Overstock Problems in a Purchase-to-Pay Process: An Object-Centric Process Mining Case Study*

Dina Kretzschmann[1], Gyunam Park[1], Alessandro Berti[1], and Wil M.P. van der Aalst[1,2]

[1] Process and Data Science (PADS), RWTH Aachen University, Aachen, Germany
[2] Celonis, Munich, Germany
{dina.kretzschmann,gnpark,a.berti,wvdaalst}@pads.rwth-aachen.de

Abstract. This paper addresses overstock issues in a real-life Purchase-to-Pay process in cooperation with the industry leader in pet retail in Europe. It highlights the development of solutions for more efficient inventory management, thereby reducing overstock. Our approach involves identifying patterns leading to overstock and proposing specific improvement measures within the existing logistics systems. This includes technical modifications in the order suggestion and purchase order processes using Logomate and SAP systems. The research utilizes object-centric process mining techniques as a crucial tool to uncover these patterns, with a focus on the practical solutions derived for overstock reduction. The case study conducted with the PM^2 methodology demonstrates potential benefits in optimizing inventory structure and suggests a path for future research in generalizing these findings across various sectors and automating overstock pattern detection.

Keywords: Object-Centric Process Mining · Case Study · Purchase-to-Pay Process.

1 Introduction

The Purchase-to-Pay process is one of the core business processes across different industries. The overarching objective of this process is to ensure the availability of goods for the customers. This includes ensuring that the right *products* are available at the right *time, quantity, quality, place*, and at the right *costs*. Reaching this objective and controlling the related supply chain is one of the most important success factors for companies [11].

In this study, we examine a pet retail company with a notable market presence, characterized by a sales volume of 4 billion euros in 2023 and over 18,000 workers. The company's ability to provide a diverse array of pet products is underpinned by an extensive supply chain comprising hundreds of suppliers, ten warehouses across five countries, more than 1900 physical stores in Europe, and an online shopping platform. The supply chain mechanism involves procuring pet products from various suppliers, primarily storing them in a central warehouse before distribution

* We thank the Alexander von Humboldt (AvH) Stiftung for supporting our research.

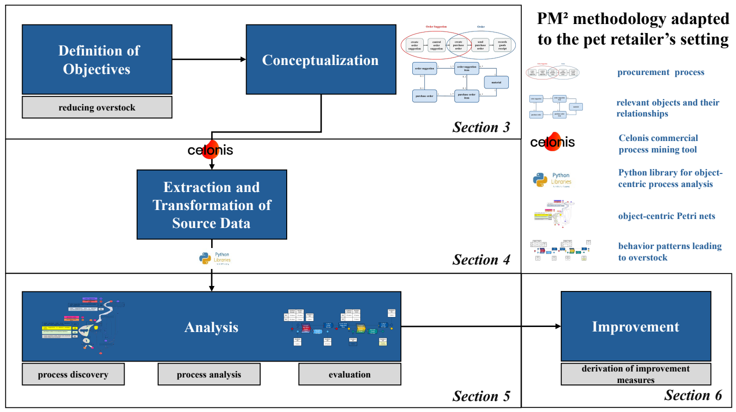

Fig. 1. PM^2 methodology adapted to the pet retailer's setting

to regional warehouses, physical stores, or directly to customers. Managing such a widespread supply chain is highly complex. Talking to the logistics management of the pet retailer, it becomes clear that high stock levels compensate for current challenges along the supply chain. As a result, the company is facing inefficiencies in the inventory structure regarding overstock. There is currently overstock worth several million euros. All Purchase-to-Pay process activities leading to overstock should be reduced accordingly [21].

In this context, the pet retail company opted for the commercial process mining tool Celonis[3], aiming to analyze and enhance its supply chain operations. However, the traditional approach has its limitations when it comes to realistically representing the complexity of the supply chain. Recognized for its robust traditional process mining capabilities, Celonis also offers support for object-centric data models. This study is one of the first documented real-life cases for object-centric process mining and demonstrates that the object-centric feature is particularly beneficial for understanding complex supply chain dynamics, where multiple interrelated objects, such as purchase orders, items, and materials, interact within the same process. For example, a single purchase order in this process might encompass numerous items, these items distributed across various shipments [1]. This approach not only overcomes the constraints of traditional methods but also unveils patterns that otherwise remain hidden. Despite their significant potential, object-centric techniques are still not extensively utilized in practice, highlighting the importance of their application in real-world scenarios such as this study. By expanding traditional process mining to object-centric process mining, the pet retailer aims to gain new comprehensive insights, address bottlenecks, optimize inventory management, and enhance order processing efficiency more effectively.

