From Chaos to Clarity: An Object-Centric Process Mining Case Study in Complaint Requisition Management System

Anukriti Tripathi*, Ayush Raj, Himanshu Singla, Rohini Nandan, Ranjana Vyas and O.P. Vyas

Indian Institute of Information Technology Allahabad, Prayagraj,U.P. India

Abstract

The Complaint Requisition Management System (CRMS) of the Institute is a web-based platform designed to address and resolve issues raised by students, faculty, and staff. It aims to ensure transparency, accountability, and fairness in handling various complaints within the institute. This case study investigates the application of Object-Centric Process Mining (OCPM) to analyze a requisition management system. Traditional process mining techniques often focus on activity sequences, potentially overlooking the relationships between involved objects. OCPM addresses this limitation by providing a granular view centered around business objects. In this study, we aim to analyze the complaint requisition management process from the perspectives of various objects, including engineers, technicians, items and requisitions themselves.

Keywords

Process Mining, Object Centric Process Mining, Complaint Requisition Management System, Process Discovery

1. Introduction

Process Mining (PM) is a data-driven approach which uses event log to provide many business process solutions like visualization of actual workflow, conformance checking between event log & discovered process model, and enhancement of process flow [1]. Traditional PM is a case-centric approach in which only a single perspective is considered for the analysis at a time. Due to this, case-centric PM performs with limited capacity in many application domains. The advancement in the PM field has led to the development of object-centric process mining (OCPM), which could provide a multi-perspective approach to analyze business processes.

OCPM is a relatively new approach of process mining that tackles the limitations of traditional PM by analyzing processes through the lens of objects and their interactions. It considers multiple objects for the analysis, and for this, it uses the Object-Centric Event log (OCEL) [2] as the input and discover the Object-Centric Process Models (OCPM) to understand processes in a multidimensional view. OCPM can also deal with convergence and divergence issues; convergence occurs when one event impacts multiple cases (e.g., one payment for many orders), overstating its frequency, while divergence occurs when a single case has multiple instances of the same activity (e.g., multiple inspections for one product), misrepresenting the process flow. Both issues hinder accurate process analysis [3]. The OCPM allows us to see processes from multiple perspectives, capturing the dynamic connections between objects and the associated events. This leads to more effective process improvement initiatives, increased efficiency, and better overall operational performance [4]. The technological advancement in OCPM has made it capable of playing a significant role in enhancing and resolving the issues of many business processes, such as resolving the overstock problem[5], enhancing the blocking chain process[6], providing better insights into the healthcare process [7], and many more.

Proceedings of the Best BPM Dissertation Award, Doctoral Consortium, and Demonstrations & Resources Forum co-located with 22nd International Conference on Business Process Management (BPM 2024), Krakow, Poland, September 1st to 6th, 2024.

*Corresponding author.
rsi2023003@iiita.ac.in (A. Tripathi); iib2021012@iiita.ac.in (A. Raj); iib2021021@iiita.ac.in (H. Singla); iit2021213@iiita.ac.in (R. Nandan); ranjana@iiita.ac.in (R. Vyas); opvyas@iiitaa.acc.in (O.P. Vyas)
0009-0005-5469-4583 (A. Tripathi); 0009-0005-0296-3307 (A. Raj); 0009-0004-4721-7554 (H. Singla); 0009-0002-0198-7575 (R. Nandan); 0009-0003-7030-2389 (R. Vyas); 0000-0001-5014-8837 (O.P. Vyas)

