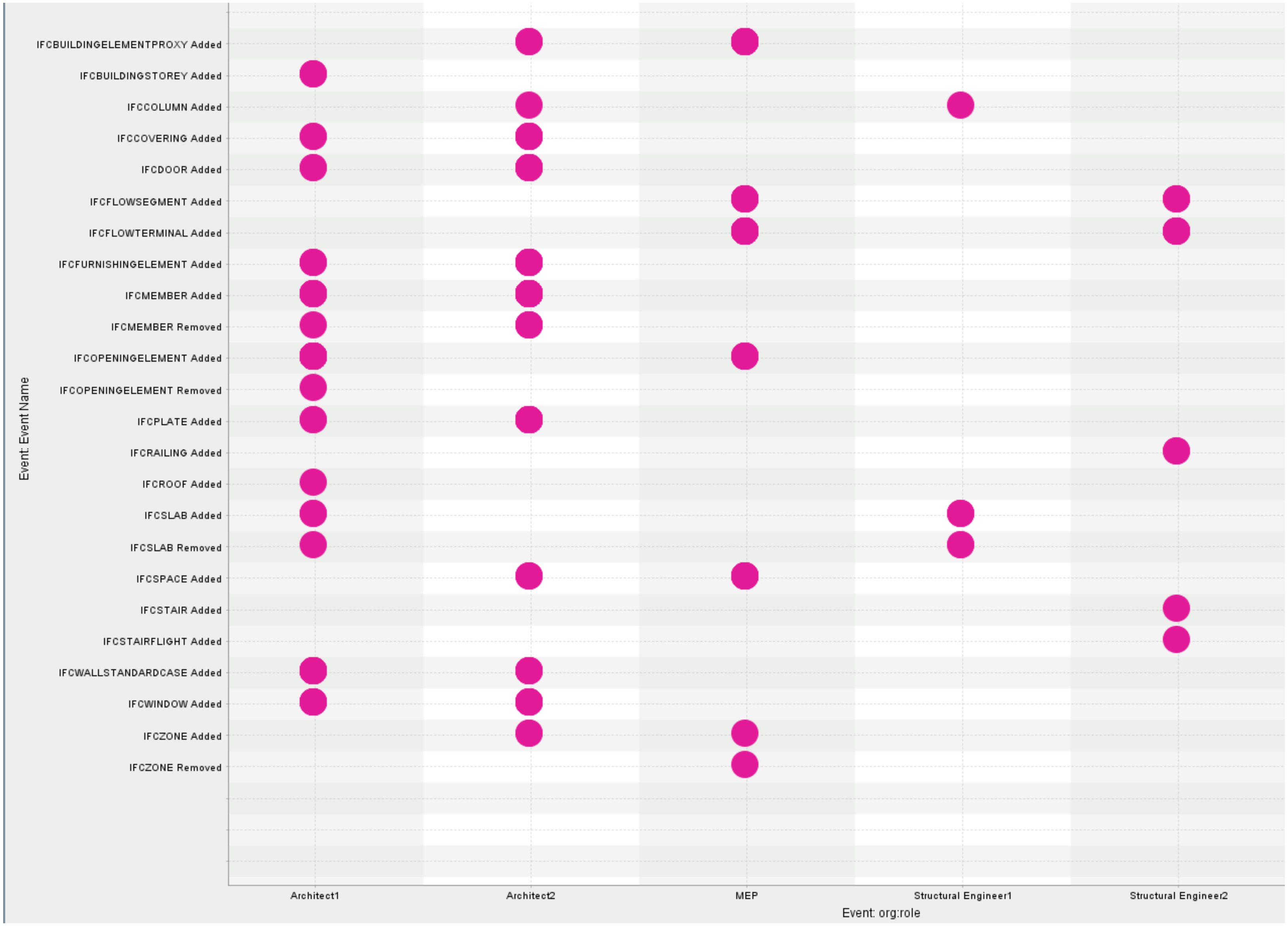



**Figure 21.** Designers based on activities they are involved in- Case study #1

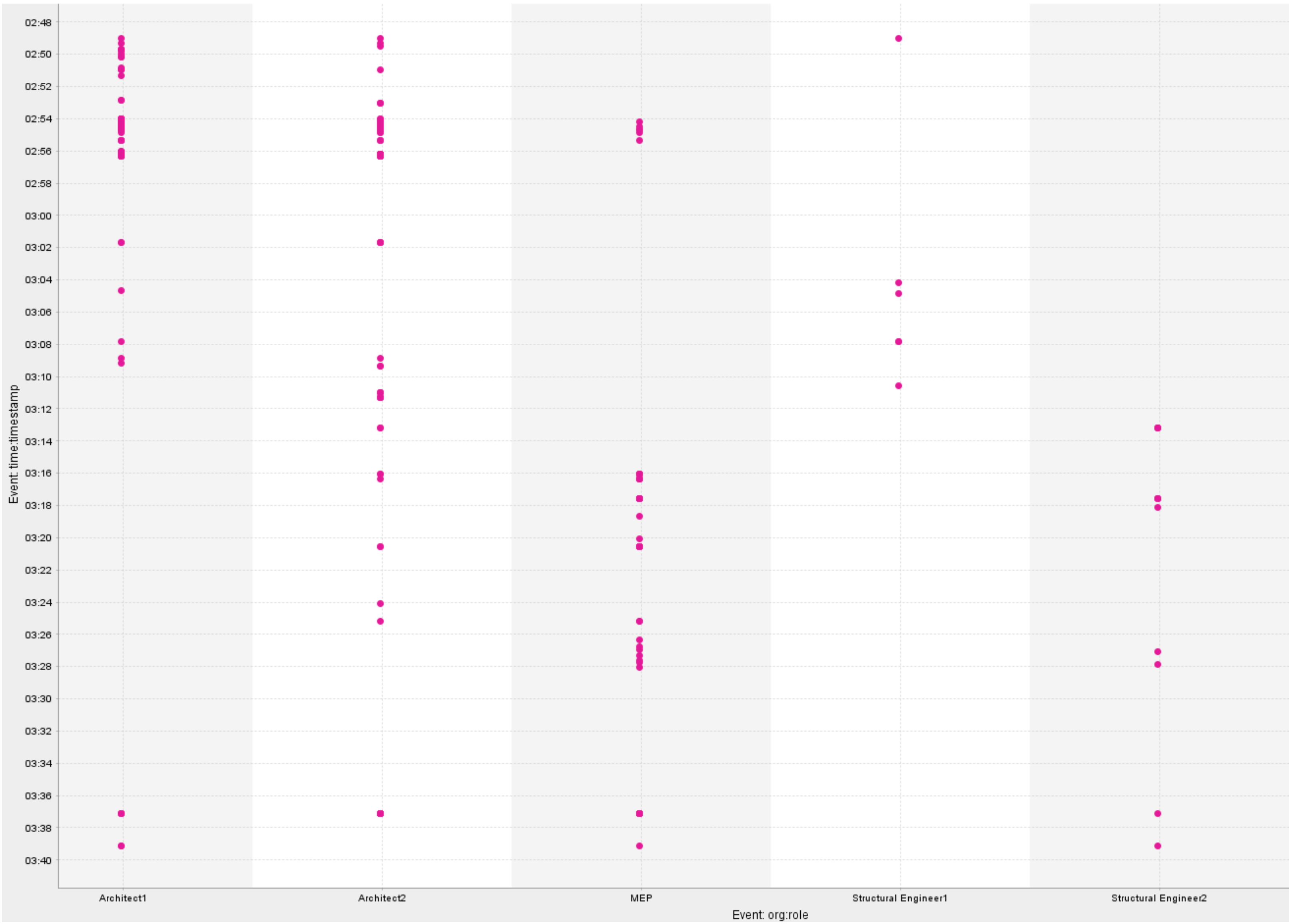



**Figure 22.** Design phases based on timestamp- Case study #1

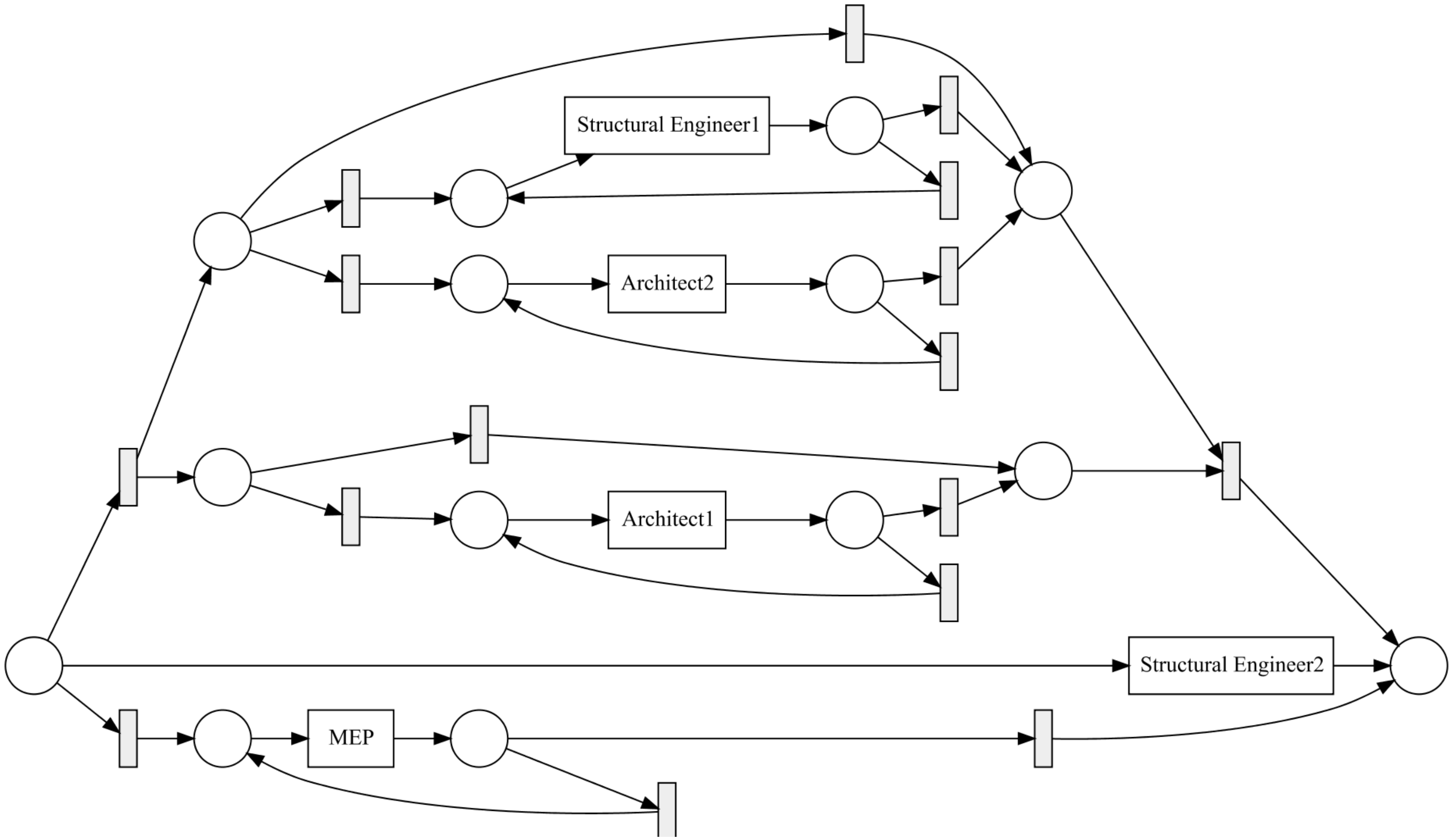



**Figure 23.** level 2 Petri net (as-happened)- Case study #1

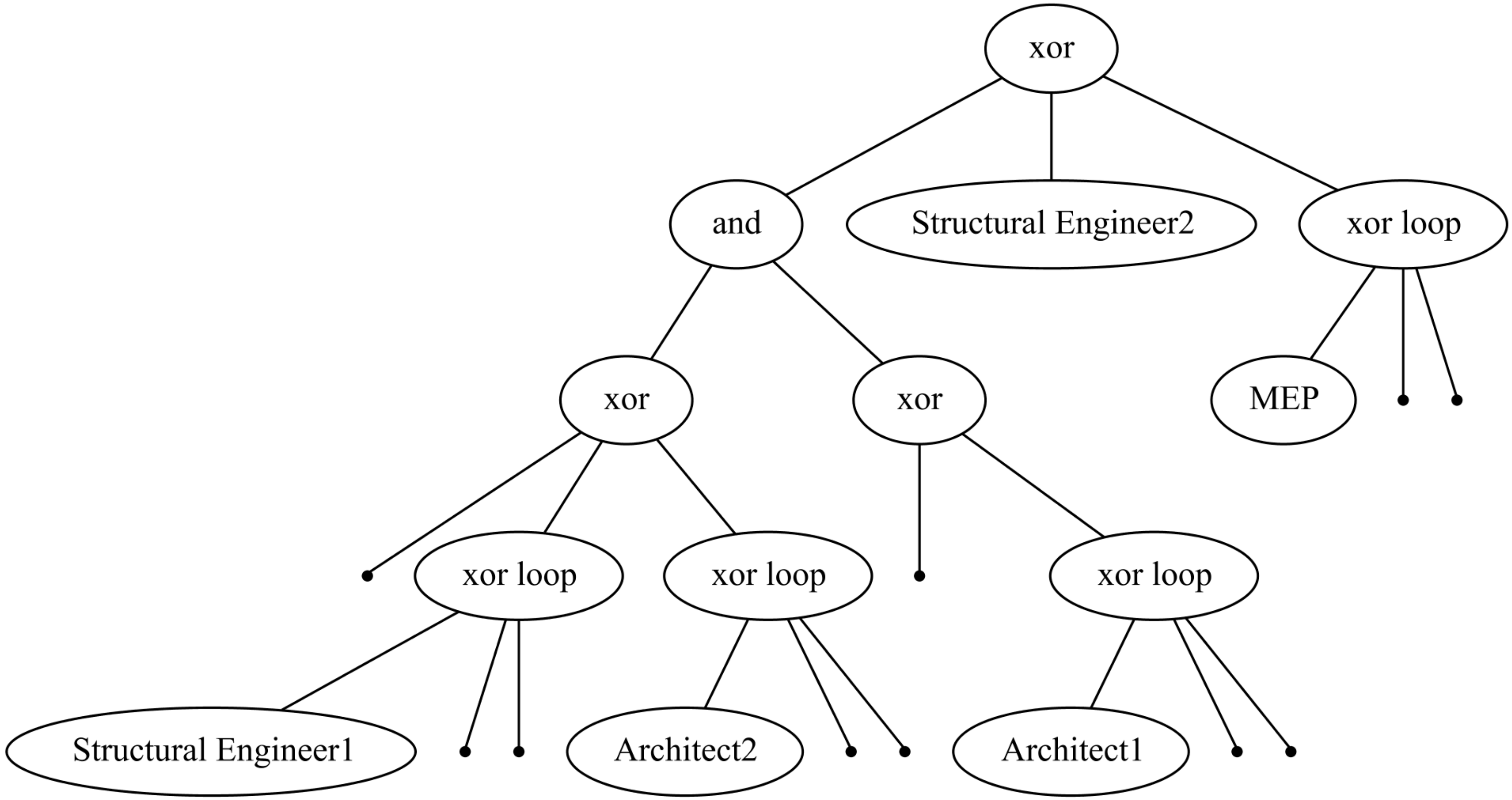



**Figure 24.** level 2 process Tree Graph Visualization- Case study #1

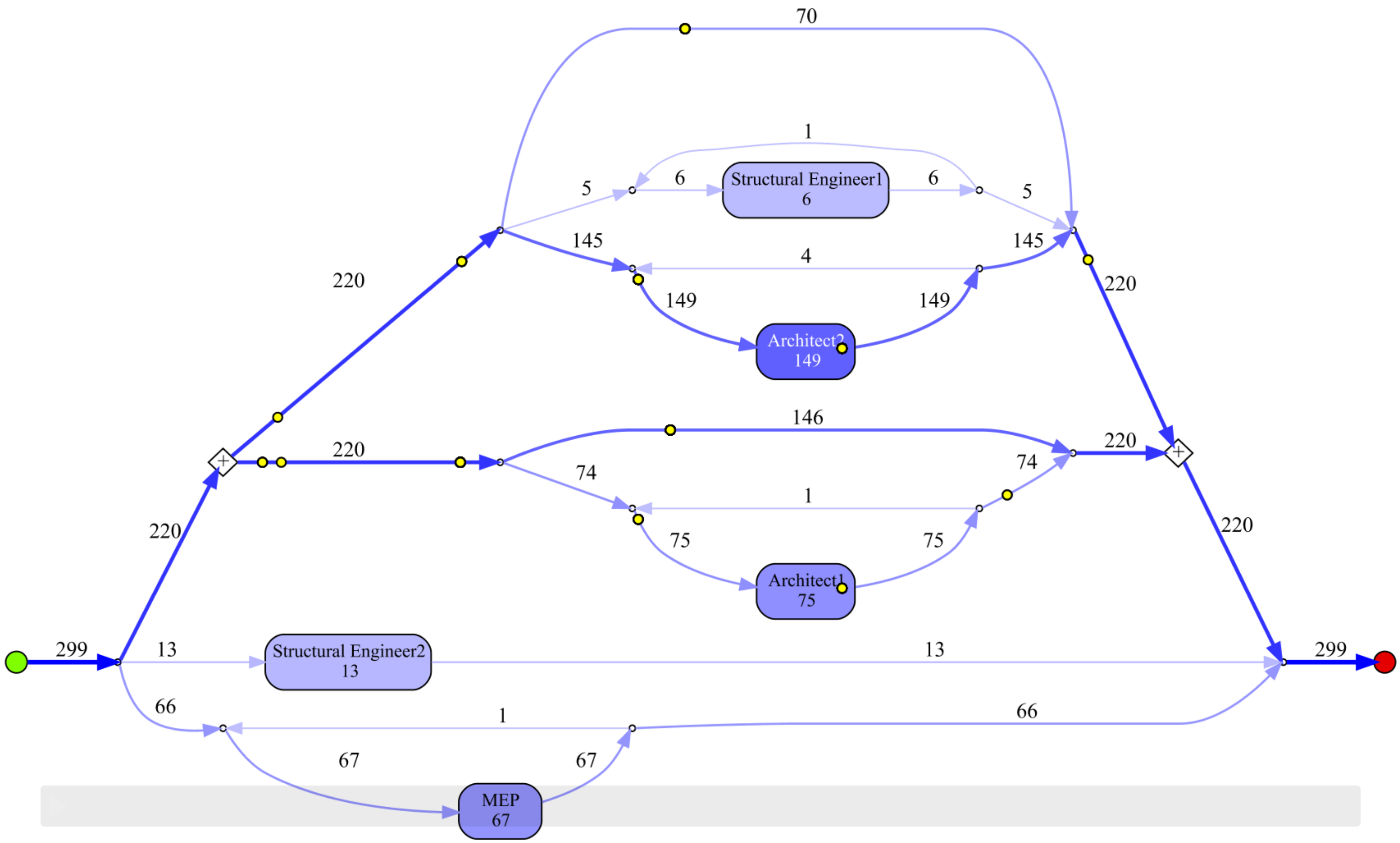
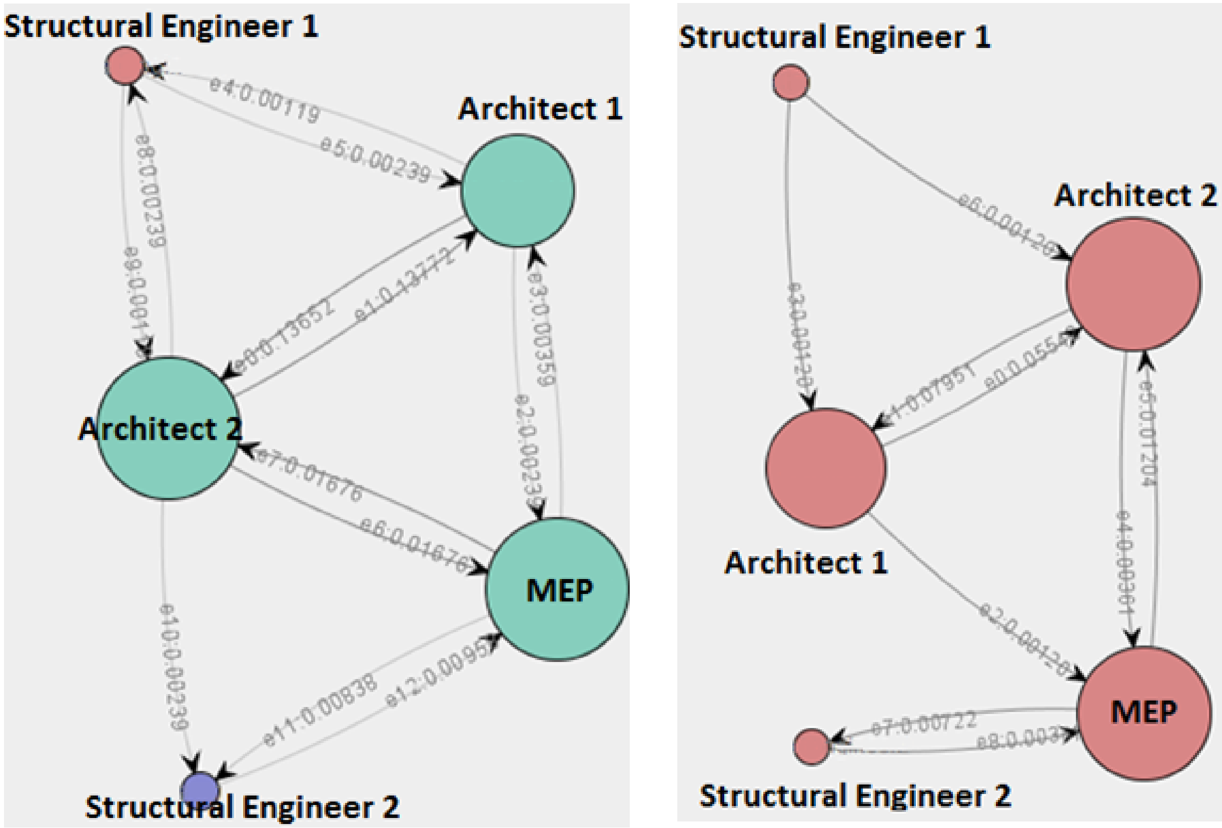



**Figure 25.** level 2 process animation: the number in each box shows the frequency of changes for the elements which are passing through each box (activity) within the project timeline - Case study #1

### 4.2.2. The analytics

"Figure 26 (a) and (b) show the social networks formed respectively based on "handover of work matrix"[1] and "subcontracting"[2]. Each color represents one class of actor; the members in each class have more handover of work among one another. The bigger nodes are more strategic and represent the higher "Betweenness"[3] in the social network. Further, the values on the edges are the strength[4] of the connections between every two nodes. The "Handover of Work" figure, shows the resources in three different colors and different sizes. To illustrate, architect 1 and 2 and MEP (represented as green) are collaborating more frequently and forming a strong community. Moreover, MEP, Architect 2 and 1 are the most strategic roles. In the "Subcontracting" figure, structural engineer 1 hires architect 1 and 2 as subcontractors to execute some of her/his tasks. Based on this method, all actors are considered as one group; i.e., all actors are extremely dependent on all others in the execution of activities so that they cannot be categorized as different communities. Just as the first method, the strategic actors are architect #2 and #1 and MEP. Furthermore, in both figures, there is no link between the two structural engineers which is somewhat unusual for a design authoring BIM project. Where this is a real case, the reason should have been investigated more closely by the responsible party." [2]

[1] "Within a case (i.e., process instance) there is a handover of work from individual *i* to individual *j* if there are two subsequent activities where the first is completed by *i* and the second by *j*." [142]
[2] "The main idea is to count the number of times individual *j*executed an activity in-between two activities executed by individual *i*." [142]
[3] "A ratio based on the number of geodesic paths visiting a given node" [142]
[4] "The strength of the link from person p1 to person p2 is calculated based on the number of times one activity of p1 is followed by and activity of p2 for the case." [142]




(a) Mining for Handover of work matrix Social Network

(b) Mining for Subcontracting Social Network

**Figure 26.** Social Network of the case study #1 [2]

### 4.2.3. Filtering the discovered process based on roles ("level 3")[5]

There are different types of design specialists involved in the design phase of construction projects: Architects, and Mechanical/Structural/Electrical Engineers. Previously, the authors (I and Dr. Nik-Bakht) proposed a method to construct a BIM related XES log by comparing consecutive IFC files captured during the design phase of a project. Then, the as-happened process using Inductive Miner algorithm was discovered. Although the idea of having a whole as-happened design authoring process is intriguing and might have insights for a BIM manager such as design team collaboration, having specific processes for each group of designers (e.g., architects) gives more accurate information on the as-happened process. In the previous discovered processes, the traces were elements. But in this section, the designers are considered as traces. In other words, the discovered process tells the story of the sequence of activities from the actors' point of view.