This case study focuses on addressing the overstock issues within the Purchase-to-Pay process. In particular, we aim to identify and categorize general patterns that lead to overstock using the application of object-centric process mining techniques. The overstock patterns, potentially common across various Purchase-to-

[3] https://www.celonis.com

Pay processes in different sectors, are analyzed with the goal of developing targeted improvement measures. These measures are designed to optimize inventory structure, addressing the core challenges in the supply chain and contributing to more efficient operations.

The case study applied the *PM² methodology* to the pet retailer's setting (see Figure 1) [9]. The paper is structured as follows: Section 2 reviews related work. Section 3 sets the objective of reducing future overstock and outlines the target data format. Section 4 details the extraction of source data with Celonis and its transformation into an object-centric event log. In Section 5, process models are discovered and analyzed using a tool for object-centric process analysis [3], identifying overstock patterns. This section also covers the evaluation of case study results with the pet retailer's logistics management. Section 6 proposes measures to decrease overstock-causing activities. The conclusion and future research directions are presented in Section 7, emphasizing the need to generalize and automate these findings.

2 Related Work

The paper tackles overstock problems by detecting patterns leading to overstock in a real-life Purchase-to-Pay process using the application of object-centric process mining techniques. Therefore, we consider the scientific results of previous work related to overstock problems in a real-life Purchase-to-Pay process and traditional/object-centric process mining.

Overstock problems in a real-life Purchase-to-Pay process: In Purchase-to-Pay processes, companies across different industries face challenges like supply bottlenecks and often maintain high stock levels to ensure product availability, leading to frequent overstock situations and inefficiencies in inventory structure. Overstock, or excess stock without demand, hinders flexibility in responding to customer demand changes, adversely affecting sales and costs [19,21]. The studies [8,12,14,15,19] have analyzed the root causes of overstock in efforts to mitigate its impact. In [8,14,15], a mathematical model from operations research is used to optimize the inventory structure, the *economic order quantity* calculation. The mathematical model, which relies on assumptions such as constant demand and ordering costs, is often not applicable in real-world scenarios. In [12], the business process management method *process mapping* is used to visualize the control-flow of the supply chain processes. In [19], the *fishbone diagram* is used, which also belongs to the business process management, to determine the root causes of overstock. The applied operations research mathematical model and the business management methods lack the capability for event-data-driven process analysis and optimization, highlighting a research gap in identifying and addressing overstock patterns through process mining techniques.

Traditional Process Mining in ERP: Recent research highlights process mining's role in improving logistics and supply chain management. A systematic review demonstrated process mining's utility in identifying inefficiencies and aiding data-driven decision-making within manufacturing organizations [4]. Another study introduced a maturity model for integrating process mining in supply chain management, offering a framework for organizations to assess readiness and pinpoint areas for improvement [11]. Further, research on optimizing order processing with process mining has shown its effectiveness in enhancing order processing efficiency, offering valuable insights for business practices [20]. These studies underscore process mining's significant benefits in streamlining supply chain and logistics operations, notably in the Purchase-to-Pay process.

Object-centric Process Mining in ERP: Object-centric process mining enhances traditional process mining by linking events to multiple objects, offering a more nuanced view of complex processes [1]. Applied particularly in SAP ERP systems, methodologies have evolved from artifact-centric approaches [18,17] to object-centric behavioral constraint models (OCBC) [16], and multi-graph techniques for capturing interactions among different object types [2,5]. These developments utilize object-centric event logs, following the OCEL specification [10], with research also exploring semi-automated extraction from SAP ERP systems [7] and practical applications in processes like Purchase-to-Pay [6]. Despite challenges in data extraction and analysis complexity, such studies indicate potential for process improvement.

Industry adoption, exemplified by Celonis, reflects this shift. Celonis integrates object-centric concepts through features like multi-event logs and the process sphere, offering advanced process visualizations and analysis capabilities to pinpoint inefficiencies. Yet, the broader application of object-centric process mining in the industry remains limited, highlighting a gap for further exploration and documented real-life implementation studies to harness its full business potential.