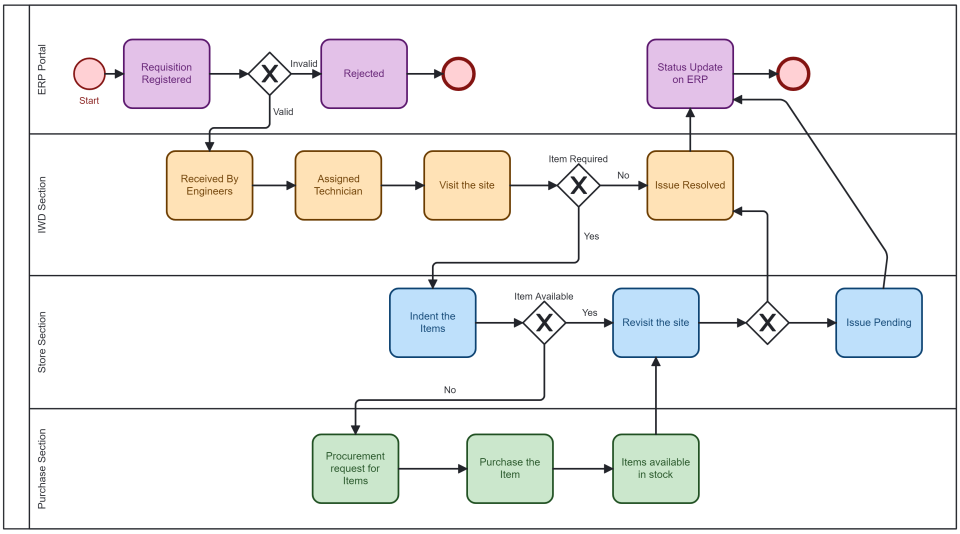

Figure 1: BPMN Process Modelling of CRMS (usingCamunda Software [8])

The complaint requisition management system (CRMS) is integrated with the institute's Enterprise Resource Planning (ERP) system. ERP system, which registers user complaints (covering various service heads such as electrical, plumbing, sanitary repairs, civil work, and AC repair) with unique IDs and forwards them to designated engineers for resolution. Engineers subsequently assign complaints to multiple technicians, who visit the site to address the issue. In case the item requires replacement, technicians indent the item, revisit the site, and resolve the problem. The corresponding requisition status is also visible to the users. The entire process of the ERP portal is illustrated in Figure 1, outlining departmental involvement in the CRMS as swim lane

The proposed work uses the CRMS data of an Indian academic institute (name anonymized). The discussion was made with the concerned authorities of the institute and it was found that the institute is having various complex and unresolved issues. Despite having CRMS, they face uneven load distribution, improper utilization of indent items, delays due to high dependency of one department on another, etc. However, the institute management was also unaware of the exact issues in their system and was trying to get it in their way. This was supposed to be a problem for us when investigating the issue with the PM technique. Hence, we have requested them for their RMS data with some non-disclosure agreement conditions.

Although traditional PM on the CRMS could reveal the actual workflow and bottlenecks of the system but our objective is to analyze the multi-perspective view of the workflow. Specifically,it aims to understand the engineer's perspective in the complete workflow, how technicians receive and distribute daily jobs, and the demand-versus-utilization ratio of indent items. To achieve this, we get motivated to apply OCPM for multi-perspective visualization and analysis of CRMS data. Hence, the work proposes a case study by leveraging OCPM to gain a granular understanding of the CRMS using the institute's real dataset. We aim to analyze the process not just through the traditional activity sequence lens but also by applying OCPM hammer to find the real fabric of CRMS.

2. Relational Data to OCEL Conversion

The proposed work uses OCPM, which uses OCEL for the process discovery. Therefore the conversion of dataset into OCEL is discussed as follows.