To have different processes for different actors we use "pre-mining filters" on the primary event log to divide the whole design authoring process to Structural, Architect, and MEP processes. Here I will explain the main activities sequence, during the MEP modeling. Figure 27 shows that first, the MEP team created opening elements. Second, the MEP team assign spaces and zones for energy analysis purposes. In the next step, they created flow terminals and flow segments. In parallel, the team added mechanical and electrical equipment which are all categorized as building element proxy; this is a Revit limitation when it comes to mapping MEP elements to IFC schema.

Figure 28 shows that one of the structural modelers created columns and slabs and then removed some of the slabs (a rework loop), sequentially. While the second designer sequentially added rails, stairs, and stair flights. Then, the first modeler incorporated in adding the "flow terminal" and "flow segments" which mainly must be within MEP modeler tasks; it might show a close and healthy communication between the two actors or it may show they are doing things outside of their own scope. Also, this collaboration is shown and explained in the social network of the process (Figure 26).

[5] This section is part of a conference paper published in ECPPM 2018. The authors are Sobhan Kouhestani and Dr. Mazdak Nik-Bakht.

In Figure 29, the Architects first added a column and some doors (the loop shows repetition of the "adding a door" activity) and, in parallel, they created a building story; since they start the model from scratch it makes sense that if they added their first elements, the building story would be created. Then, windows and openings are detected by our algorithm; it is also reasonable because when one adds a window an opening will be created in the wall automatically. Subsequently, architects added some other elements which all categorized as IFC Members in our log. This limitation is related to IFC schema which any load bearing elements other than beam and columns is considered as an IFC Member. Later, the modelers added space to the model. Then, it will be completed by Zones and building element proxies. In the end, furnishing elements are added.

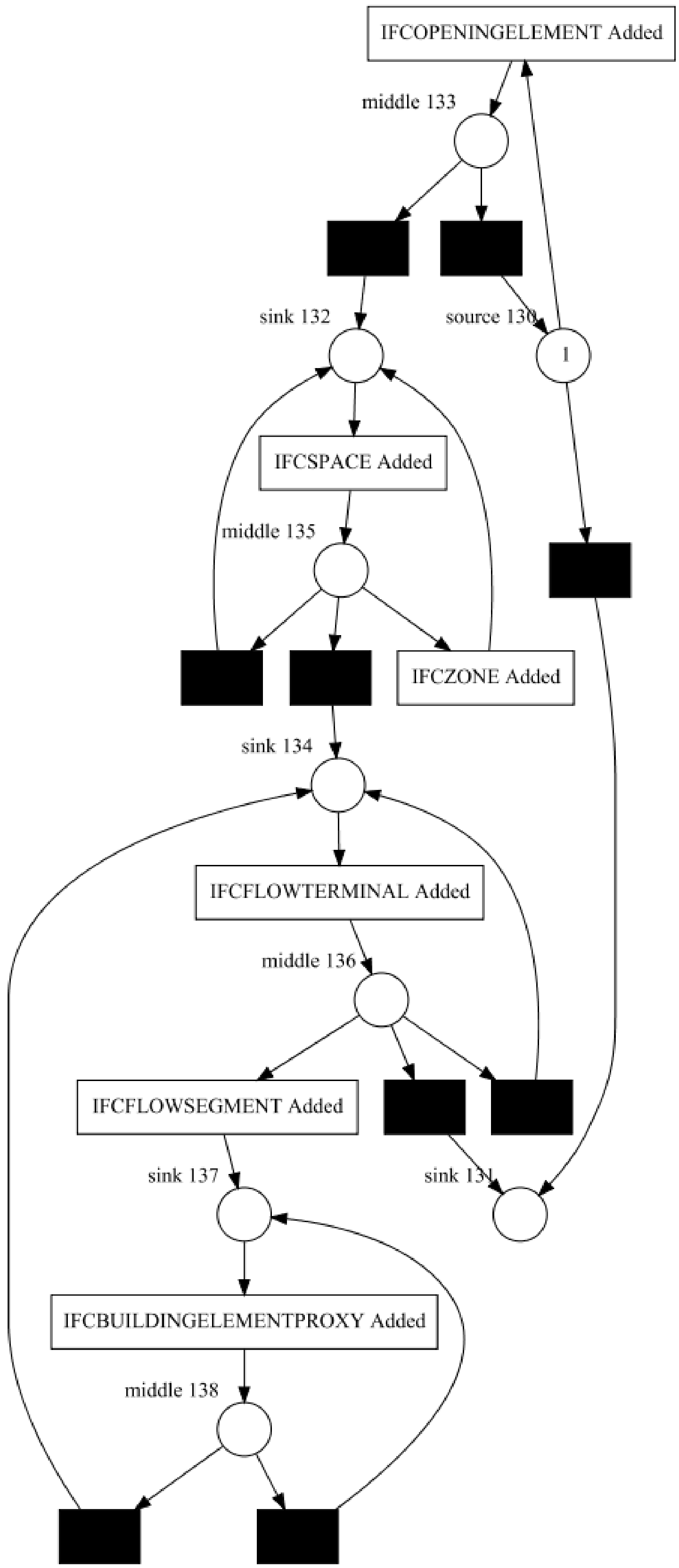



**Figure 27.** MEP "level 3" process

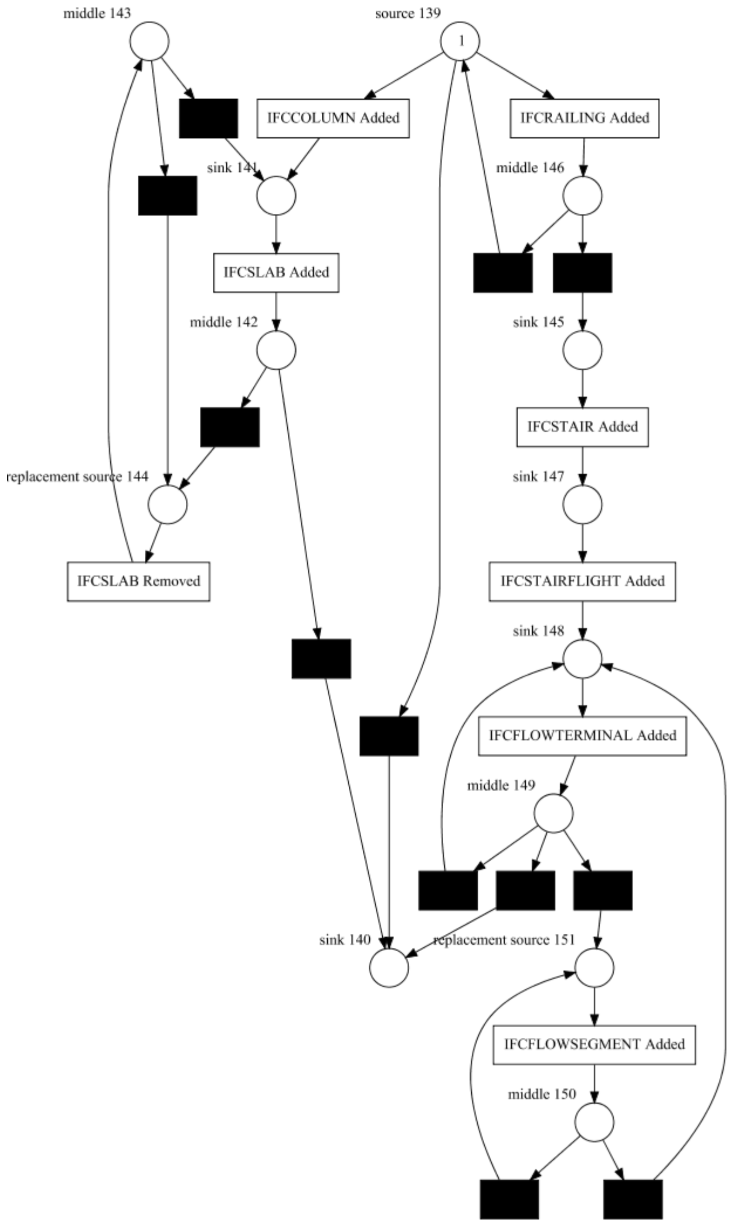



**Figure 28.** Structural Modelling "level 3" process

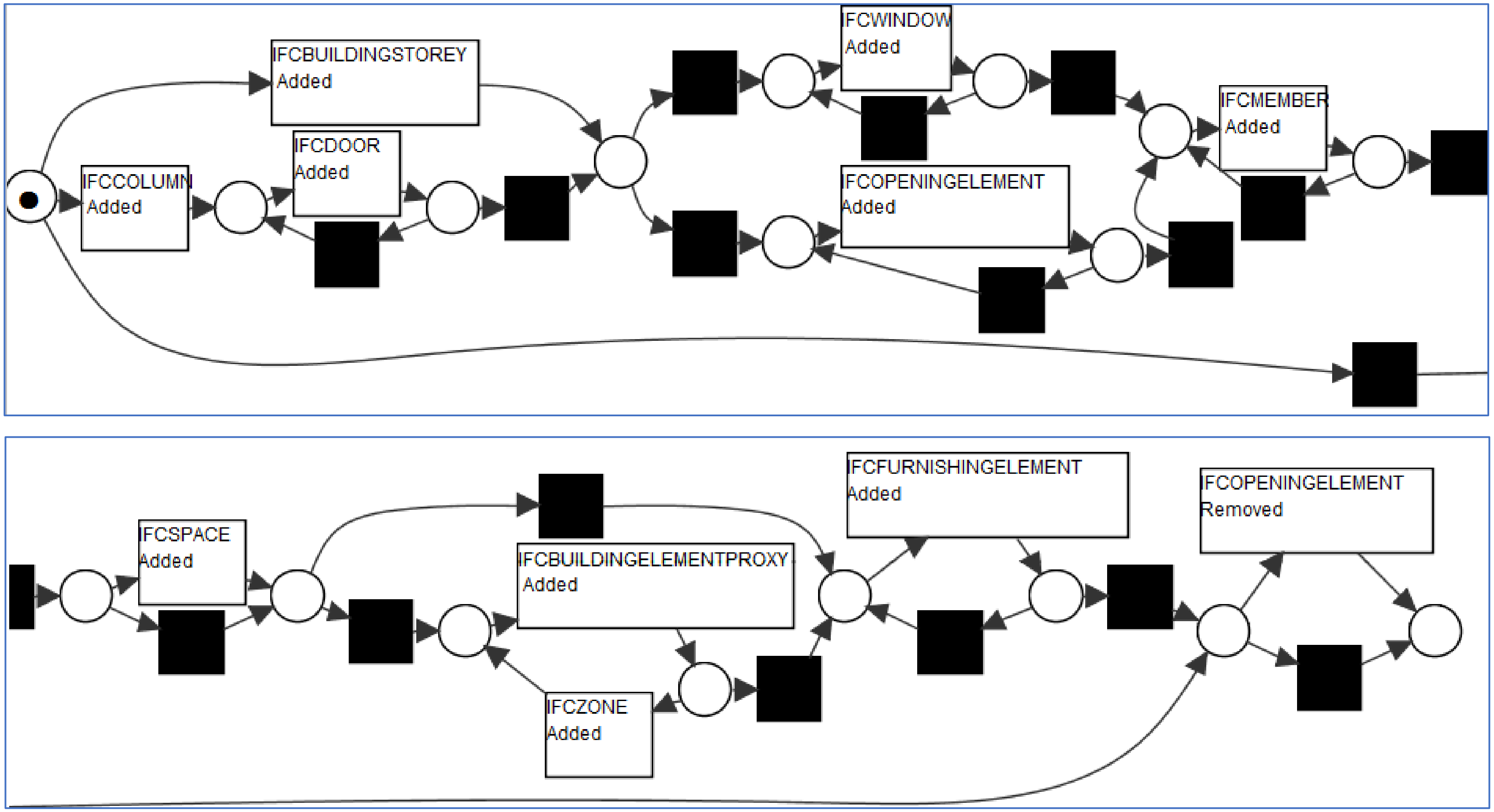



**Figure 29.** Architectural Modelling "level 3" Process (the upper figure is the first part, and the bottom one is the second part)

## 4.3. Case study #2

In order to apply and test the system developed, debugged and improved (based on the findings of the first case study) we studied the process of designing a 3-story hotel building using Revit, involving six resources including four architects, one structural engineer, one MEP engineer. During the design phase, the IFC archiver was running periodically (every 60 seconds), checking if there were changes in the IFC file size. Also, all selected elements with their ID would be captured in a different CSV file, Called “element log”. This element log will help us to reduce the number of searches during the Comparison analysis phase. After completion of the design, every two consecutive IFC files were compared, and the differences were saved into a master CSV file. Then, the generated CSV event log was imported to ProM to produce the XES file using the XESame module. Finally, the process discovery and analytics resulted in finding a social network of the project, the bottlenecks, and process deviations.