3 Preparation

In this section, we prepare the object-centric process mining case study. We present the defined objectives and the target data format of the object-centric event data in terms of Object-to-Object (O2O) and Event-To-Object (E2O) relationships.

3.1 Definition of Objectives

From the logistics management of the pet retailer, we know that the current overstock is worth several million euros. The goal is to optimize the inventory structure to build up less overstock in the future. The following research questions aim to create transparency about the Purchase-to-Pay process activities leading to overstock by using object-centric process mining:

1. How do the relevant objects in the Purchase-to-Pay process behave and interact with each other?
2. Which behavior and interaction of the objects in the Purchase-to-Pay process lead to overstock?

3.2 Conceptualization

To achieve the mentioned objectives and to answer the research questions, we focus on the purchase part of the Purchase-to-Pay process. The considered part refers to

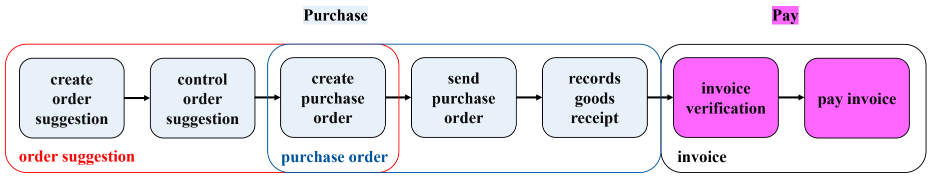

Fig. 2. Standardized representation of a Purchase-to-Pay process

all process activities from predicting demand, ordering goods to goods receipt [13]. Accordingly, we assume that the predicted demand is equal to the actual demand. In reality, these may differ from each other. The Purchase-to-Pay process can be simplified as depicted in Figure 2.

The figure shows that there are two sub-processes in the purchase part of the Purchase-to-Pay process: (1) order suggestion process and (2) purchase order process. In the order suggestion process, a merchandise planning tool suggests an order regarding multiple factors like predicted demand, stock levels, and additional factors. These suggestions are manually controlled and transformed into a purchase order. In the purchase order process, a purchase order is created and sent to the supplier. Afterward, the purchase order gets delivered.

The relevant object types of a general Purchase-to-Pay process are the following: *order suggestion, order suggestion item, purchase order, purchase order item*, and *material*. To show the cardinalities of the Object-to-Object (O2O) relationships of the object types we use a "UML-like notation" (see Figure 3) [1].

The figure illustrates classes and associations where an *order suggestion* includes one or more *order suggestion item(s)*, and a *purchase order* contains one or more *purchase order item(s)*, indicated by *1* and *1..**. An *order suggestion* may or may not refer to a *purchase order*, shown by *0..1*, highlighting that not every suggestion leads to a purchase order. Similarly, the relation holds for *order suggestion item* and *purchase order item*. Quantities, though not shown in the figure, are important; for instance, an *order suggestion item* for 100 pieces might become a *purchase order item* for 50 pieces. Each item links to one *material*, which can be suggested or purchased any number of times, indicated by *0..**, reflecting the varying frequency of suggestions or purchases for different materials.

The relevant events of a general Purchase-to-Pay process are events of the event types: *create order suggestion, control order suggestion, create purchase order, send purchase order*, and *records goods receipt*. The cardinalities from Event-to-Object (E2O) relationships and from Object-to-Object (O2O) relationships are depicted in Figure 4 [1].

In the figure, the left area indicates how often the events of the event types are executed for objects of the object types on the right side. The right area indicates how many objects of the object types are involved in events of the event types on the left side [1]. For example, the event type *create order suggestion* occurs once for each *order suggestion* and once for one or more *order suggestion item(s)*. The event type *control order suggestion* occurs once or more for each *order suggestion* and for each *order suggestion item*.