2.1. The CRMS Dataset

We received the CRMS dataset in SQL format (relational data) with total 23 number of tables. All these tables were analyzed, and some irrelevant tables were filtered out. In this line, we found the 8 tables (dataset) directly involved in CRMS which are described as follows:

1. **Services Dataset:** Each record in this dataset details a specific service offering like electrical, plumbing, sanitary repair, AC repairs, etc.
2. **Engineers Dataset:** This dataset provides a comprehensive profile for each engineer with their service head in charge.
3. **Utilizes Dataset:** Stores the information of utilization of various resources or items which are indent during the item demanded.
4. **Technicians Dataset:** Provide the list of technicians with their service heads.
5. **Tech Jobs Dataset:** Details the technical jobs or tasks undertaken within the ERP system.
6. **Requisitions Dataset:** Central to service demand and management, this dataset includes data on complaint filed within the system. Key attributes include requisition ID, requested service ID, location of the service demand, a description of the problem, urgency, preferred timings, dates of requisition, job take-up, resolution, current status, and remarks. A total of 4601 requisitions (complaints) are registered.
7. **Items Dataset:** Retrieve the inventory log of the ERP system, listing items or resources that may be indented during the resolving of the problem.
8. **Demands Dataset:** It represents the item demanded during the requisition resolving.

2.2. Data Conversion Method

To convert relational data into OCEL, the proposed work defines a function called create_initial_event_log that creates an initial event log based on input data and generation of OCEL using the PM4PY tool [9]. The following steps explain the technical flow of data conversion, which is also represented in figure 2.

Step 1: Data Loading The script initiates by loading data from several CSV files into Pandas DataFrames. These files contain information on services, engineers, utilization of items, technicians, technical jobs, requisitions, items, and demands. Each data frame represents a different aspect of the service request system.

Step 2: Creating Initial Event Log The create_initial_event_log function is defined to create an initial event log by iterating over each row in the requisitions_df DataFrame. For each requisition (service request), the function performs the following tasks:

1. Service and Engineer Details Extraction:
   - Finds the service details from the services_df DataFrame that match the service ID in the current requisition.
   - Finds the engineer details from the engineers_df DataFrame that match the engineer ID associated with the service.
2. Utilized Items Details Extraction:
   - Selects rows from the utilizes_df DataFrame that match the current requisition ID to identify utilized items.
   - Extracts the unique item IDs and then finds the corresponding item names from the $items\_df DataFrame$.
   - Concatenates item names into a single string for inclusion in the event log.
3. Demand Date Extraction:
   - Finds the demand date from the demands_df DataFrame for the current requisition.
4. Event Log Entry Creation:

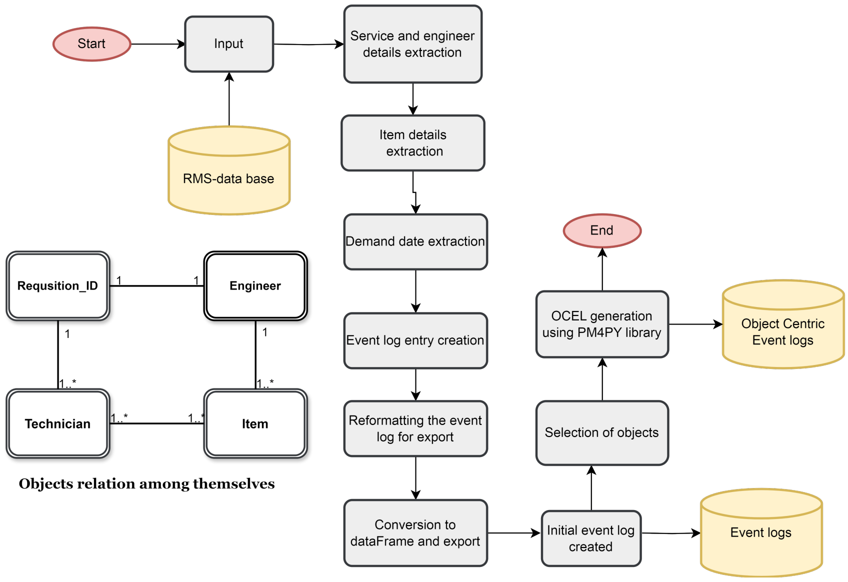

Figure 2: Flow Diagram for Object Centric Event Log Extraction from Relational Data

- Creates an event log entry for the requisition, including the case ID, a list of activities (such as service requests received, engineer assigned, items indented, etc.), and timestamps for these activities.