### 4.3.1. Process Discovery

Similar to the first case study, the process discovery for the second case study was conducted in two different approaches. The outcome of the first approach can be considered the as-happened “level 3” BIM map for the design authoring process. The “phases as activities approach ” (second approach) is performed to discover the as-happened level 2 BIM map. This map is created to be compared with the as-planned process map (Figure 4).

*Elements’ changes as activities approach: “level 3” as-happened BIM map discovery-* For this case study, the event log contains 7923 different traces (each trace is related to one element) and 15154 activities (88 unique activities which are shown in Table 3). Each activity contains two events: ending time and starting time; in the first case study, the activities just had end time. Also, each event contains five variables including event name, event ID, start time, end time and resource. The underlying process is discovered by inductive miner algorithm and illustrated in Figure 30. The discovered process is too case-specific and using it for planning purposes, would involve the Overfitting issue. Therefore, in order to make the process useful for the BIM Manager, we applied some levels of filtering (considering Paths filter as 0.0 and Activity filter as 1) by omitting some paths that were not repetitive enough while keeping all of the activities. The Petri net model of the filtered process is shown in Figure 31. Figure 32 demonstrates a snapshot of "level 3" process animation (as-happened). The number in each box shows the frequency of changes for the elements which are passing through the box (activity) within the project timeline. For example, the "IFC Building Story added" activity happened 22 times in the project; it is shown in the bottom of the figure.

**Table 3.** Event Categories- Case study #2

| add activities | remove activities | change activities | rotation activities | relocation activities |
|---|---|---|---|---|
| member | slab | top is attached of "wall" | flow fitting | wall |
| member | wall | area of "wall" | flow terminal | slab |
| slab | covering | volume of "wall" | door | flow terminal |
| railing | member | top is attached of "slab" | furnishing element | flow fitting |
| stair | railing | area of "covering" | | member |
| stair flight | stair | volume of "covering" | | beam |
| covering | stair flight | area of "slab" | | column |
| wall | roof | volume of "slab" | | opening element |
| buildingstory | opening element | area of "flow fitting" | | furnishing element |
| roof | buildingstory | area of "flow segment" | | |
| door | door | volume of "flow segment" | | |
| opening element | site | area of "column" | | |
| beam | beam | volume of "column" | | |
| column | column | area of "beam" | | |
| footing | footing | volume of "beam" | | |
| flow segment | flow segment | area of "stair" | | |
| site | flow fitting | volume of "stair" | | |
| flow fitting | flow terminal | area of "member" | | |
| flow terminal | window | volume of "member" | | |
| building element proxy | | area of "stair flight" | | |
| window | | volume of "stair flight" | | |
| furnishing element | | area of "building element proxy" | | |
| | | structural usage of "slab" | | |
| | | structural usage of "beam" | | |
| | | area of "footing" | | |
| | | volume of "footing" | | |
| | | structural usage of "footing" | | |
| | | is external of "slab" | | |
| | | area of "flow terminal" | | |
| | | is external of "wall" | | |
| | | area of "door" | | |
| | | volume of "door" | | |
| | | structural usage of "wall" | | |

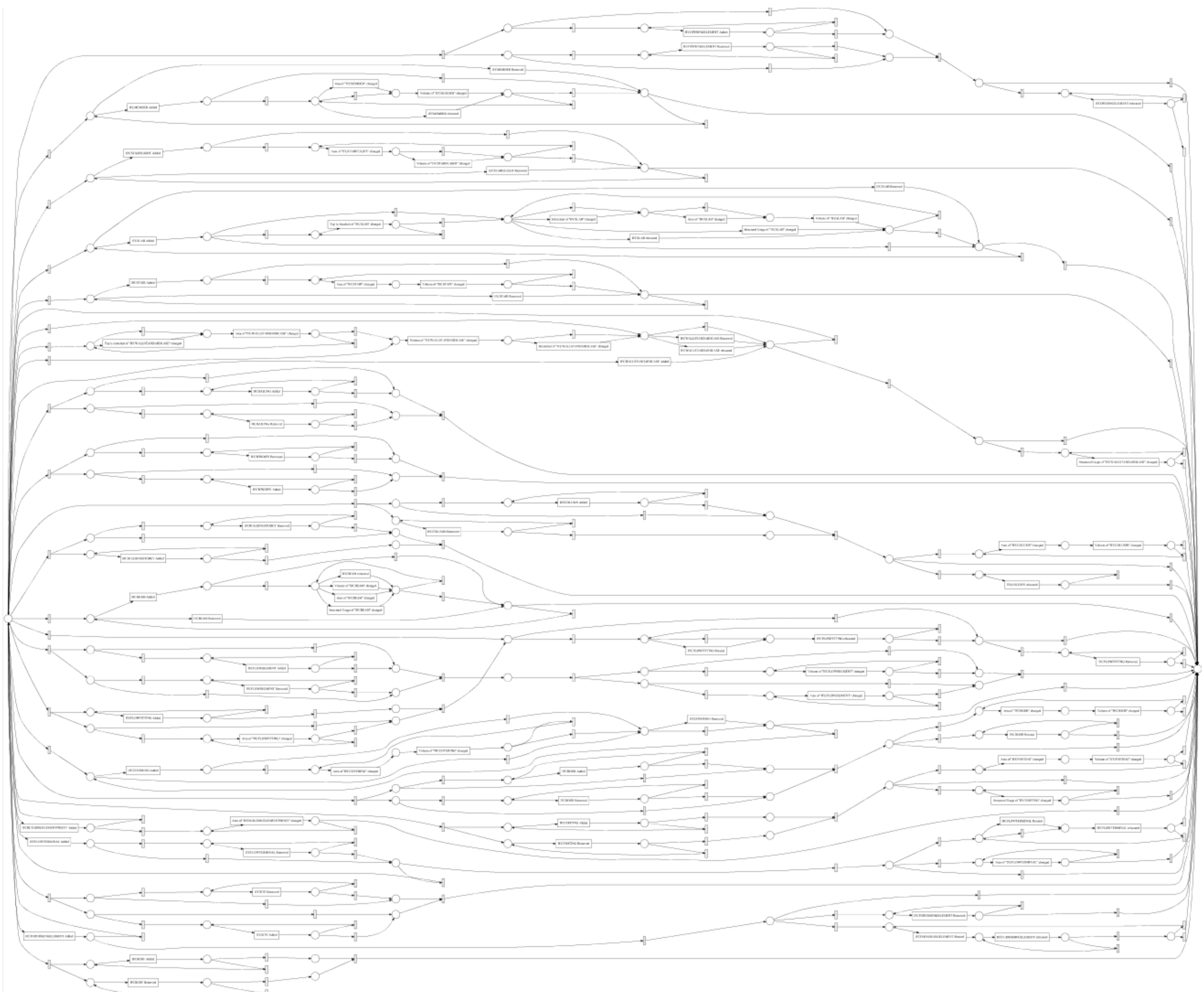


**Figure 30.** "level 3" Petri-net (as-happened) of case study #2 (Over-fit and unreadable!)- Case study #2

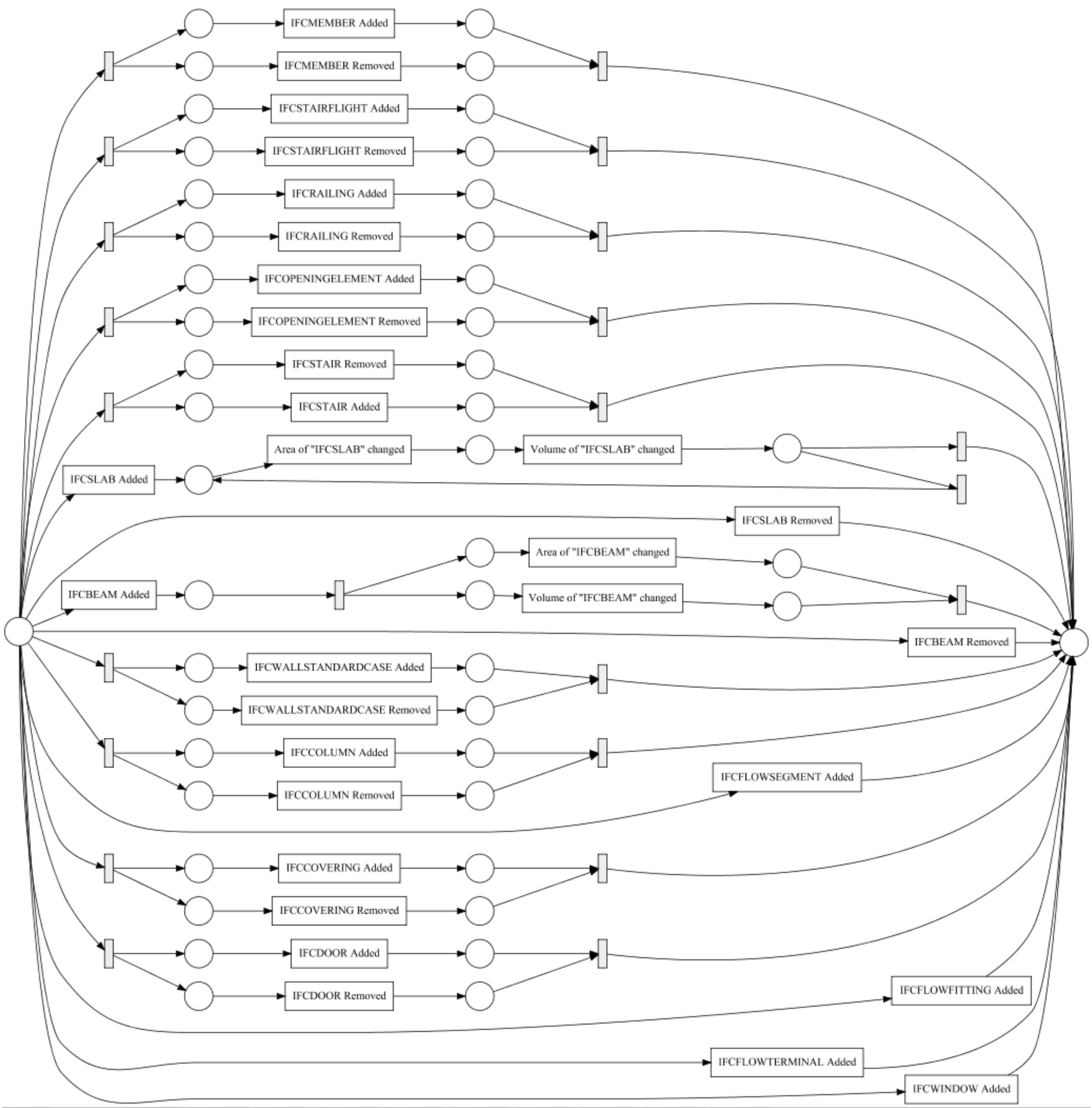



**Figure 31.** The filtered "level 3" Petri-net (as-happened) of the process (generalized)- Case study #2

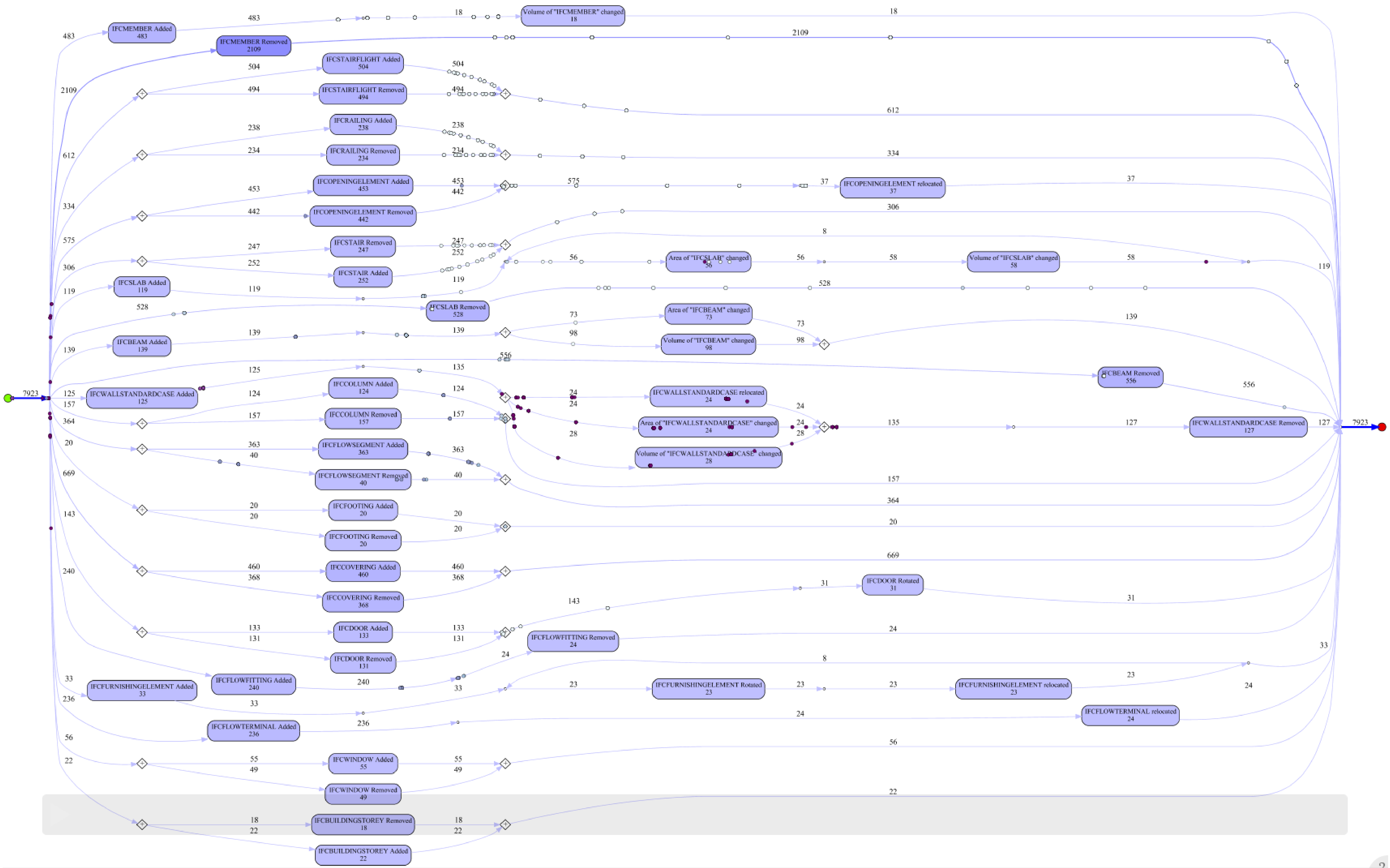



**Figure 32.** "level 3" Process animation (as-happened): the number in each box shows the frequency of changes for the elements which are passing through each box (activity) within the project timeline- Case study #2

*Phases as activities approach: level 2 as-happened BIM map discovery*- Figure 33 visualizes a summary of the event log for the second case study. in the “key data” taskbar on the left, it shows that there are 7923 cases (which corresponds to unique elements) and 30308 events; the events are in two types (start and end moments of an activity such as “IFC Beam added”). “Originators” represents the number of designers. The “Event per case” window categorizes the number of events per case; i.e., it classifies the elements (cases) based on the number of events which has happened for each element (case). For instance, the majority of elements have been changed once (2 events) or twice (4 events) during the second case study. However, there are a few elements in the log which have eleven associated activities (22events).

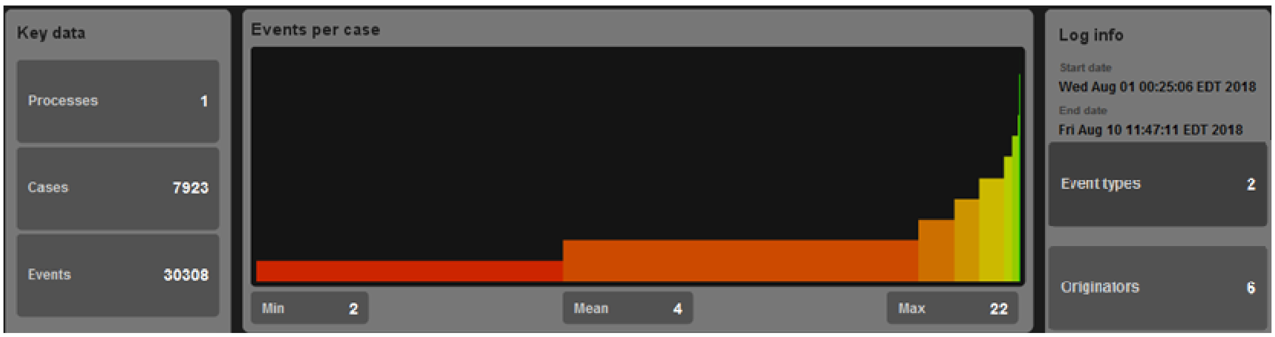



**Figure 33.** Log visualizer- Case study #2

Figure 34 is a graph classifying activities based on different steps of design authoring. For instance, in Stru SD (standing for Structural Schematic Design) phase there are different activities such as “beam added” and “footing added”. Figure 35 shows different design phases based on different weekdays for modeling a three-story building. The design started with Arch SD (standing for Architectural Schematic Design) phase in Wednesday and continued with Stru SD and Mech SD (standing for Mechanical Schematic Design) on Thursday. Then, Arch DD1 and Stru DD1 (standing for the first stage of Structural Design Development) phases were performed on Friday. On Saturday, structural engineering team worked on Stru DD1 phase. Next, Arch DD2 phase was conducted on Sunday followed by Stru DD2 (standing for the second stage of Structural Design Development) and Mech DD2 (standing for Mechanical Design Development) on Monday. Tuesday was the off day for the project. Finally, the modeling was finished by performing the Arch CD (standing for Architectural Construction Documents) phase of the model in the next three days (Wednesday to Friday).

Figure 36 shows different design phases and their timestamps. The whole project was done within ten days. Moreover, it can be inferred that the project was started with Arch SD phase on the first day (day 1) around midnight and finished on the same day in the afternoon. Next, Mech SD phase started the same day (day 1) around the midnight and continued in the next day (day 2) in the afternoon. In day 2, Stru SD phase was started around noon and finished later in the night. In the same day, Mech SD was started early in the morning and finished in the evening shift. Arch DD1 was the next phase of the project which was performed in the next day (day 3) in two shifts of morning and afternoon. The work has been handed over to Struc DD1 phase to be developed by the structural engineering team. It was started on the same night (day3) and finished on the next day (day 4) around 2 pm. The next phase of the project was Arch DD2. It was commenced in that night (day 4) and finished in the next day (day 5) by noon. Afterward, the work was handed over for modeling to Stru DD2 and Mech DD2 phases. Mech DD2 was started on the same night and continued in the next day (day 6) in the morning and afternoon. In parallel structural engineering team was working on Stru DD2 in the afternoon of the same day (day 6). Next day (day 7) was the off day of the project. Finally, the Architectural team finished the project by conducting the Arch CD phase in the last three days (days 8, 9, and 10) of the project.