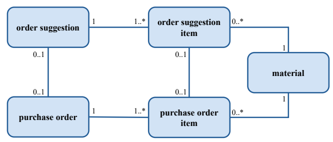

Fig. 3. Object-to-Object (O2O) relationships

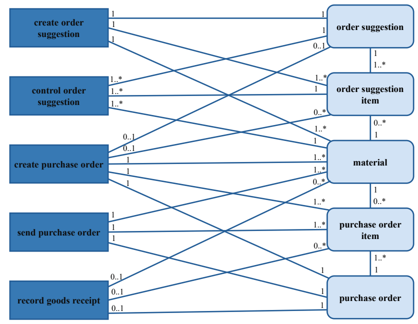

Fig. 4. Event-to-Object (E2O) relationships where the left area shows event types and the right area shows object types including Object-to-Object (O2O) relationships

Next to the O2O and E2O relationships, we assume that one *order suggestion* can be completely, partially, or not transformed into one *purchase order*. One *purchase order* can also include one or more *purchase order items* that were not suggested (see Figure 5).

4 Extraction and Transformation of Source Data

In this section, we present the extraction of the event data from Celonis and the data processing steps to create an object-centric event log using OCEL Standard format[4].

We explain how we extract the event data for the conceptually defined objects and events. For this case study, we extract all required event data for the Purchase-to-Pay process from Celonis. All event data in Celonis are previously extracted from two source systems: (1) Logomate[5] and (2) SAP[6].

The Logomate information system is a tool designed to optimize inventory management and replenishment processes in retail and distribution. Its primary purpose is to automate order suggestions. This system analyzes sales data, inventory levels, and other relevant parameters to forecast demand and recommend the most efficient purchase orders. This helps in maintaining optimal stock levels, reducing overstock or stockouts, and improving overall inventory management efficiency. The system employs algorithms to predict future demand patterns and suggests purchase orders accordingly, ensuring that inventory is aligned with anticipated sales. The order suggestion process and the purchase order process can be described as follows concerning the source systems: Logomate suggests automatically at the item level which *materials* should be ordered, in which quantity, and at what time. The automatic generation of the *order suggestion* is rule-based. The *order suggestion* is checked manually by an employee, who decides whether the *order suggestion*

[4] `https://ocel-standard.org`
[5] `https://www.remira.com/en/products/purchasing/logomate`
[6] `https://www.sap.com`

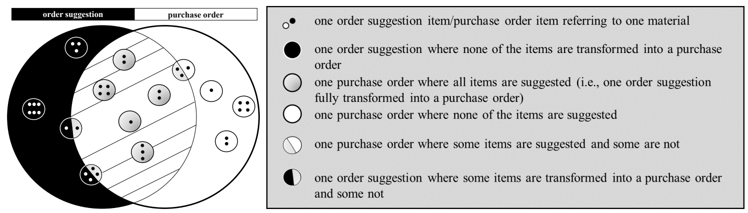

Fig. 5. Connection between order suggestions and purchase orders

should be transformed into a *purchase order*. If the *order suggestion* is approved, then the *purchase order* is created in SAP and sent to the supplier. When the supplier delivers the *purchase order*, the receipt of the goods is posted in SAP.

For the Purchase-to-Pay process in this study, we extracted the necessary event data from the source systems, namely Logomate and SAP, using Celonis. In consultation with the logistics management of the pet retailer, we applied a set of criteria to pre-filter the event data for clarity and focus. These criteria included selecting data within the date range from January 1, 2023, to April 30, 2023. We specifically concentrated on events associated with one central warehouse located in Krefeld, Germany. Further, we filtered for materials that have a purchase order lead time of less than 10 days, are part of the standard assortment, and have a status of 'listed and active'. Then, we extract the event data from Celonis to a CSV file. The extraction, in general, produces three tables: (1) an order suggestion activity table (2) a purchase order activity table, and (3) a relationship table. These are related to

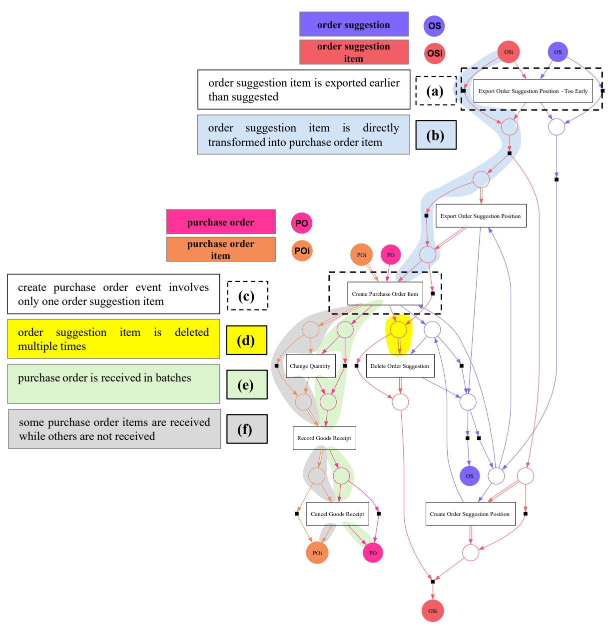

Fig. 6. An object-centric Petri net describing a holistic view of the Purchase-to-Pay process

the predefined conceptually relevant five object types, i.e., order suggestion, order suggestion item, purchase order, purchase order item, and material.