The function collects all event log entries into a list called $event\_log$

Step 3: Reformatting the Event Log for Export: In this step, the reformat_event_log_for_export function takes the initial event log and reformat it for easier export. It processes each entry in the event log to create a flat structure where each row corresponds to a single activity, including the case ID, activity description, item ID, engineer involved, technician ID, and timestamp. This flat structure is more suitable for exporting to CSV or similar formats because it aligns with the conventional tabular data structure.

Step 4: Conversion to DataFrame: The reformatted event log is converted into a Pandas data frame and is ready for export or further analysis. This DataFrame (reformatted_event_log_df) can easily be saved to a CSV file or used in other data processing tasks.

Step 5: Selection of Objects: We extracted our initial event log in CSV format, which will be helpful in future experiments; we selected four object types named Requisition_id, Engineer, Technician, and item. The relation between objects is given in figure 2 and the objects are described as:

1. **Requisition_id :** The requisition ID is the unique ID the user gets when he/she fills out the requisition form.
2. **Engineer:** There are seven engineers with a different service provider like electrical, plumbing, etc.
3. **Item:** Items are those that may be indented during the CRMS process.
4. **Technician:** Technicians get the work assignment from the concerned engineer and visit the site for problem resolution.

Step 6: OCEL Generation We leverage the PM4PY tool for the final generation of OCEL in .xmlocel format with four types of objects (requisition_id, engineer, technician, and item). The specifications of OCEL are detailed in Table 1.

3. Object Centric Process Mining

The proposed work is to apply object centric process mining in the complaint requisition management system of the institute. The work uses OCEL to develop object centric process model for the in-depth

Table 1
Generated Object-Centric Event Log Summary

Information	Value
Number of events	46260
Number of objects	5970
Number of activities	30
Number of object types	4
Events-objects relationships	142487
Object Types Occurrences (Number of Objects):	
Object Type	Number of Objects
Requisition_id	4601
Item	1305
Technician	57
Engineer	7

insights.

3.1. Discovery Process

The work employed PM4PY tool to discover an object-centric process model in which the object-centric directly follows graph(OCDFG) notation is used, as shown in figure 3. Some brief insights from the generated OCDFG model are as follows:

- A total of 2376 complaints (1999 Academic/Official + 377 Residential/Visitor hostel) were registered in the electrical service head, while 815 for plumbing and 210 for fan workshop complaint requisitions were found.
- Engineer3 took the responsibility of up civil works (whitewashing, painting, mason), welding work, sanitary repair, and plumbing work, making a total of 1051 out of the total requisition 4601 where 1042 are resolved, and eight are pending. Similarly, Engineer4 took the responsibility of electrical, external electrification, and lift service, making a total of 2078 requisitions, where 2045 were resolved and 32 are pending. The Engineer1 is in charge of the Fire Fighting Dept.
- In a total of 4601 cases, there are 8912 technicians assigned, and a total of 11766 items are indented.

3.2. Key Insights of CRMS Process

Analyzing the model obtained from process discovery reveals some significant insights, which are discussed as follows:

1. **Average throughput time:** The average time to complete a requisition is 17 days. The distribution of work or high-demand requisition may require more engineers and technicians. The average throughput time can be reduced by redistributing tasks among engineers and technicians based on current workloads and priorities.
2. **Workload distribution on Engineers:** The electrical (academic/official) service head has the highest number of requisitions (1999 out of 4601), indicating the most workload for engineers in this department, and the process model depicts that carpentry and civil work services are less required in the institute. The engineer3,4,5 are assigned tasks from Multiple work dept. It also gets a large number of requests out of the total. This may cause a delay in further processing, which will take up a lot of time overall. To address the issue, engineers with high workloads, particularly those handling electrical services, should have tasks redistributed to prevent bottlenecks. Consider cross-training engineers from less busy departments like carpentry and civil work to handle basic tasks in high-demand areas, ensuring a more even distribution of work.
3. **Resolved/Pending/underway:** We find the ratio of pending requisition is likely to be similar in every service head. To improve the resolving rate, implement a tracking system that automatically flags pending requisitions for review after a set period.