Figure 4 shows an as-planned BIM process map for design authoring phase. Figure 37 is the as-happened BIM process map of our second case study with the same granularity level (aka level 2 BIM map). It was discovered by the proposed workflow of this research work (BIM-based Process Mining workflow). The process tree for elements in the process is shown in Figure 38. It shows the underlying logic of the decision points of the process (Figure 37). For instance, the right branch of the tree demonstrates the sequential nature of Mech SD and Mech DD2 phases. Additionally, it shows that each of these phases was repeated in a loop. Afterward, the reality (event log) was played on top of this model, and the results were shown in Figure 39. In all of the figures, the dots are in different colors; the darker dots are the elements which stays longer in the process. In Figure 39, the number in each box shows the number of the step execution (each execution of the phase corresponds to a change for an element) in each phase throughout the project.

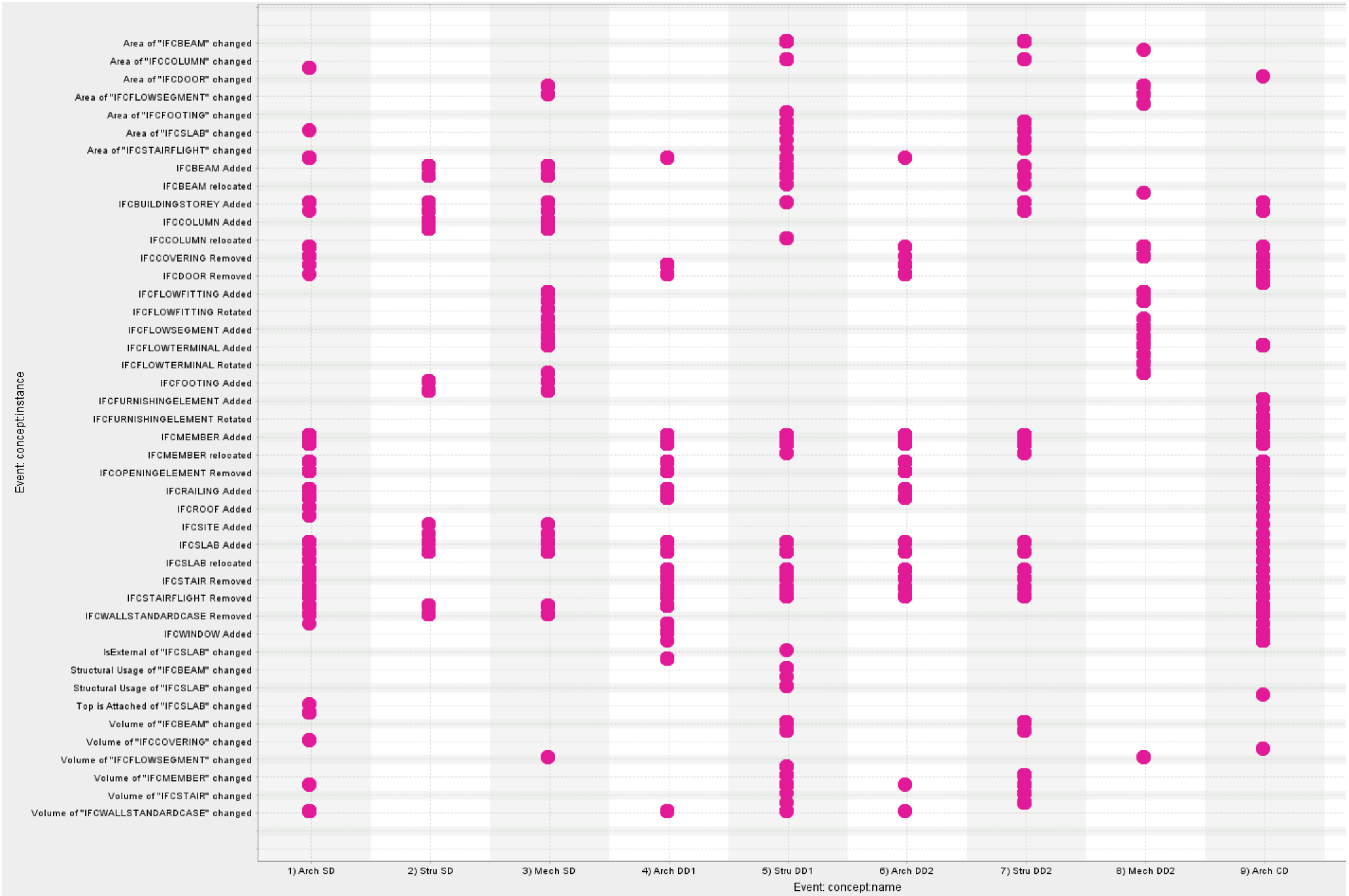



**Figure 34.** Different design authoring phases based on activities they are involved in- Case study #2

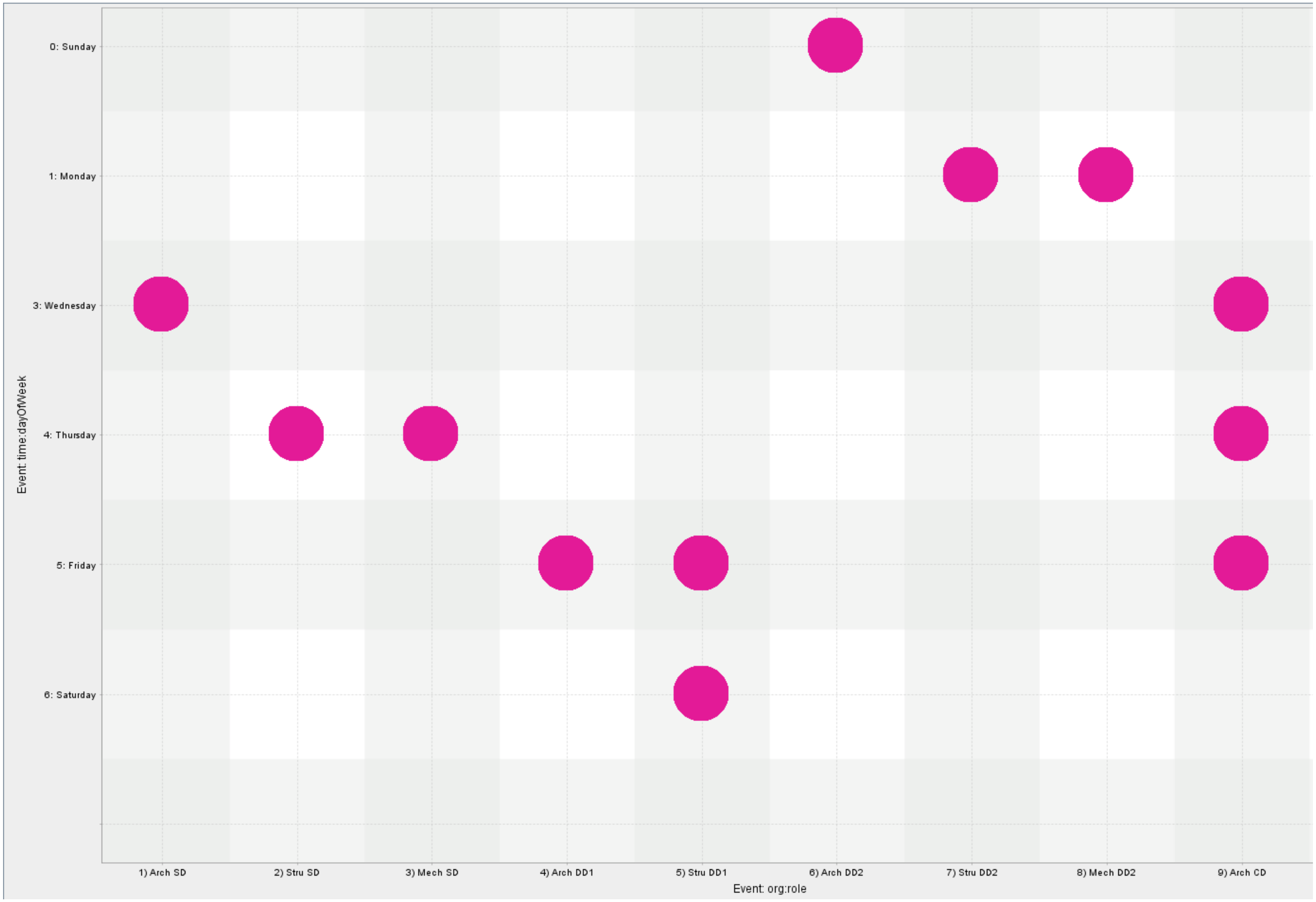



**Figure 35.** Design phases based on days of the week- Case study #2

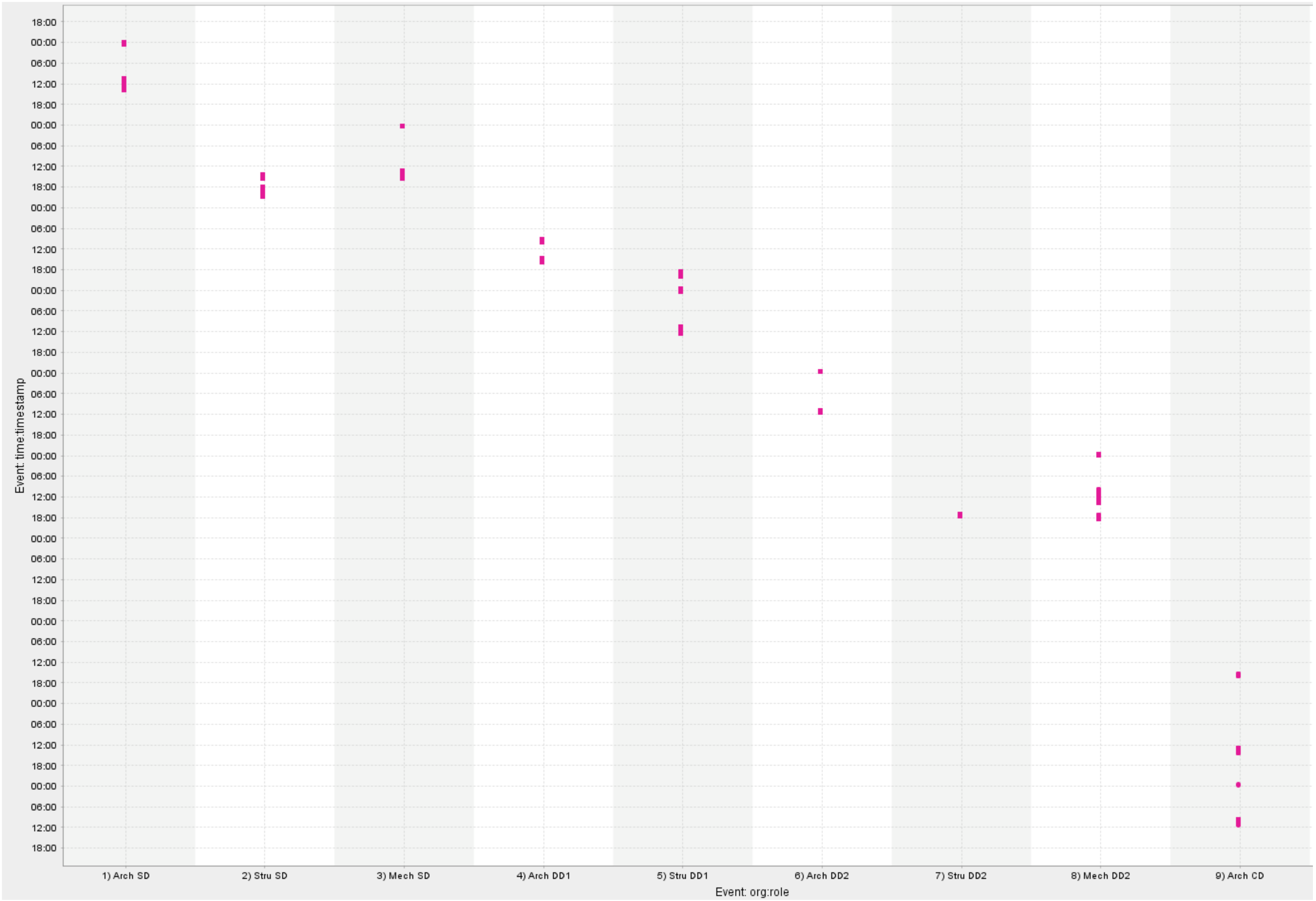



**Figure 36.** Design phases based on timestamp- Case study #2

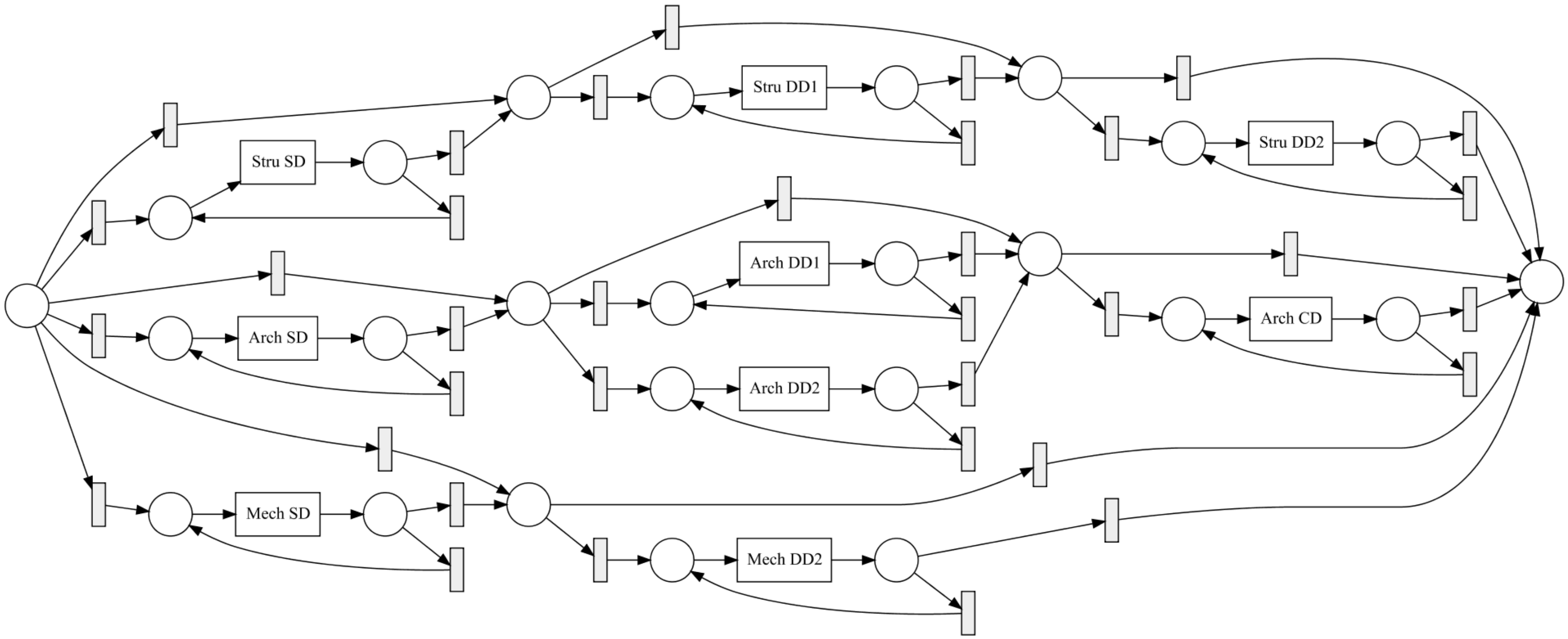



**Figure 37.** Level 2 BIM process map (as-happened) for design authoring which is illustrated in Petri net format - Case study #2