Data processing involves preprocessing extracted data and constructing an object-centric event log. Preprocessing focuses on removing duplicate events from the order suggestion and purchase order activity tables and filtering activities deemed relevant by logistics management. The data is then transformed into an object-centric event log in OCEL standard format, resulting in a log with 13,151 events, encompassing 1,918 order suggestions, 39,853 order suggestion items, 3,144 purchase orders, 49,176 purchase order items, and 5,144 materials.

5 Analysis

In this section, we present the discovered process models, we analyze the behavior and interaction of the relevant object types, and evaluate the results to answer the research questions using an iterative approach in cooperation with the logistics management.

Using a tool for object-centric process analysis [3], we generate object-centric Petri nets from the event log to model the Purchase-to-Pay process. These nets combine multiple control-flows for different object types, addressing concurrent actions and interactions within the process. They support conformance checking, comparing events to models to identify deviations. This approach provides a detailed view of the process phases. The choice of this open-source library over Celo-

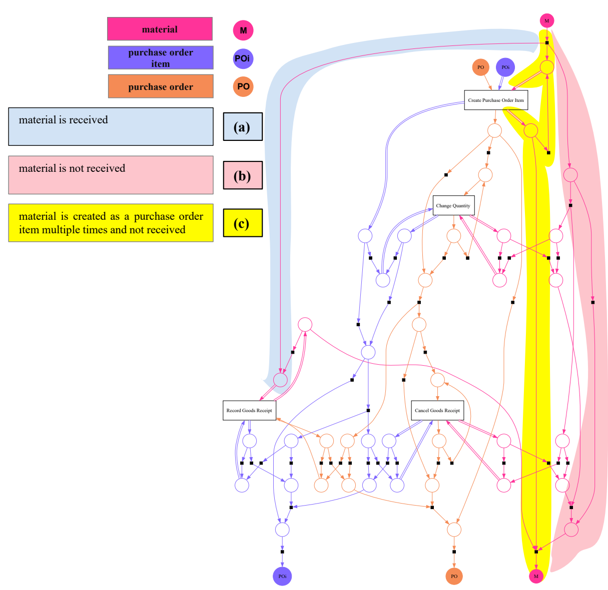

Fig. 7. An object-centric Petri net describing a detailed view with a focus on the material processes

Patterns	Assessment
order suggestion item is exported earlier than suggested	potential overstock
order suggestion item is directly transformed into purchase order item	high level of automation, but risk of errors
create purchase order event involves only one order suggestion item	inefficient
order suggestion item is deleted multiple times	rework

Table 1. Assessment of the patterns in the order suggestion processes

Patterns	Assessment
purchase order is received in batches	shortfall
some purchase order items are received while others are not received	shortfall
some materials are received while others are not received	shortfall
material is created as a purchase order item multiple times and not received	shortfall and potential overstock

Table 2. Assessment of the patterns in the purchase order processes

nis is based on its superior preprocessing, performance analysis, and conformance checking capabilities, offering a deeper and more accurate process analysis.

We developed two process models: a holistic view in Figure 6 and a detailed view on material processes in Figure 7. Figure 6 outlines a model with *order suggestion*, *order suggestion item*, *purchase order*, and *purchase order item* object types, incorporating eight event types like *create order suggestion position* and *record goods receipt*. Color coding differentiates between order suggestion and purchase order processes. The model illustrates the flow from creating order suggestions to the delivery outcomes of purchase orders, emphasizing the central role of *create purchase order item* in connecting sub-processes. We presented the process model describing the holistic view to the logistics management of the pet retailer and conduct interviews with the management to discuss noteworthy patterns and relevant aspects of the holistic view. The process model for order suggestion and order suggestion item processes outlines several patterns as depicted in Figure 6. These include the premature exportation of the order suggestion item, its direct conversion into a purchase order item, the creation of purchase orders that involve only a single order suggestion item, and the repeated deletion of an order suggestion item.