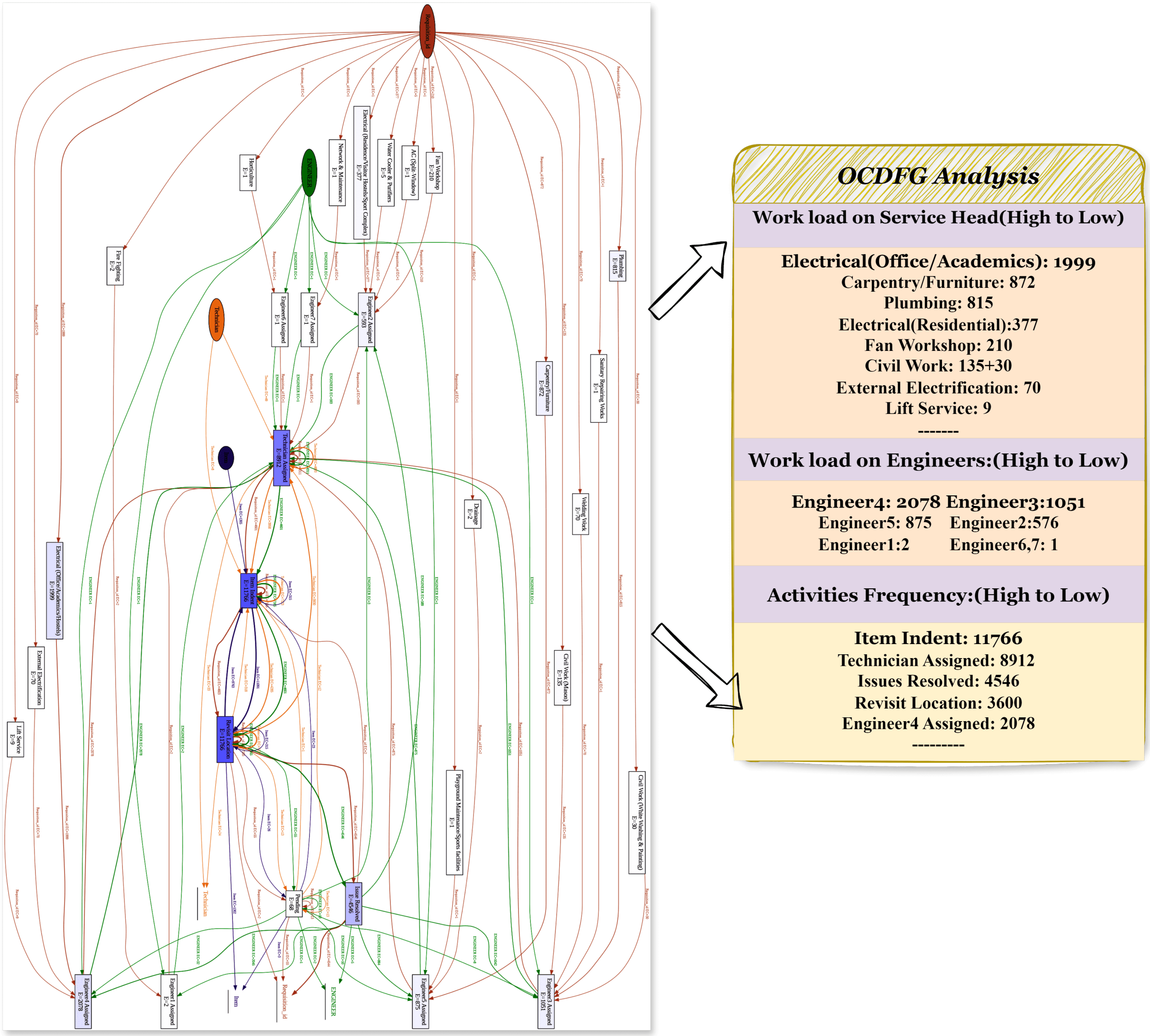

Figure 3: Process Discovery in OCDFG of CRMS process

4. **Workload distribution on Technician:** Multiple technicians are allotted per requisitions(CaseID) for even a fan repair work, ranging from 2 to 7 people. However, only 1 to 2 technicians indented the item and revisited the place. This leads to misconceptions about resource allocation. To address this, streamline the assignment of technicians by ensuring that only the necessary number of people are allocated to each task. A centralized resource management system can be used to match the right number of technicians to each job based on the complexity and requirements, reducing the over-allocation of personnel
5. **Manual Work:** The process seems entirely manual, with many steps that require human intervention. For example, the person requesting maintenance has to visit the place where the maintenance is needed, and then multiple workers visit the same place. A computerized system to track requests and assign workers could make this more efficient.

All of the above outcomes were discussed with the institute's management, and when they checked it, it was found to be correct. All the insights matched the issues that were unclear to them earlier. Therefore, the proposed work shows its effectiveness in dealing with the issue of an institute's CRMS.

4. Conclusion and Future Scope

The work employed object-centric process mining to the complaint requisition management system. The study potentially identified inefficiencies and bottlenecks which is not apparent through traditional PM analysis focusing on activity sequences. Building upon this initial exploration, several avenues exist for further research like comparing the discovered process model with the documented procedures to identify any deviations, Analyzing the process metrics such as cycle time, rework rates, and handoff delays to identify areas for optimization, generating the predictive modeling and integration with improvement initiatives. Our findings reveal specific areas where workflow efficiency can be improved.