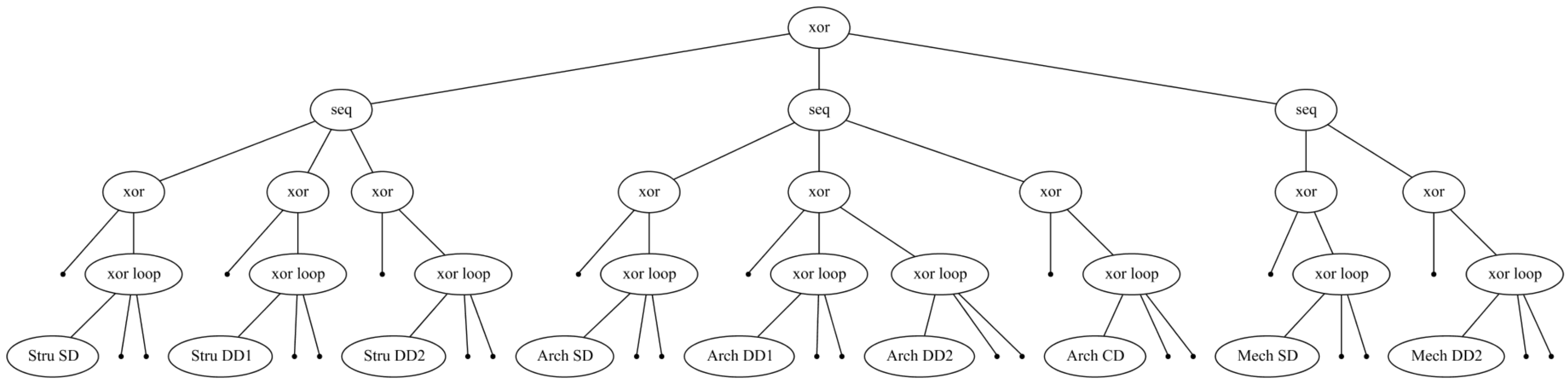



**Figure 38.** Level 2 process tree graph visualization (as-happened)- Case study #2

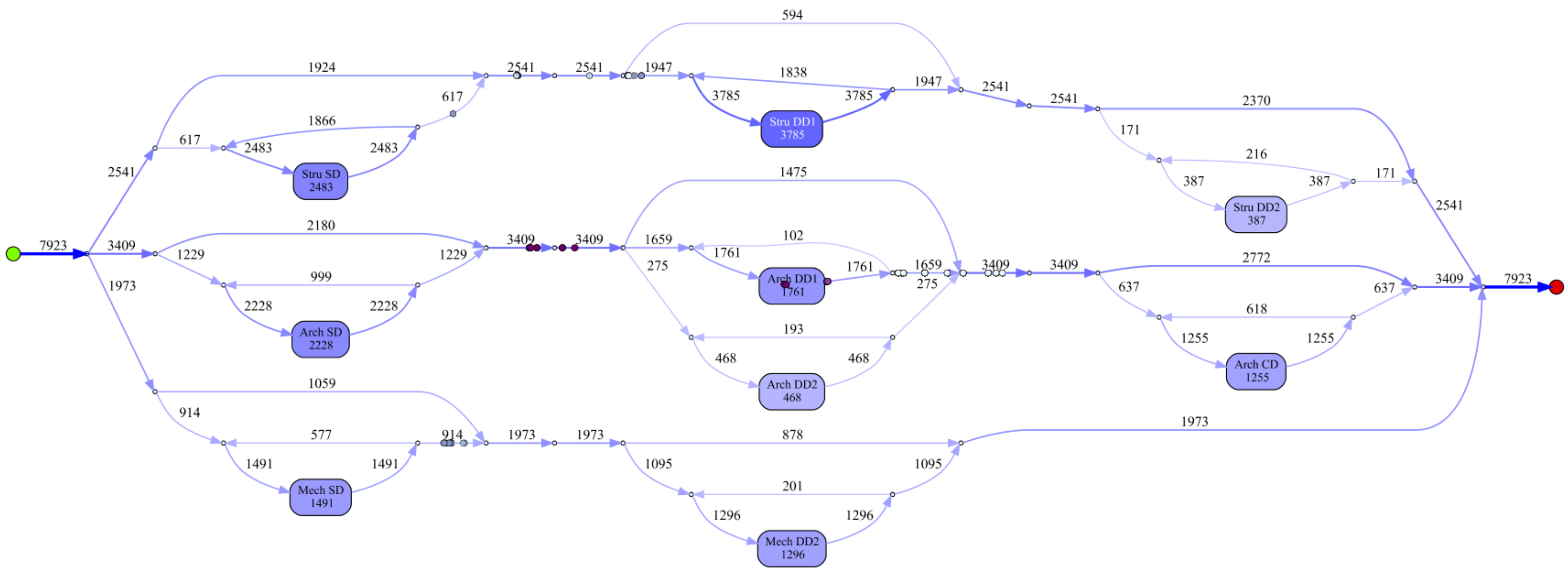


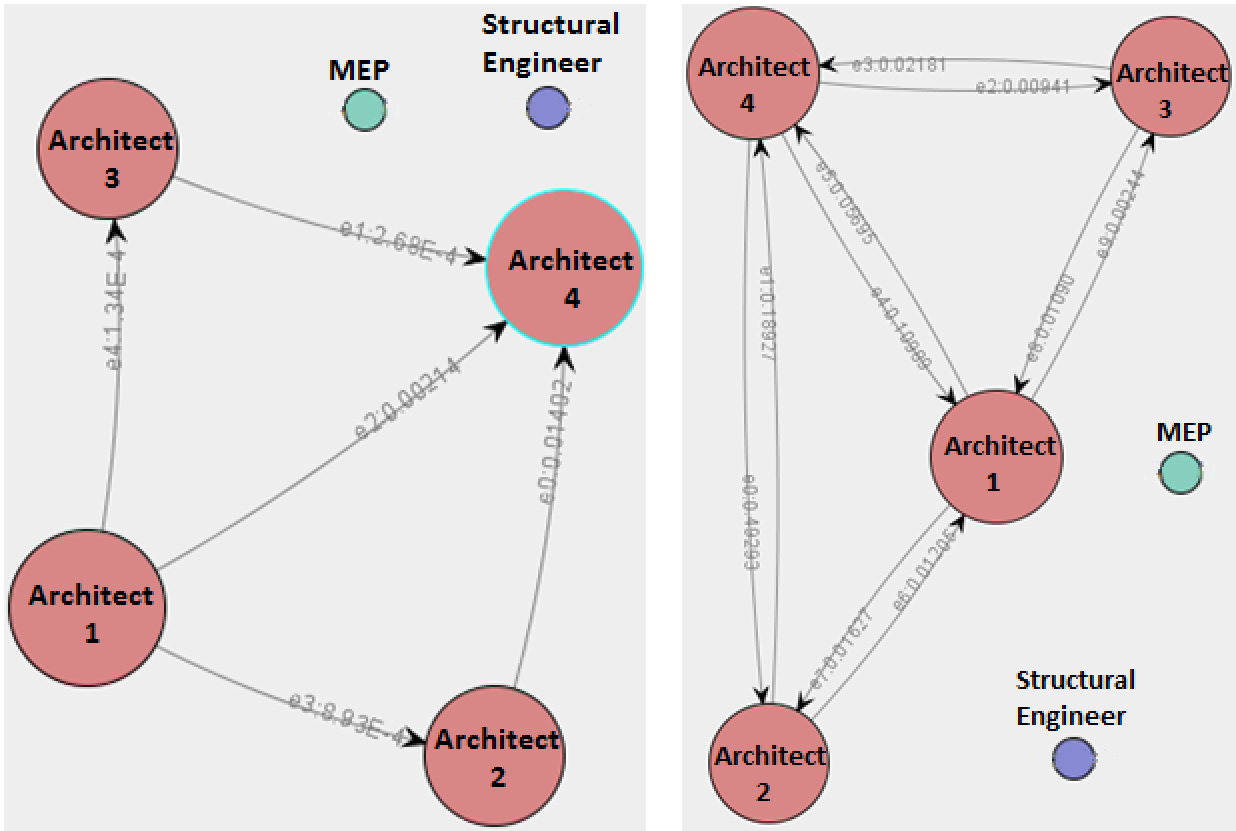

**Figure 39.** Level 2 process animation (as-happened): the number in each box shows the frequency of the step execution (each execution of the phase corresponds to a change for an element) in each phase throughout the project Case study #2

### 4.3.2. The analytics

Figure 40 (a) and (b) show the social networks formed respectively based on “handover of work matrix” and “Working Together”[6] criteria. For instance, in Figure 40 (a), the number on the arrow from Architect 1 to Architect 4 is greater than the number on the arrow from Architect 1 to Architect 3; it means that the handover of the work from Architect 1 to Architect 4 was more comparing to the corresponding amount on the arrow from Architect 1 to Architect 3. Figure 37 can help to clarify this discussion here. As can be seen, the work on an element first is performed in Arch SD phase by Architect 1. Then, the element moved either directly to the Arch CD phase (Architect 4) or arch DD1/DD2 phase (Architect 2/Architect 3); it can be verified from Figure 40 (a). Also, the elements which has been changed by Structural Engineer (the only structural designer of the project) did not change by any other member of the group; which corresponds to solitude of the Structural Engineer node in the Figure 40 (a) and the separate path for structural modeling phases in the process map shown in Figure 37. The same explanation is valid for MEP (the only mechanical modeler in the project).

In the “working together” network shown in Figure 40 (b), the arrows represent the working on the same case (cases correspond to elements in the case study). Based on this method, all architects (Architect 1, Architect 2, Architect 3, and Architect 4) are considered as one group; i.e., there are some elements which architects work on them in the project; e.g., a wall which is created by Architect 1 in Arch SD phase, would be rotated, relocated, resized or one of its property would be changed in the Arch DD or CD phases by Architect 2/Architect 3 or Architect 4. The elements in the structural and mechanical department would not be modified by architects; that is the reason for their solitude in the working together network. Both networks are demonstrating more connection between designers from the same disciplines. The networks are a good representation of the real workflow that we expected to see. Also, most of the work was done in architectural phases which is resulted in bigger nodes for architects in both networks.

[6] “Metrics based on joint cases ignore causal dependencies but simply count how frequently two individuals are performing activities for the same case. If individuals work together on cases, they will have a stronger relation than individuals rarely working together.” [142]




(a) Mining for Handover of work matrix Social Network (b) Mining for Working Together Network

**Figure 40.** Social Network of the case study #2

In this chapter, the BIM-process mining integration model and its algorithms (IFC archiver and IFC logger) were tested, debugged, verified and validated through the implementation of two case studies. Furthermore, two types of as-happened BIM maps for design authoring BIM use of case studies were discovered and explained as outcomes of the proposed workflow (Figure 6). Other types of analytics were investigated over the case studies to see some of the many applications of Process Mining in design authoring including social network analyses, process deviations, and detection of time-consuming activities. In the next chapter, concluding remarks, the research limitation and future works are discussed.

# Chapter 5

# DISCUSSION AND CONCLUSION

The main contribution of this project was the linking workflow between BIM and Process Mining. Process Mining aims discovery and analysis of processes based on event logs; causality detection is out of the scope of process mining. In previous works, there were no direct link to map BIM data structures to the information architecture required in Process Mining. IFC is a platform-neutral BIM model which is not controlled by a single vendor or a group of vendors. Thus, it was selected as the best candidate among BIM data structures to be converted to XES which is the primary input of Process Mining tools. For our case studies, an automatic IFC archiver algorithm was developed using Dynamo to construct the primary IFC database. Then, an IFC comparison algorithm was developed to automatically compare every two consecutive IFC files and log the differences as activities in a CSV file. Subsequently, the CSV event log converted to XES by XESame module. We discovered two types of as-happened processes for design authoring, by feeding the XES event log to Process Mining tools. The discovered design authoring processes were called level 2 and “level 3” process maps. Finally, some analytics have been conducted on the case studies including social network, bottleneck and process deviations analysis.

Also, the case studies were used for verification purposes. A simulated example was used to showcase the potential of Process Mining application using a very known concept in the AEC industry, discrete event simulation of operations. Throughout this exercise, an earth moving operation was modeled using WebCyclone and EZstrobe. The produced raw data after simulation was captured and modified as a CSV format to be used by Process Mining tools. The outcome of the Process Mining tool was an exact process as we simulated at the beginning. A real-life case study is needed to be performed for validation of the proposed methodology and show the power of the suggested analytics. Due to time limitation of this project, validation was excluded from the scope of this project.

Case #1 was done by one person acting in different roles for design authoring process to inspect and debug the codes ('IFC archiver' and 'IFC logger' algorithms), verify, and validate the BIM-Process Mining integration model. Hence, the result of the social network analysis was biased towards more collaboration between the virtual designers. Case #2 was applied by six different colleagues in academia to perform a simulation of the real-world design authoring process, usability test, and customer acceptance test. Due to the different configurations of available computers used by different designers, a better time setting for element logging and IFC archiving was proposed during the two case studies.

## 5.1. Contributions

The goal of this research was "Enabling BIM to act as an ERM system for design authoring processes". To satisfy this goal, the objectives and contributions regarding this project are as follows:

<u>(1) Archiving IFC in the design authoring process regularly (RO 1):</u> Since there were neither an automatic IFC archiver nor a reliable IFC archive available, we created the automatic IFC archiver using Dynamo which acts as a spyware while it is running during modelling; it takes model snapshots (as IFC format) from gradual development of the model. The time duration between two snapshots (IFC files generation) was set based on the IFC file size so that it can reduce the interruption caused by the archiver code during the design authoring phase.

<u>(2) Detecting changes by comparison of consecutive IFC files, extracting their differences, and mapping the discovered differences into the XES format as activities (RO 2):</u> This objective was satisfied by the second dynamo algorithm (IFC logger). The algorithm behind the code received the IFC archives from the previous step as an input. Then, it automatically compared every two consecutive IFC files in chronological order from the beginning to the end. The discovered differences were saved into a master CSV file as activities. Each activity was constituted of one event (end time of the activity in case #1) or two events (start and end time of the activities in case #2). The discovered differences for each case study were saved in a CSV file. The XESame module available in ProM was used to map CSV format event logs to XES event logs.

(3) Mining the as-happened design authoring process (RO 3): Then the as-happened processes of the case studies were mined by the inductive miner algorithm by analyzing XES format event logs. The as-happened models were presented in two different levels (level 2 and "level 3") for each case study. While level 2 BIM maps were showing the actual sequence of designers (case #1) or design phases (case #2), the discovered "level 3" BIM maps were concentrating on the sequence of activities (changing attributes of elements) in both case studies. Also, the event logs were played again on top of the model showing the animation of the elements in the discovered processes throughout the project. The settings of the inductive miner module such as filtering level were set based on the judgment of the BIM manager (in the case studies, I had the role of BIM manager) to decide on the granularity of the process discovery. Moreover, for the first case study the captured process filtered by the designers to generate a detailed process for each discipline. The results and the produced maps were discussed in "Chapter 4 Implementation".

(4) Gaining design related information using Process Mining techniques (RO 4): The social network, deviation and bottleneck analysis were conducted upon the case studies. However, it resulted in interesting insights about the case studies including detection of time-consuming activities, social network and deviations; it can be more insightful when it is done on a real-life project with more involved human resources.

## 5.2. Impacts

Impact of this research effort is summarized at three primary levels: (1) Capturing as-happened BIM maps: until now, two levels of BIM process maps (level 1 and level 2) were introduced by BIM protocols, both of which were referring to "as-planned" processes. In this article, two types of as-happened processes for design authoring (level2 and "level 3" process maps) were discovered. The methodology of the paper and the developed tools which were used for discovery of level 2 and "level 3" design authoring maps can be used for other BIM uses. (2) Reducing reworks in the design phase: reworks in design processes are responsible for 50% of the design time [78]. The reworks can be potentially mitigated by decisions based on real-world information; the information which may be extracted from the big data and process analysis of as-happened processes. (3) Enhancing corporate memory: based on professionals' experiences view in the construction industry, corporate memory of organizations (also referred to as 'organizational memory') is often lost in the transition from one project to the future projects; for instance, once one of the experienced employees leaves an organization, a considerable amount of learned lessons would go with her/him. This is responsible for the loss of information, knowledge and productivity. Exchange of BIM-based event logs can help organizations by integrating their business processes and gaining information about their running projects. This information can be beneficial to assist BIM managers/project managers to break the existing silos and improve collaboration issues among the team members; this problem is responsible for delays in integrated project delivery.

## 5.3. Limitation

The limitations of this project are listed as follows:

1) Need for a more trustable IFC comparison methods: In the literature review, it was mentioned that a signature-based approach for the comparison of IFC models is more accurate compared to the GUID-based comparison method. GUID is also can be considered as a simple signature, although it might not be the best one. In the comparison algorithm, a combination of IFC and Revit GUID was used; this combination is a more accurate signature for elements comparison than just an IFC GUID. This combination can be considered as a signature-based approach and using this idea increased the accuracy of the algorithm. However, there may still be a better and more trustable combination of element attributes as a signature which requires further research and investigation.

(2) Abnormal duration for some activities: Although this problem was investigated and fixed in the section "3.3.2. Duration of activities"; for the case studies, this problem had existed. Table 4 shows the abnormally long durations (the difference between start and end time of the activity) of the activity "IFC footing added" as around five hours, which apparently was not the case. The reason for this error was because of not excluding idleness time from the duration of the discovered activities.

**Table 4.** the error in the duration of activities- The rest time was added to the duration of the first activities of each working shift

| Event name | element id | start time | end time | role | designer |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| IFCFOOTING Added | 2OEjxMhlj5XAiG4iyF 3mSQ | 8-02-2018 16-08-49 | 8-02-2018 21-09-39 | 2) Stru SD | Structural Engineer |
| IFCFOOTING Added | 2OEjxMhlj5XAiG4iyF 3mSK | 8-02-2018 16-08-49 | 8-02-2018 21-09-39 | 2) Stru SD | Structural Engineer |

(3) Insufficient IFC classes for MEP elements: Revit exports and classifies most of the MEP elements in a generic IFC class called "IFCBuildingElementProxy". The problem is that either there is no defined IFC class for different MEP elements or the Revit does not support the MEP element mapping to IFC classes very well.

(4) Detection of more activities: The system must be extended to detect adding or changing more kinds of element properties (such as cost, thermal and acoustic features, etc.) as activities. The current algorithm can capture some property changes including "Area", "Volume", "Family Name", "Type Name", "Phase Created", "Structural Usage", "Is External", "Base is Attached", "Top is Attached", "Load Bearing", "Extend To Structure". While this is comprehensive enough as a starting point; supporting more attributes can give rise to capturing more design activities, and even can help to extend the scope into other BIM Uses.

(5) Interruption caused by IFC archiver algorithm: For large and heavy BIM projects, exporting the IFC files will be as a time-consuming action and can hinder and interrupt designers during the modeling phase. Since it would take longer to export IFC for more complex models, I decided to improve the frequency of IFC archiver algorithm to export IFC files on a time setting based on the file size (Figure 7); the more the file size is, the less frequent IFC generations would happen. This was done on the basis of an assumption that when a model reaches toward its final stages, changes in the model happen more slowly. Although this way I could alleviate the scalability problem for my code, the nature of the problem which is the high IFC generation time remained an issue and needs to be fixed by Revit producer which is Autodesk company, or via another alternative solution.

6) Size-based elimination of extra IFC files: There was an assumption in the development of IFC archiver algorithm to eliminate the produced IFC files if the size difference of the newly produced IFC file and the previous one is less than "0.0004*the newer IFC file size". This assumption might

be inappropriate once we reach the construction documentation phase of a design authoring process. The reason is that most of the activities in that stage are changes of elements' attributes which might not result to a big change in IFC file size.

## 5.4. Future works

By no means the "future works" of this research are limited to the items stated in this section. Five potential continuations of this research are explained in the following paragraphs:

*(1) Revit-based logger:* To address the issue which mentioned as limitation #5, one of the contributions of this work led to two hypothetical algorithms shown in Figure 46 and Figure 47 (Appendix 4). These algorithms will help the researcher as a guide to develop an event log generator extension (add-ons) in Revit. The benefit of these Revit-based logger algorithms would be less interruption for designers and faster activity logging in IFC archiver and IFC logger algorithm, respectively. In other words, designers would encounter less interruption due to the continuous generation of IFC files, and BIM managers may get faster analytics and more real-time information. The Revit based algorithm would act as a real-time process management tool in the design authoring processes of the AEC industry, although the current research has focused on the IFC-based comparison algorithms which make it software independent.

(2) IFC comparison functions in Python: I developed an IFC logger algorithm which compares IFC files consecutively and exports the differences as activities. This algorithm was written in dynamo environment. For further development of this algorithm, it is better to write this algorithm as a set of functions in an object-oriented programming language such as Python and Java. To help the researcher to write these functions, I summarized the more complex regex-based comparison algorithms (detection of "elements' property changes" and "elements' relocation and rotation") which I used over IFC files in Figure 44 and Figure 45 (Appendix 3).

(3) IFC semantic and syntax for event logs: Proposing the required IFC semantics and syntax to store event logs and as-happened process information. It can be done by adding extensions to the IFC MVDs which are designed for 4D BIM models such as design transfer view and reference view.

(4) More Process Mining applications: Other process mining application can be used to derive more information out of the generated event logs using the research methods including

Conformance Checking (event log deviations, model deviations and fitness checking), Organizational Mining (learning organizational entities and corporation information) and Operational Support (detection of deviation, prediction based on the detected error and recommendation based on the best predicted scenario).

(5) Towards more informative BIM maps: The discovered processes can be enhanced with more event data including time, resources and decision behaviors to became data-aware models. For instance, interpretations regarding decision points in the process tree of the as-happened level 2 and level 3 process maps. Also, knowing the properties of the elements (in the level 3 and level 2 maps) that regularly are changed through a certain sequence of activities (process path) in the design authoring process may be insightful for BIM managers.

# REFERENCES


[1] K. Slowey, "The productivity 'train wreck': why construction struggles to compete with rival industries," 19 May 2016. [Online]. Available: http://www.constructiondive.com/news/the-productivity-train-wreck-why-construction-struggles-to-compete-with/419450/.

[2] S. Kouhestani and M. Nik-Bakht, "Towards level 3 BIM process maps with IFC & XES process mining," in *The 12th European Conference on Project and Process Modelling (ECPPM 2018)*, Copenhagen, 2018.