Figure 7 presents a detailed process model focusing on material processes with *material*, *purchase order item*, and *purchase order* as object types and includes *create purchase order item*, *change quantity*, *record goods receipt*, and *cancel goods receipt* as event types. Pink denotes material processes, while dark purple and orange signify purchase order item and purchase order processes. The model highlights the initiation of material processes with the creation of a *purchase order item*, shown by pink double lines in Figure 7, where one *material* becomes a *purchase order item* multiple times. After presenting this process model to the pet retailer's logistics management and conducting interviews, the model's significant patterns for material processes include the receipt and non-receipt of *material* and the repeated creation of *material* as a purchase order item without receipt.

In the order suggestion and order suggestion item processes, we discovered previously the following noteworthy patterns and these patterns are assessed as follows (see Table 1). Exporting *order suggestion items* to *purchase order items* prematurely results in overstock before actual demand. Assuming daily recalculations of predicted demand, early exports rely on less accurate forecasts, inhibiting response to demand fluctuations and increasing overstock risk. This manually decided pattern was observed 1,123 times in the review period. In the purchase order and purchase order item processes, we discovered previously the following noteworthy patterns which are assessed as follows (see Table 2). The pattern that *material* is created as a *purchase order item* multiple times and not received leads to overstock. This pattern and how it leads to overstock is illustrated by the following data-based example: predicted demands for *Material 1 (M1)* at 100 units and *Material 2 (M2)* at 200 units lead to a purchase order *(PO1)* containing both items. *M1* is not received, prompting a new order for *M1* through *PO2*, which also includes *Material 3 (M3)* due to minimum order requirements or capacity utilization of the truck. *M1* remains unreceived, while *M3* is received without initial demand, resulting in overstock. This pattern, affecting 171 materials, highlights the complexities in linking orders and understanding material flows, challenges not addressable with traditional process mining.

It is important to note, that we would not be able to detect these patterns using traditional process mining approaches. For example, in the previously explained pattern, we would not be able to establish the link between PO1 and PO2 based on M1 and analyze the interaction of these objects. Instead, we could only analyze the life cycle of a single object (i.e. one purchase order).

The two identified patterns leading to overstock are not specific to the pet retailer's environment, indicating that these patterns may be generalizable and can occur in Purchase-to-Pay processes in different sectors. The initial research questions can be answered as follows:

1. **How do the relevant objects in the Purchase-to-Pay process behave and interact with each other?** We developed two object-centric Petri net models: a holistic model covering *order suggestion, order suggestion item, purchase order*, and *purchase order item*, starting with order suggestions merging into a single suggestion and focusing on the *create purchase order item* activity, including *purchase order* statuses. The second model, centered on materials, features *material*, *purchase order*, and *purchase order item*, emphasizing the repeated listing of *material* as a *purchase order item* and tracking its reception.
2. **Which behavior and interaction of the objects in the Purchase-to-Pay process lead to overstock?** In analyzing the Purchase-to-Pay process, two main patterns contributing to overstock were identified. The first involves prematurely converting *order suggestion items* to *purchase order items*, leading to overstock before actual customer demand arises, noted 1,123 times in the analysis period. The second pattern features *materials* repeatedly listed in *purchase orders* but not received, often including new items without demand, further contributing to overstock, observed in 171 materials during the review period.

After discussing these patterns with the logistics management of the pet retailer, they acknowledged the importance of these findings in understanding and reducing overstock within the Purchase-to-Pay process.

6 Improvement

In this section, we derive important improvement measures for the two detected patterns leading to overstock in cooperation with the logistics management of the pet retailer, which need to be operationalized:

1. **Adjustment in Order Suggestion Process:** Within the Logomate system, a modification is required to manage the premature export of order suggestion items. The current system allows manual overrides that can lead to early exports. This capability should be restricted, with overrides permitted only in exceptional circumstances, such as an impending promotional event. This change aims to align order suggestions more closely with actual demand, reducing the likelihood of overstock due to premature ordering.
2. **Technical Modification in Purchase Order Process:** A dual-system adjustment involving Logomate and SAP is necessary. When a material, listed as a purchase order item, experiences a delivery failure, an automated notification should be sent to the designated employee. This notification's role is to prompt an inquiry into the cause of these delivery issues, such as potential shortages at the supplier's end. Furthermore, Logomate should temporarily cease generating order suggestions for the affected material until the supplier confirms its availability. This approach ensures that orders are not placed for materials facing supply chain disruptions, thereby preventing overstock.