These insights present an opportunity for strategic improvements. By optimizing task distribution and enhancing resource management, the organization can achieve more streamlined operations and better meet its operational objectives. Continuing analysis and adjustment based on these findings are essential for driving further efficiency and effectiveness in organizational processes.

References

[1] J. Rudnitckaia, Process mining. data science in action, University of Technology, Faculty of Information Technology (2016) 1–11.

[2] A. F. Ghahfarokhi, G. Park, A. Berti, W. M. van der Aalst, Ocel: a standard for object-centric event logs, in: European Conference on Advances in Databases and Information Systems, Springer, 2021, pp. 169–175.

[3] W. M. van der Aalst, Object-centric process mining: Dealing with divergence and convergence in event data, in: Software Engineering and Formal Methods: 17th International Conference, SEFM 2019, Oslo, Norway, September 18–20, 2019, Proceedings 17, Springer, 2019, pp. 3–25.

[4] W. M. van der Aalst, Object-centric process mining: unraveling the fabric of real processes, Mathematics 11 (2023) 2691.

[5] D. Kretzschmann, G. Park, A. Berti, W. M. van der Aalst, Overstock problems in a purchase-to-pay process: An object-centric process mining case study, in: International Conference on Advanced Information Systems Engineering, Springer, 2024, pp. 347–359.

[6] R. Hobeck, I. Weber, Towards object-centric process mining for blockchain applications, in: International Conference on Business Process Management, Springer, 2023, pp. 51–65.

[7] A. Tripathi, Aneesh, Y. Shivam, S. Pandey, A. Vyas, O. Vyas, Exploring object centric process mining with mimic iv: Unlocking insights in healthcare, in: International Conference on Advanced Information Systems Engineering, Springer, 2024, pp. 360–372.

[8] M. Geiger, S. Harrer, J. Lenhard, G. Wirtz, Bpmn 2.0: The state of support and implementation, Future Generation Computer Systems 80 (2018) 250–262.

[9] A. Berti, S. van Zelst, D. Schuster, Pm4py: A process mining library for python, Software Impacts 17 (2023) 100556.