[3] W. van der Aalst, "Business Process Simulation Revisited," *Enterprise and Organizational Modeling and Simulation,* vol. volume 63 of Lecture Notes in Business Information, pp. 1-14, 2010.

[4] P. S. University, "BIM project execution planning guide and templates-Version 2.0," CIC Research Group, Pennsylvania State University, State College, PA, USA, 2010.

[5] R. Quirijnen and S. van Schaijk, "Meten = Weten," Avans Hogeschool, Tilburg, 2013.

[6] J. Yang, M.-W. Park, P. Vela and M. Golparvar-Fard, "Construction performance monitoring via still images, time-lapse photos, and video streams: Now, tomorrow, and the future," *Advanced Engineering Informatics,* vol. 29, no. 2, pp. 211-224, 2015.

[7] W. van der Aalst, Process Mining: Discovery, Conformance and Enhancement of Business Processes, Heidelberg: Springer, 2011.

[8] IEEE, "Process mining case studies," 2013. [Online]. Available: http://www.win.tue.nl/ieeetfpm/doku.php?id=shared:process_mining_case_studies.

[9] A. Manuel , "Process mining - Ana Aeroportos de Portugal," *BPTrends,* 2012.

[10] A. Rozinat, R. S. Mans, M. Song and W. M. P. van der Aalst, "Discovering simulation models," *Journal of Information Systems,* vol. 34, no. 3, pp. 305-327, 2009.

[11] W. van der Aalst, Process mining - Data science in action, second edition, Eindhoven, The Netherlands: Springer, 2016.

[12] In Merriam-Webster's dictionary (11th ed.), Springfield, MA: Merriam-Webster, 2003.

[13] K. Jensen and L. Kristensen, Coloured Petri Nets, Berlin: Springer, 2009.

[14] W. Reisig and G. Rozengberg, Lectures on Petri Nets I: Basic Models, vol. volume 1491 of Lecture Notes in Computer Science, Berlin: Springer, 1998.

[15] M. Weske, Business Process Management: Concepts, Languages, Architectures., Berlin: Springer, 2007.

[16] W. van der Aalst, "Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management," *Desel,* vol. 3098, pp. 1-65, 2004.

[17] W. van der Aalst and C. Stahl, Modeling Business Processes: A Petri Net Oriented Approach, Cambridge, MA: MIT press, 2011.

[18] W. van der Aalst and K. M. Van Hee, Workflow Management: Models, Methods, and, Cambridge, MA: MIT Press, 2004.

[19] W. van der Aalst, "The Application of Petri Nets to Workflow Management," *The Journal of Circuits, Systems and Computers,* vol. 8, no. 1, pp. 21-66, 1998.

[20] W. van der Aalst, "Extracting event data from databases to unleash process mining," in *Proceedings of BPM. Springer* , 2015.

[21] "IDC iView. The Digital Universe Decade—Are You Ready?," International Data Corporation, Framingham, MA, 2010. [Online]. Available: http://www.emc.com/digital_universe.

[22] D. Hand, H. Mannila and P. Smyth, Principles of Data Mining, Cambridge, MA: MIT Press, 2001.

[23] W. van der Aalst, S. Guo and P. Gorissen, "Comparative Process Mining in Education: An Approach Based on Process Cubes," in *12th IFAC International Workshop on Discrete Event Systems*, 2014.

[24] W. van der Aalst and A. Rozinat, "Conformance Checking of Processes Based on Monitoring," *Information Systems,* vol. 33, no. 1, pp. 64-95, 2008.

[25] J. Cook, D. Zhidian, C. Liu and A. Wolf, "Discovering models of behavior for concurrent," *Computer in Industry,* vol. 53, no. 3, pp. 297-319, 2004.

[26] J. Cook and A. Wolf, "Event-based detection of concurrency," *ACM SIGSOFT Software Engineering,* vol. 23, no. 6, pp. 35-45, 1998.

[27] R. Agrawal, D. Gunopulos and F. Leymann, "Mining process models from workflow logs," in *6th International Conference on Extending Database Technology*, Heidelberg, 1998.

[28] S. Hwang and W. Yang, "On the discovery of process models from their instances," *Decisision Support Systems,* vol. 34, no. 1, pp. 41-57, 2002.

[29] G. Shimm, "Process miner - a tool for mining process schemes from event-based data," in *JELIA 2002*, Heidelberg, 2002.

[30] G. Schimm, "Mining most specific workflow models from event-based data," in *BPM 2003*, Heidelberg, 2003.

[31] W. van der Aalst, T. Weijters, J. Herbst, B. van Dongen, L. Maruster and G. Schimm, "Workflow mining: a survey of issues and approaches," *Data Knowledge in Engineering,* vol. 47, no. 2, pp. 237-267, 2003.

[32] W. van der Aalst and B. van Dongen, "Discovering workflow performance models from timed logs," in *EDCIS 2002*, Heidelberg, 2002.

[33] W. van der Aalst, T. Weijters and L. Maruster, "Workflow Mining: Which processes can be rediscovered?," Eindhoven University of Technology, Eindhoven, 2002.

[34] S. Pinter and M. Golani, "Discovering workflow models from activities' lifespans," *Computer in Industry,* vol. 53, no. 3, pp. 283-296, 2004.

[35] M. Golani and S. Pinter, "Generating a process model from a process audit log," in *BPM 2003*, Heidelberg, 2003.

[36] W. van der Aalst and T. Weijters, "Rediscovering workflow models from event-based data using little thumb," *Integrated Computer-Aided Engineering,* vol. 10, no. 2, pp. 151-162, 2003.

[37] G. Greco, A. Guzzo and L. Pontieri, "Mining hierarchies of models: from abstract views to concrete specifications," in *BPM 2005*, Heidelberg, 2005.

[38] G. Greco, A. Guzzo and D. Saccá, "Mining expressive process models by clustering workflow traces," in *PAKDD 2004*, Heidelberg, 2004.

[39] A. de Medeiros, "Genetic Process Mining. Ph.d. thesis," Technische Universiteit Eindhoven, Eindhoven, 2006.

[40] W. van der Aalst, A. de Medeiros and T. Weijters, "Process Mining: Extending the α-Algorithm to Mine Short Loops," Eindhoven Universtiy of Technology, Eindhoven, 2004.

[41] W. van der Aalst, A. Weijters and L. Maruster, "Workflow Mining: Discovering Process Models from Event Logs," *IEEE Transactions on Knowledge and Data Engineering,* vol. 16, no. 9, pp. 1128-1142, 2004.

[42] C. Günther and W. van der Aalst, "Fuzzy Mining: Adaptive Process Simplificaiton Based on Multi-Perspective Metrics," in *International Conference on Business Process Management (BPM 2007)*, Berlin, 2007.

[43] A. Weijters and J. Ribeiro, *Flexible Heuristics Miner (FHM).,* Eindhorven: Eindhoven University of Technology, 2010.

[44] W. van der Aalst, V. Rubin, H. Verbeek, B. van Dongen, E. Kindler and C. Günther, "ProcessMining: A Two-Step Approach to Balance Between Underfitting and Overfitting," *Software and Systems Modeling,* vol. 9, no. 1, pp. 87-111, 2010.

[45] E. Badouel and P. Darondeau, "Theory of Regions," *Lectures on Petri Nets I: Basic Models,* vol. volume 1491 of Lecture Notes in Computer Science, pp. 529-586, 1998.

[46] R. Bergenthum, J. Desel, R. Lorenz and S. Mauser, "Process Mining Based on Regions of," in *International Conference on Business Process Management (BPM 2007)*, Berlin, 2007.

[47] J. Carmona and J. Cortadella, "Process Mining Meets Abstract Interpretation," in *ECML/PKDD 2010*, Berlin, 2010.

[48] J. van der Werf, B. van Dongen, C. Hurkens and A. Serebrenik, "Process Discovery Using Integer Linear Programming," *Fundamenta Informaticae,* vol. 94, pp. 387-412, 2010.

[49] A. de Medeiros, A. Weijters and W. van der Aalst, "Genetic Process Mining: An Experimental Evaluation," *Data Mining and Knowledge Discovery,* vol. 14, no. 2, pp. 245-304, 2007.

[50] S. Leemans, D. Fahland and W. van der Aalst, *Process and Deviation Exploration with Inductive visual Miner,* Eindhoven: Eindhoven University of Technology, 2014.

[51] S. Leemans, "Inductive Visual Miner manual (ProM 6.7)," 2017.

[52] P. Dinaz and D. Ferreira, "Automatic Extraction of Process Control Flow from I/O Operations," *Business Process Management (BPM 2008),* vol. volume 5240 of Lecture Notes in Computer Science, pp. 342-357, 2008.

[53] S. van Schaijk, "Case study: Construction design process mining," Eindhoven University of Technology, Eindhoven, 2016.

[54] S. van Schaijk, BIM based process mining: Enabling knowledge reassurance and fact-based problem discofery within AEC industry, Eindhoven: Eindhoven University of Technology, 2016.

[55] S. van Schaijk, "Case study: Process Mining with facility management data," Eindhoven University of Technology, Eindhoven, 2015.

[56] C. W. Günther and E. Verbeek, "XES Standatd Definition," Technische Universiteit Eindhoven University of Technology, Eindhoven, 2014.

[57] C. Günther, *XES Standard Definition,* 2009.

[58] C. W. Günther and E. Verbeek, "OpenXES Standard Definition," Eindhoven, Technische Universiteit Eindhoven University of Technology, 2014.

[59] C. W. Günther and E. Verbeek, "OpenXES Developer Guide," Eindhoven University of Technology, Eindhoven, Netherlands, 2014.

[60] ProM, 20 Apr 2018. [Online]. Available: http://www.promtools.org/doku.php.

[61] Celonis, 20 Apr 2018. [Online]. Available: https://www.celonis.com/.

[62] Fluxicon, "Disco," 20 Apr 2018. [Online]. Available: https://fluxicon.com/disco/.

[63] myinvenio, "Process Mining Vision," myinvenio.com, [Online]. Available: https://www.my-invenio.com/process-mining-vision/. [Accessed 20 Apr 2018].

[64] Lana, 20 Apr 2018. [Online]. Available: https://lana-labs.com/en/.

[65] J. Buijs, "Mapping Data Sources to XES in a Generic Way, Master Thesis," Eindhoven University of Technology, Eindhoven, 2010.

[66] A. Rozinat and W. van der Aalst, "Decision Mining in ProM," in *International Conference on Business Process Management (BPM 2006)*, Berlin, 2006.

[67] J. Desel, W. Reisig and G. Rozenberg, "Lectures on Concurrency and Petri Nets," *Lectures on Concurrency and Petri Nets,* vol. volume 3098 of Lecture Notes in Computer Science, 2004.

[68] S. Wasserman and K. Faust, Social Network Analysis: Methods and APolications, Cambridge: Cambridge University Press, 1994.

[69] W. van der Aalst, H. Reijers and M. Song, "Discovering Social Networks from Event Logs," *Computer Supported Cooperative Work,* vol. 14, no. 6, pp. 549-593, 2005.

[70] M. Song and W. van der Aalst, "Towards Comprehensive Support for Organizational Mining," *Decision Support Systems,* vol. 46, no. 1, pp. 300-317, 2008.

[71] W. van der Aalst, M. Pesic and M. Song, "Beyond Process Mining: From the Past to Present and Future," in *Proceedings of the 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10)*, Berlin, 2010.

[72] W. van der Aalst, M. Schonenberg and M. Song, "Time Prediction Based on Process Mining," *Information Systems,* vol. 36, no. 2, pp. 450-475, 2011.

[73] J. Goedert and P. Meadati, "Integrating Construction Process Documentation into Building Information Modeling," *Journal of Construction Engineering and Management,* vol. 134, no. 7, pp. 509-516, 2008.

[74] S. Azhar, A. Nadeem, J. Mok and B. Leung, "Building Information Modeling (BIM): A New Paradigm for Visual Interactive Modeling and Simulation for Construction Projects," in *First International Conference on Construction in Developing Countries (ICCIDC–I)*, Karachi, Pakistan, 2008.

[75] Y. Rezgui, C. Hopfe and C. Vorakulpipat, "Generations of knowledge management in the architecture, engineering and construction industry: An evolutionary perspective," *Advanced Engineering Informatics,* vol. 24, pp. 219-228, 2010.

[76] D. Bryde, M. Broquetas and J. Volm, "The project benefits of Building Information Modelling (BIM)," *International Journal of Project Management,* vol. 31, no. 7, pp. 971-980, 2013.

[77] G. Ballard and L. Koskela, "On the Agenda of Design Management Research," in *IGLC*, Guaruja, Brazil, 1998.

[78] G. Ballard, "Positive vs. negative iteration in design," in *the 8th International Group for Lean Construction Annual Conference*, Brigthon, UK, 2000.

[79] M. Al Hattab and F. Hamzeh, "Simulating the dynamics of social agents and information flows in BIM-based design," *Automation in Construction,* vol. 92, pp. 1-22, 2018.

[80] H. Kim and F. Grobler, "Design Coordination in Building Information Modeling (BIM) Using Ontological Consistency Checking," in *ASCE Computing in Civil Engineering*, Austin, Texas, 2009.

[81] C. Baoping, W. Wei and H. Xin, *A Research on Construction Project Based on BIM,* IEEE, 2010.

[82] S. Azhar, "Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AEC industry," *Leadership and Management in Engineering,* vol. 11, no. 3, pp. 241-252, 2011.

[83] T. Hartmann, H. van Meerveld, N. Vossebeld and A. Adriaanse, "Aligning building information model tools and construction management methods," *Automation in Construction,* vol. 22, pp. 605-613, 2012.

[84] S. Azhar and J. Brown, "BIM for Sustainability Analyses," *International Journal of Construction Education and Research,* vol. 5, no. 4, pp. 276-292, 2009.

[85] A. Kiviniemi and R. Codinhoto, "Challenges in the Implementation of BIM for FM—Case Manchester Town Hall Complex.," in *Computing in Civil and Building Engineering*, Orlando, Florida, 2014.

[86] Y. Jiao, Y. Wang, S. Zhang, Y. Li, B. Yang and L. Yuan, "A cloud approach to unified lifecycle data management in architecture, engineering, construction and facilities

management: Integrating BIMs and SNS.," *Advanced Engineering Informatics,* vol. 27, no. 2, pp. 173-188, 2013.

[87] K. Ammari and A. Hammad, "Collaborative BIM-Based Markerless Mixed Reality Framework for Facilities Maintenance.," in *Computing in Civil Engineering*, Orlando, Florida, 2014.

[88] Y. Arayici, "Towards building information modelling for existing structures.," *Structural Survey,* vol. 26, no. 3, pp. 210-222, 2008.

[89] K. Jaakko, "Co-ordination of Plans and Needs for Information Exchange in a Data Modeled," Aalto University, Dipoli, Finland, 2014.

[90] L. Seng, *Singapore BIM Guide Version 2,* Singapore Building and Construction Authority, 2013.

[91] M. Kassem, N. Iqbal and N. Dawood, "A practice oriented BIM framework and workflows," *Computing in Civil Engineering,* pp. 524-532, 2013.

[92] M.-H. Tsai, A. M. Md, S.-C. Kang and S.-H. Hsieb, "Workflow re-engineering of design-build projects using a BIM tool," *Journal of the Chinese Institute of Engineering,* pp. 88-102, 2014.

[93] B. Hola, "Identification and evaluation of processes in a construction enterprise," *Archives of Civil and Mechanical Engineering,* vol. 15, pp. 419-425, 2015.

[94] B. Golzarpoor, C. T. Haas and D. Rayside, "Improving process conformance with industry founcation processes (IFP)," *Advanced Engineering Informatics,* vol. 30, pp. 143-156, 2016.

[95] USACE, "USACE BIM project execution plan, Version 1.0," U.S. ARmy Corps of Engineers, Washington DC, U.S., 2010.

[96] buildingSMART Alliance, "National BIM standard-United States (NBIMS-US), Version 3," National Institute of Building Science, Washington, DC, U.S., 2015.

[97] A. Khanzode, M. Fischer and D. Reed, "Benefits and lessons learned of implementing building virtual design and construction (VDC) technologies for coordination of mechanical, electrical and plumbing (MEP) system on a large healthcare project," *ITCon,* vol. 13, pp. 324-342, 2008.

[98] S. Coates, Y. Aryaici, L. Koskela and K. O'Reilly, "The key performance indicators of the BIM implementation process," in *Computing in civil and building engineering*, Nottingham, 2010.

[99] E. A. Poirier, S. Staub-French and D. Forgues, "Assessing the performance of the building information modeling (BIM) implementation process within a small specialty contracting enterprise," *Canadian Journal of Civil Engineering,* vol. 42, no. 10, pp. 766-778, 2015.

[100] "buildingSmart," [Online]. Available: https://www.buildingsmart.org/. [Accessed 1 November 2018].

[101] A. Gupta, A. Cemesova, C. J. Hopfe, Y. Rezgui and T. Sweet, "A conceptual framework to support solar PV simulation using an open-BIM data exchange standard," *Automation in Construction.*

[102] T. Froese, "INDUSTRY FOUNDATION CLASS MODELING FOR ESTIMATING AND SCHEDULING," in *Durability of Building Materials and Components 8*, Vancouver, 1999.

[103] J. Tah, V. Carr and R. Howes, "Information modelling for case-based construction planning of highway bridge projects," *Advances in Engineering Software,* vol. 30, no. 7, pp. 495-509, 1999.

[104] T. Froese, "Models of Construction Process Information," *Journal of Computing in Civil Engineering,* vol. 10, no. 3, pp. 183-193, 1996.

[105] M. Hassanain, T. Froese and D. Vanier, "Development of a maintenance management model based on IAI standards," *Artificial Intelligence in Engineering,* vol. 15, pp. 177-193, 2001.

[106] T. Liebich, "IFC 2x Edition 3 Model Implementation Guide," buildingSMART International Modeling Support Group, 2009.

[107] T. Liebich, "Unveiling IFC2x4 - The next generation of OPENBIM.," 2010.

[108] C. Groome, "IFC Certification 2.0: Specification of Certification Process," pp. 1-7, 2010.

[109] T. Liebich, "Unveiling IFC2X4 - The Next Generation of OpenBIM," in *CIB W78 2010: 27th International Conference*, Cairo, Egypt, 2010.

[110] J. Won, G. Lee and C. Cho, "No-Schema Algorithm for Extracting a Partial Model from an IFC Instance Model," *Computing in Civil Engineering,* vol. 27, no. 6, pp. 585-592, 2013.

[111] Y. Adachi, "Overview of partial model query language," 22 05 2002. [Online]. Available: http://cic.vtt.fi/projects/ifcsvr/. [Accessed 15 Oct. 2009].

[112] Y.-S. Hwang, "Automatic quantity takeoff from drawing through IFC mopel," *Architectural Institute of Korea,* vol. 20, no. 12, pp. 89-98, 2004.

[113] P.-H. Chen, L. Cui, C. Wan, Q. Yang, S. Ting and R. Tiong, "Implementation of IFC-based web server for collaborative building design between architects and structural engineers," *Automation in Construction,* vol. 14, no. 1, pp. 115-128, 2005.

[114] W. Jeong, J. Kim, M. Clayton, J. Haberl and W. Yan, "Translating Building Information Modeling to Building Energy Modeling Using Model View Definition," *The Scientific World Journal,* vol. 2014, p. Article ID 638276, 2014.

[115] S. Pinheiro, J. O'Donnell, R. Wimmer, V. Bazjanac, S. Muhic, T. Maile, J. Frisch and C. van Treeck, "Model View Definition for Advanced Building Energy Performance Simulation," in *CESBP/BauSIM 2016 Conference*, Dresden, Berlin, 2016.

[116] J. Patacas, N. Dawood and M. Kassem, "Evaluation of IFC and Cobie as data sources for asset register creation," in *14th International Conference on Construction Applications of Virtual Reality*, Sharjah, UAE, 2014.

[117] "IFC Extended Coordination View," buildingSMART Alliance, 2011. [Online]. Available: http://buildingsmart-tech.org/specifications/ifc_specification/ifc-view-definition/coordination-view/summary.

[118] W. Solihin, C. Eastman and Y. Lee, "Toward robust and quantifiable automated IFC quality validation," *Advanced Engineering Informatics,* vol. 29, no. 3, pp. 739-756, 2015.

[119] R. Lipman, M. Palmer and S. Palacios, "Assessment of conformance and interoperability testing methods used for construction industry product models," *Automation in Construction,* vol. 20, no. 4, pp. 418-428, 2011.

[120] "IFC 2x3 Coordination View 2.0 Sub schema," builidingSMART International Ltd., 2013. [Online]. Available: http://www.buildingsmart-tech.org/specifications/ifc-view-definition/coordination-view-v2.0/sub-schema. [Accessed 15 1 2019].

[121] J. Plume and J. Mitchell, "Collaborating design using a shared IFC building model_Learning from experience," *Automation in Construction,* vol. 6, no. 1, pp. 28-36, 2007.

[122] V. Singh, N. Gu and X. Wang, "A theoretical framework of a BIM-based multi-disciplinary collaboration platform," *Automation in Construction,* vol. 20, no. 2, pp. 134-144, 2011.

[123] M. Shafiq and S. Lockley, "Signature-based matching of IFC models," in *35th International Symposium on Automation and Robotics in Construction (ISARC)*, Berlin, Germany, 2018.

[124] E. Hjelseth and N. Nisbet, "Overview of concepts for model checking," in *Proceedings of the CIB W78: 27th International Conference*, Cairo, Egypt, 2010.

[125] A. Kiviniemi, M. Fischer and V. Bazjanac, "Multi-model Environment: Links between Objects in Different Building Models," in *In Proceedings to the 22nd Conference on Information Technology in Construction CIB W78*, Dresden, Germany, 2005.

[126] M. Nour and K. Beucke, "An Open Platform for Processing IFC Model Versions," *Tsinghua Science & Technology,* vol. 13, pp. 126-131, 2008.

[127] E. East, N. Nisbet and J. Wix, "Lightweight Capture of As-Built Construction Information," in *26th International Conference on IT in Construction*, Istanbul, Turkey, 2009.

[128] X. Shi, Y. S. Liu, G. Gao, M. Gu and H. Li, "IFCdiff: A content-based automatic comparison approach for IFC files," *Automation in Construction,* vol. 86, pp. 53-68, 2018.

[129] H. Ma, K. Ha, C. Chung and R. Amor, "Testing semantic interoperability," in *Proceedings of the Joint International Conference on Computing and Decision Making in Civil and Building Engineering*, Montreal, Canada, 2006.

[130] Y. Jeong, C. Eastman, R. Sacks and I. Kaner, "Benchmark tests for BIM data exchanges of precast concrete," *Automation in Construction,* vol. 18, no. 4, pp. 469-484, 2009.

[131] G. Arthaud and J. C. Lombardo, "Automatic Semantic Comparison of STEP Product Models," in *Innovations in Design & Decision Support Systems in Architecture and Urban Planning*, Dordrecht, Netherlands , 2006.

[132] "AIA East Tennessee," [Online]. Available: https://www.aiaetn.org/find-an-architect/design-to-construction/. [Accessed 22 3 2019].

[133] "BIMVision," datacomp, [Online]. Available: https://bimvision.eu/en/free-ifc-model-viewer/. [Accessed 16 11 2018].

[134] H. Pham, Software Reliability, Springer, 2000.

[135] J. Cortadella, M. Kishinevsky, L. Lavagno and A. Yakovlev, "Deriving Petri Nets from Finite," *IEEE Transactions on Computers,* vol. 47, no. 8, pp. 859-882, 1998.

[136] M. Dumas, W. van der Aalst and A. ter Hofstede, Process-Aware Information Systems: Bridging People and Software through Process Technology, New York: Wiley, 2005.

[137] A. Ehrenfeucht and G. Rozenberg, "Partial (Set) 2-Structures—Part 1 and Part 2," *Acta Informatica,* vol. 27, no. 4, pp. 315-368, 1989.

[138] D. Ferreira and D. Gillbald, "Discovering Process Models from Unlabelled Event Logs," in *Business Process Management (BPM 2009)*, Berlin, 2009.

[139] A. Burattin, Process Mining Techniques in Business Environment, Heidelberg: Springer, 2015.

[140] S. Austin, A. Newton, J. Steele and P. Waskett, "Modelling and managing project complexity," *International Journal of Project Management,* vol. 20, pp. 191-198, 2002.

[141] A. Lapinski, M. Horman and D. Riley, "Lean Processes for Sustainable Project Delivery," *Journal of Construction Engineering and Management,* vol. 132, no. 10, pp. 1083-1091, 2006.

[142] W. van der Aalst, H. Reijers and M. Song, "Discovering Social Networks from Event Logs," *Computer Supported Cooperative Work,* vol. 14, no. 6, pp. 549-593, 2005.

# APPENDICES

## Appendix 1

The CSV event log example which is shown in Table 1, is mapped to XES format. The resulted XES event log is provided in this appendix.

<?xml version="1.0" encoding="UTF-8" ?>

<!-- This file has been generated with the OpenXES library. It conforms -->

<!-- to the XML serialization of the XES standard for log storage and -->

<!-- management. -->

<!-- XES standard version: 1.0 -->

<!-- OpenXES library version: 1.0RC7 -->

<!-- OpenXES is available from http://www.openxes.org/ -->

<log xes.version="1.0" xes.features="nested-attributes" openxes.version="1.0RC7">

<extension name="Lifecycle" prefix="lifecycle" uri="http://www.xes-standard.org/lifecycle.xesext"/>

<extension name="Time" prefix="time" uri="http://www.xes-standard.org/time.xesext"/>

<extension name="Concept" prefix="concept" uri="http://www.xes-standard.org/concept.xesext"/>

<classifier name="Event Name" keys="concept:name"/>

<classifier name="(Event Name AND Lifecycle transition)" keys="concept:name lifecycle:transition"/>

<string key="concept:name" value="XES Event Log"/>

<trace>

<string key="concept:name" value="1iG7Z_y9X5Qhuha6Qwoo4C"/>

```
        <event>
            <string key="org:role" value="Stru SD"/>
            <string key="concept:instance" value="0"/>
            <string key="org:resource" value="Structural Engineer"/>
            <string key="lifecycle:transition" value="start"/>
            <date key="time:timestamp" value="2018-08-02T19:12:09.000-04:00"/>
            <string key="concept:name" value="IFCBEAM Added"/>
        </event>
        <event>
            <string key="org:role" value="Stru SD"/>
            <string key="concept:instance" value="0"/>
            <string key="org:resource" value="Structural Engineer"/>
            <string key="lifecycle:transition" value="complete"/>
            <date key="time:timestamp" value="2018-08-02T19:21:36.000-04:00"/>
            <string key="concept:name" value="IFCBEAM Added"/>
        </event>
    </trace>
    <trace>
        <string key="concept:name" value="1zaq3g3QDCDRNzAQEfqMGx"/>
        <event>
            <string key="org:role" value="Stru SD"/>
            <string key="concept:instance" value="1"/>
            <string key="org:resource" value="Structural Engineer"/>
```

```
                <string key="lifecycle:transition" value="start"/>
                <date key="time:timestamp" value="2018-08-02T18:52:04.000-04:00"/>
                <string key="concept:name" value="IFCCOLUMN Removed"/>
            </event>
            <event>
                <string key="org:role" value="Stru SD"/>
                <string key="concept:instance" value="1"/>
                <string key="org:resource" value="Structural Engineer"/>
                <string key="lifecycle:transition" value="complete"/>
                <date key="time:timestamp" value="2018-08-02T18:54:04.000-04:00"/>
                <string key="concept:name" value="IFCCOLUMN Removed"/>
            </event>
        </trace>
        <trace>
            <string key="concept:name" value="2E8ZvhLNP22viTYiWP$6u7"/>
            <event>
                <string key="org:role" value="Arch SD"/>
                <string key="concept:instance" value="2"/>
                <string key="org:resource" value="Architect 1"/>
                <string key="lifecycle:transition" value="start"/>
                <date key="time:timestamp" value="2018-08-01T13:27:24.000-04:00"/>
                <string key="concept:name" value="Volume of &quot;IFCSLAB&quot; changed"/>
```

```
		</event>
		<event>
			<string key="org:role" value="Arch SD"/>
			<string key="concept:instance" value="2"/>
			<string key="org:resource" value="Architect 1"/>
			<string key="lifecycle:transition" value="complete"/>
			<date key="time:timestamp" value="2018-08-01T13:34:50.000-04:00"/>
			<string key="concept:name" value="Volume of &quot;IFCSLAB&quot; changed"/>
		</event>
	</trace>
	<trace>
		<string key="concept:name" value="2YjGjucsTDTBCmmGo9DnsD"/>
		<event>
			<string key="org:role" value="Arch DD2"/>
			<string key="concept:instance" value="3"/>
			<string key="org:resource" value="Architect 3"/>
			<string key="lifecycle:transition" value="start"/>
			<date key="time:timestamp" value="2018-08-05T11:29:31.000-04:00"/>
			<string key="concept:name" value="IFCSTAIR Removed"/>
		</event>
		<event>
			<string key="org:role" value="Arch DD2"/>
```

<string key="concept:instance" value="3"/>

<string key="org:resource" value="Architect 3"/>

<string key="lifecycle:transition" value="complete"/>

<date key="time:timestamp" value="2018-08-05T11:42:48.000-04:00"/>

<string key="concept:name" value="IFCSTAIR Removed"/>

</event>

</trace>

<trace>

<string key="concept:name" value="3I87P1CqT4YAH$ZKpB4A9w"/>

<event>

<string key="org:role" value="Arch CD"/>

<string key="concept:instance" value="4"/>

<string key="org:resource" value="Architect 4"/>

<string key="lifecycle:transition" value="start"/>

<date key="time:timestamp" value="2018-08-07T14:19:54.000-04:00"/>

<string key="concept:name" value="IFCOPENINGELEMENT relocated"/>

</event>

<event>

<string key="org:role" value="Arch CD"/>

<string key="concept:instance" value="4"/>

<string key="org:resource" value="Architect 4"/>

<string key="lifecycle:transition" value="complete"/>

<date key="time:timestamp" value="2018-08-07T14:20:35.000-04:00"/>

<string key="concept:name" value="IFCOPENINGELEMENT relocated"/>

</event>

</trace>

<trace>

<string key="concept:name" value="3I87P1CqT4YAH$ZL$B4Amq"/>

<event>

<string key="org:role" value="Arch DD"/>

<string key="concept:instance" value="5"/>

<string key="org:resource" value="Architect 2"/>

<string key="lifecycle:transition" value="start"/>

<date key="time:timestamp" value="2018-08-04T10:04:29.000-04:00"/>

<string key="concept:name" value="IFCDOOR Added"/>

</event>

<event>

<string key="org:role" value="Arch DD"/>

<string key="concept:instance" value="5"/>

<string key="org:resource" value="Architect 2"/>

<string key="lifecycle:transition" value="complete"/>

<date key="time:timestamp" value="2018-08-04T10:06:18.000-04:00"/>

<string key="concept:name" value="IFCDOOR Added"/>

</event>

```
	</trace>
	<trace>
		<string key="concept:name" value="3KBhxio8LAL94uPyQizUCc"/>
		<event>
			<string key="org:role" value="Mech DD"/>
			<string key="concept:instance" value="6"/>
			<string key="org:resource" value="MEP"/>
			<string key="lifecycle:transition" value="start"/>
			<date key="time:timestamp" value="2018-08-02T15:59:08.000-04:00"/>
			<string key="concept:name" value="IFCFLOWFITTING relocated"/>
		</event>
		<event>
			<string key="org:role" value="Mech DD"/>
			<string key="concept:instance" value="6"/>
			<string key="org:resource" value="MEP"/>
			<string key="lifecycle:transition" value="complete"/>
			<date key="time:timestamp" value="2018-08-02T16:02:10.000-04:00"/>
			<string key="concept:name" value="IFCFLOWFITTING relocated"/>
		</event>
	</trace>
</log>
```

# Appendix 2

In this section, complementary results for case study #2 are provided.

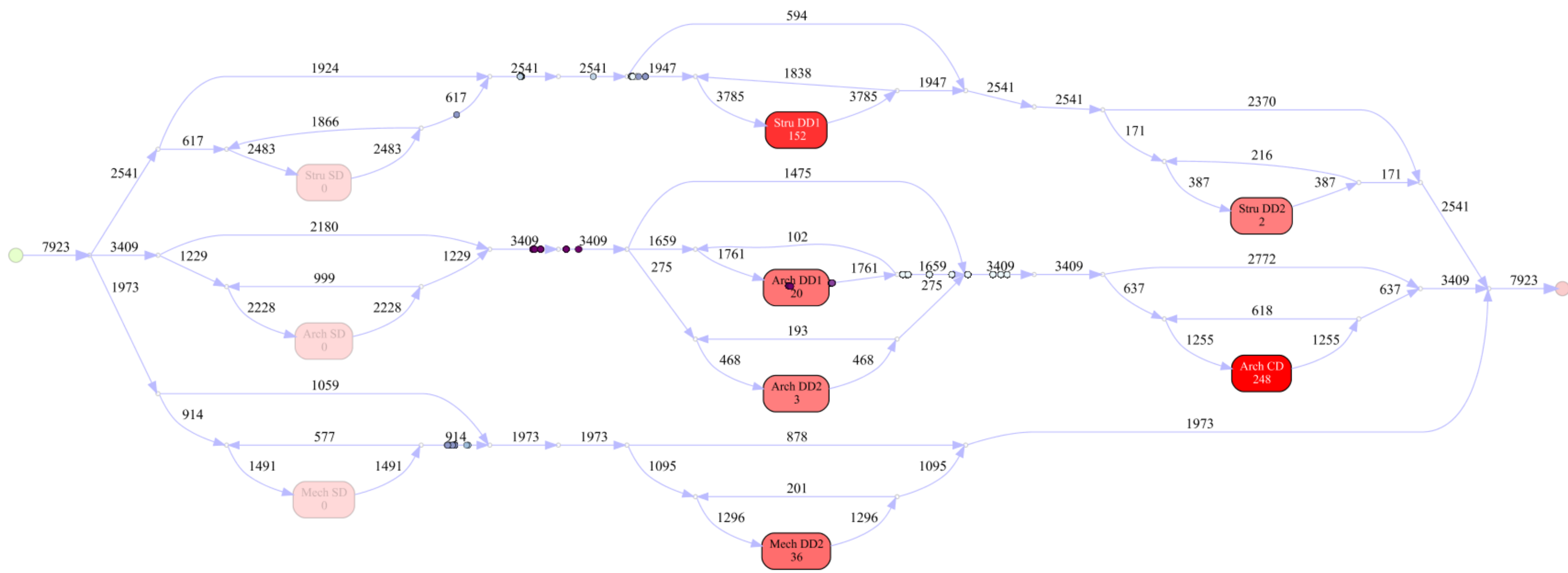



**Figure 41.** Level 2 process animation (as-happened): the variable number in each box shows the number of elements in the queue to be changed in each phase at a particular point of time- Case study #2

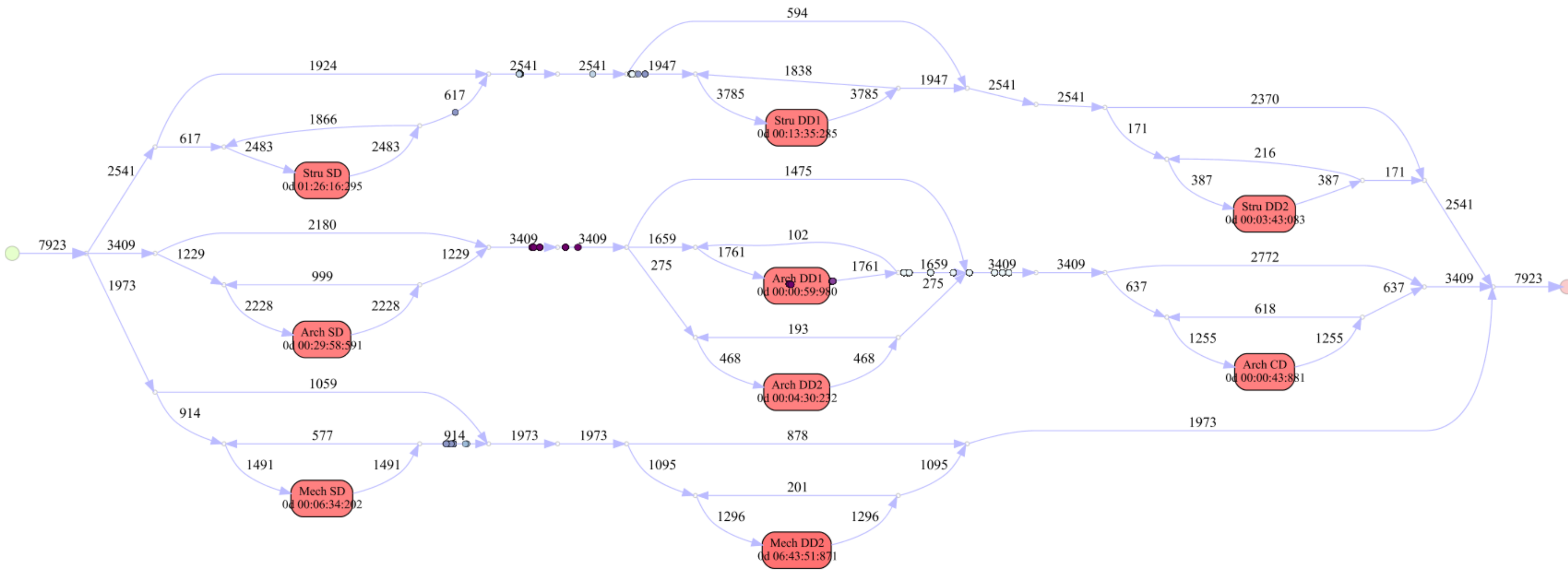



**Figure 42.** level 2 process animation (as-happened): the number in each box shows an average service time of each phase throughout the project-Case study #2

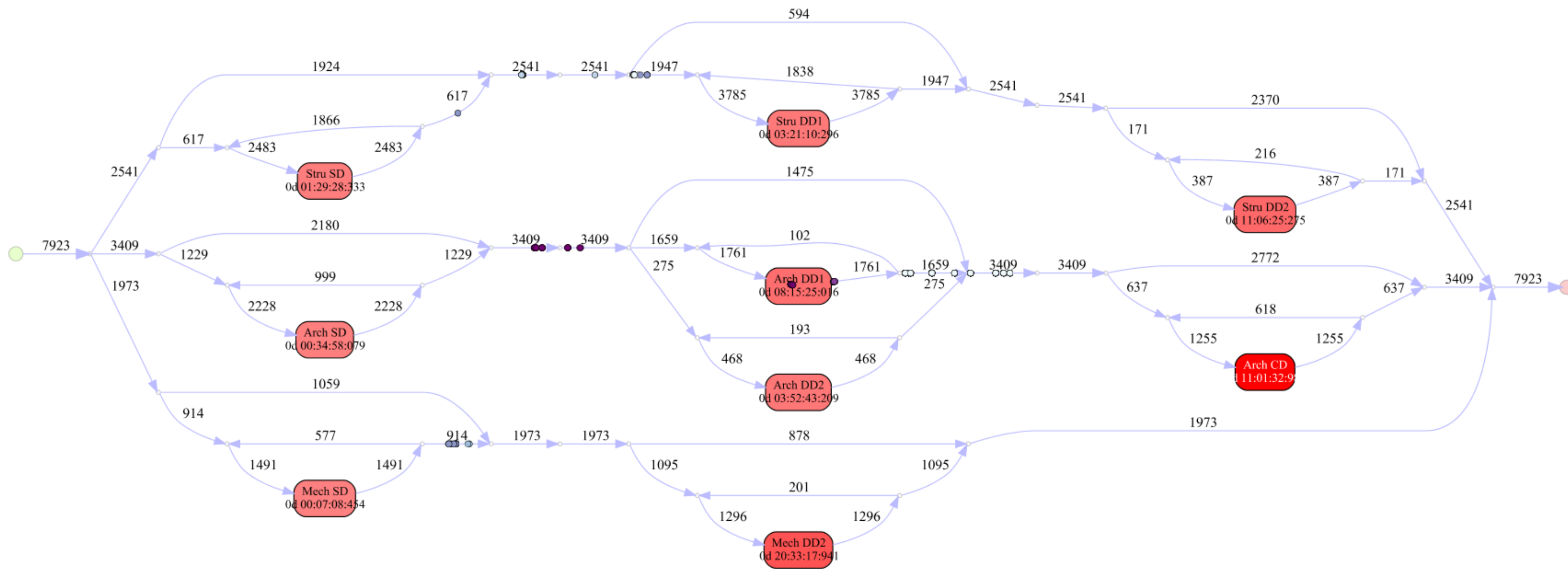



**Figure 43.** Level 2 process animation (as-happened): the number in each box shows an average sojourn time of each phase throughout the project-Case study #2

# Appendix 3

In this section, the IFC comparison algorithms are shown in Figure 44 and Figure 45. The IFC comparison algorithms are mainly based on elements' GUIDs and Regex search methods.

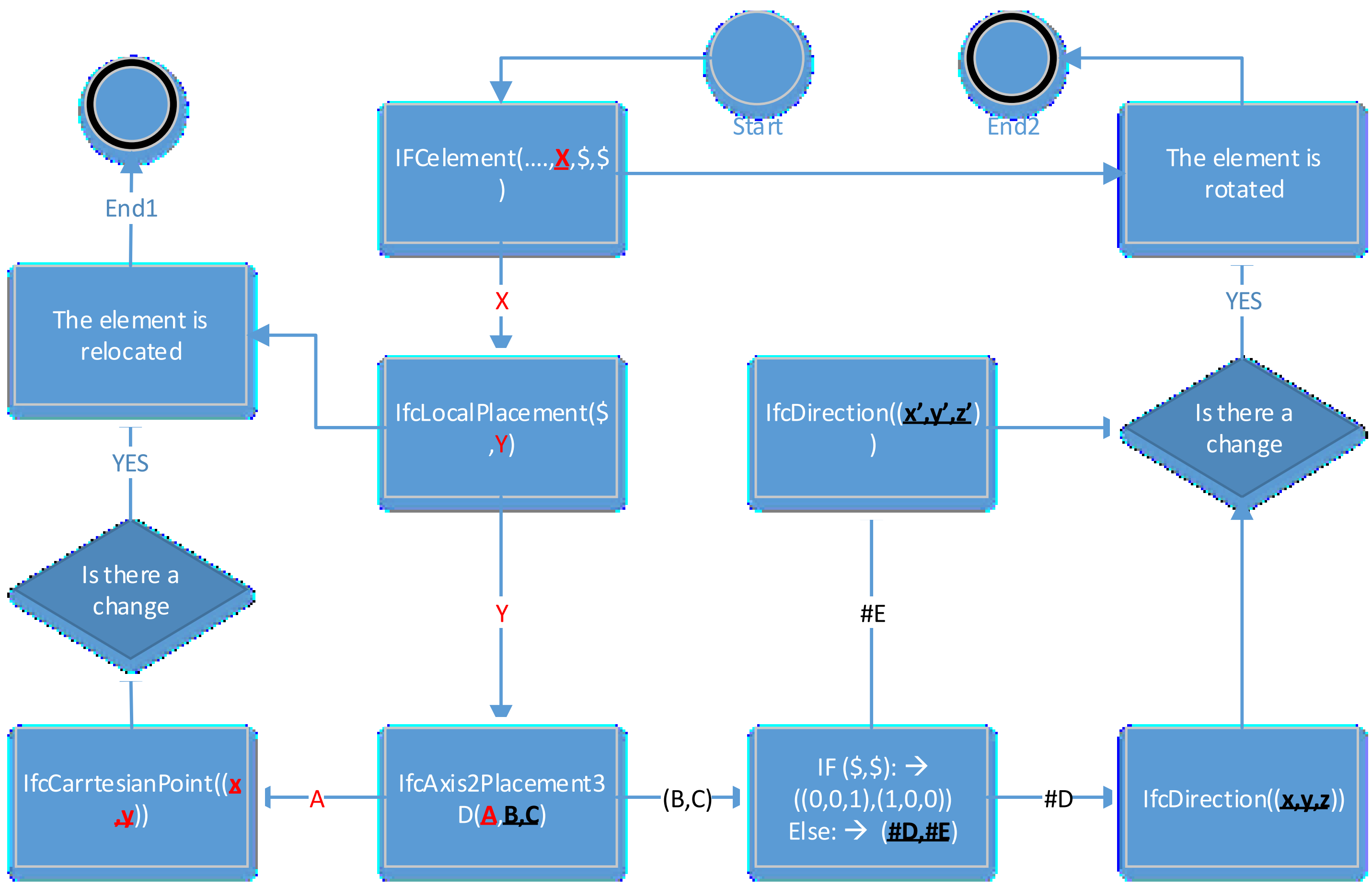



**Figure 44.** Capturing "relocation and rotation of elements" activities algorithm

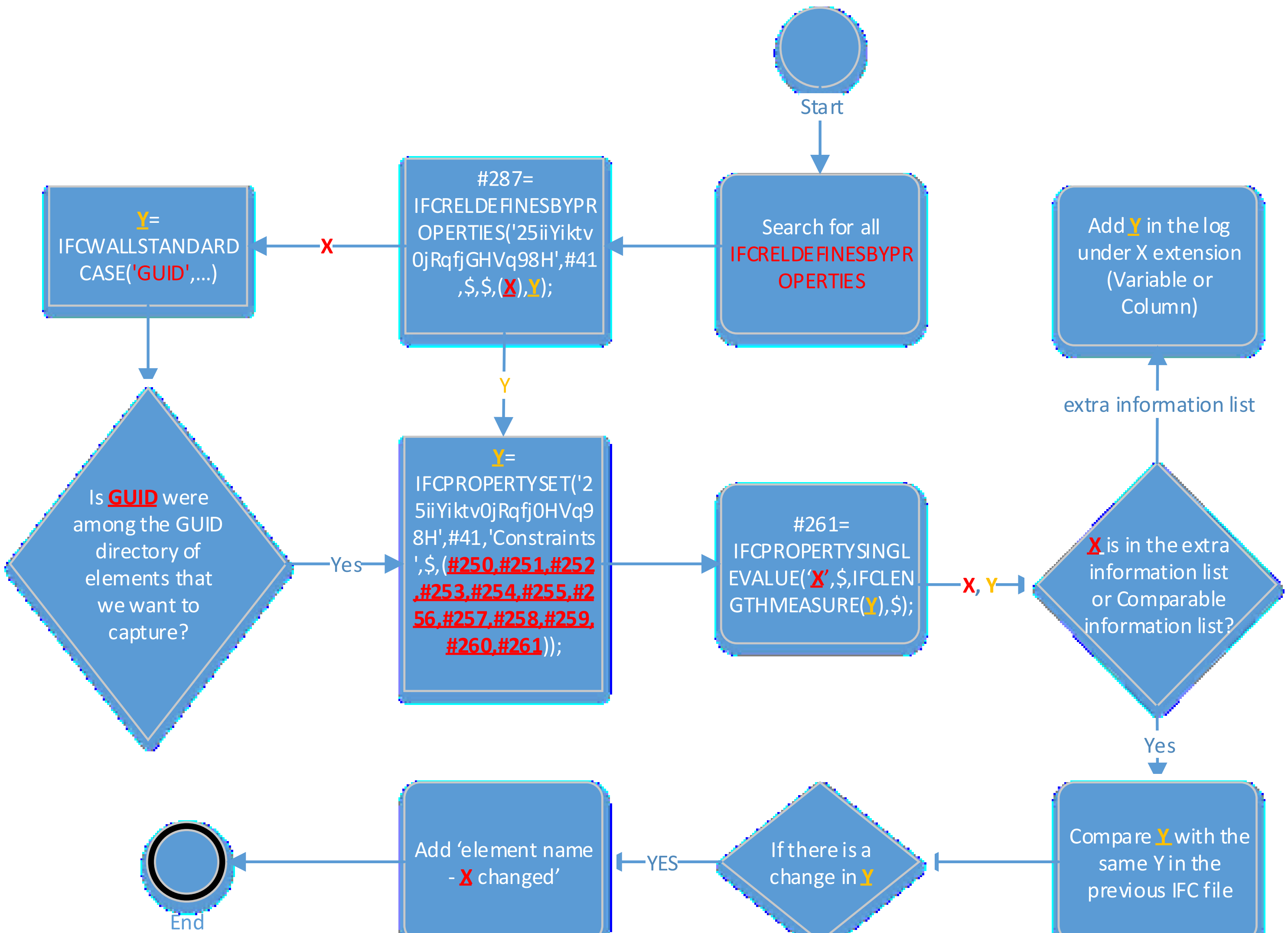



**Figure 45.** capturing "property changes in elements" activities algorithm

# Appendix 4

Revit based logger algorithms are illustrated in the following figures:

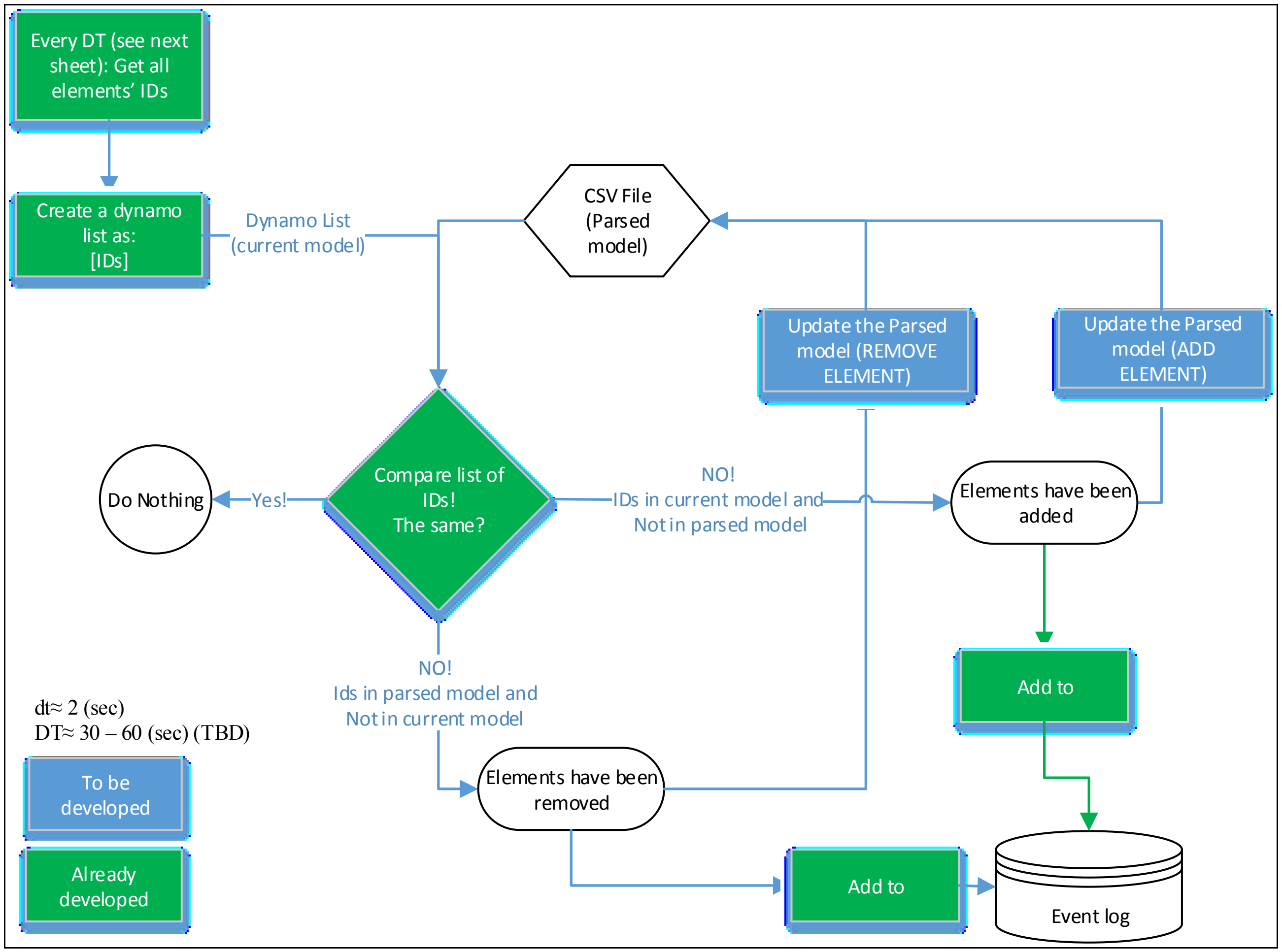



**Figure 46.** Revit based logger algorithm (add-remove)

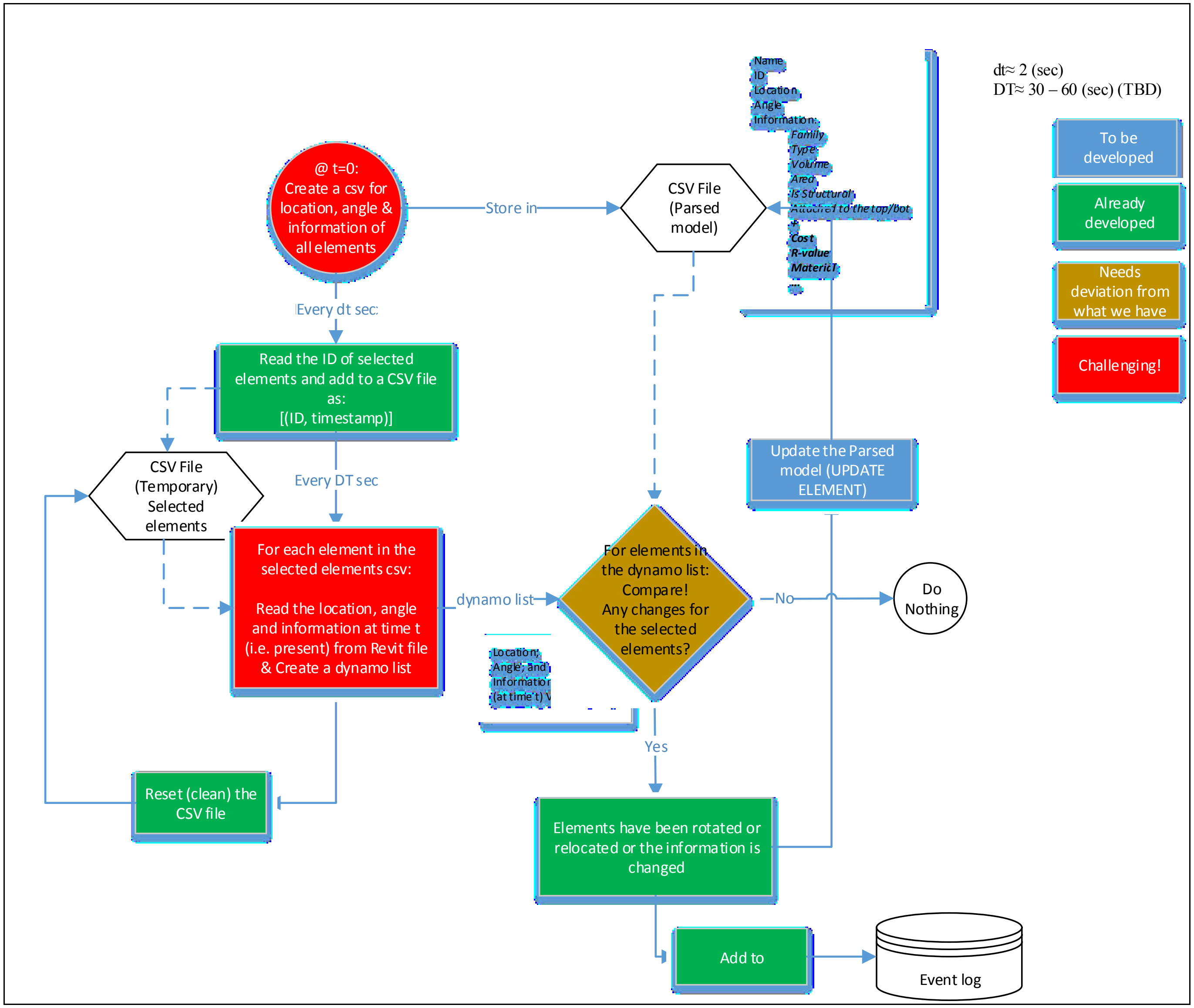



**Figure 47.** Revit based logger algorithm (rotation, relocation and information changes)