The logistics management of the pet retailer acknowledged, that there will be less overstock if these measures are operationalized. Future research will focus on a more comprehensive analysis of the overstock problems. The aim is to assess whether the identified patterns of overstock are applicable across various industry sectors. This will involve establishing a systematic framework for the automatic detection of these patterns, enhancing the efficiency of the Purchase-to-Pay process. Additionally, we plan to refine and implement specific strategies to effectively address the overstock challenge, based on the insights gained from this study.

7 Conclusion

In this paper, we addressed overstock problems in a real-life Purchase-to-Pay process of a pet retail company. Through the adapted PM^2 methodology, we identified key patterns contributing to overstock and developed corresponding improvement measures [9]. The use of object-centric process mining was crucial in uncovering these patterns, demonstrating its utility in such analyses. The study indicates that these overstock patterns may be common across Purchase-to-Pay processes in different sectors. This observation suggests a wider relevance for the proposed solutions. Future efforts will aim to deepen the understanding of these patterns, improve automatic detection methods, and apply effective overstock reduction strategies across various industries.

References

1. van der Aalst, W.M.P.: Object-centric process mining: Unraveling the fabric of real processes. Mathematics **11**(12), 1–11 (2023)
2. van der Aalst, W.M.P., Berti, A.: Discovering object-centric petri nets. Fundam. Informaticae **175**(1-4), 1–40 (2020)

3. Adams, J.N., Park, G., van der Aalst, W.M.P.: ocpa: A python library for object-centric process analysis. Software Impacts **14**, 100438 (2022)
4. Alnahas, J.: Application of process mining in logistic processes of manufacturing organizations: A systematic review. Sustainability **15**(15), 11783 (2023)
5. Berti, A., van der Aalst, W.M.P.: OC-PM: analyzing object-centric event logs and process models. Int. J. Softw. Tools Technol. Transf. **25**(1), 1–17 (2023)
6. Berti, A., Jessen, U., Park, G., Rafiei, M., van der Aalst, W.M.P.: Analyzing interconnected processes: using object-centric process mining to analyze procurement processes. International Journal of Data Science and Analytics (2023)
7. Berti, A., Park, G., Rafiei, M., van der Aalst, W.M.P.: A generic approach to extract object-centric event data from databases supporting sap erp. Journal of Intelligent Information Systems (2023)
8. Biswas, S.K., Karmaker, C., Islam, A.: Analysis of different inventory control techniques: a case study in a retail shop. Journal of Supply Chain Management Systems **6**(3), 35–45 (2017)
9. van Eck, M., Lu, X., Leemans, S., van der Aalst, W.M.P.: PM^2: A Process Mining Project Methodology. Springer (2015)
10. Ghahfarokhi, A.F., Park, G., Berti, A., van der Aalst, W.M.P.: OCEL: A standard for object-centric event logs. In: ADBIS 2021 Short Papers. Communications in Computer and Information Science, vol. 1450, pp. 169–175. Springer (2021)
11. Jacobi, C., Meier, M., Herborn, L., Furmans, K.: Maturity model for applying process mining in supply chains: Literature overview and practical implications. Logistics Journal: Proceedings **2020**(12) (2020)
12. Jie, F.: Supply chain design for global competitiveness. In: Proceeding International Conference on Science and Engineering. vol. 1 (2017)
13. Khabbazi, M., Hasan, M.K., Sulaiman, R., Shapi'i, A.: Business process modeling for domain inbound logistics system: Analytical perspective with bpmn 2.0. Journal of Basic and Applied Scientific Research **3**(9), 569–578 (2013)
14. Kusuma, Y.A.: Supply arrangement of raw material and sugar stock to organize overstock risk in warehouse. Journal of Physics: Conference Series **1375**(1), 012048 (2019)
15. Lestari, N.F., Handayati, Y.: Analysis of inventory management in order to reduce overstock (case study of tvf footwear). International Journal of Current Science Research and Review **5**(09) (2022)
16. Li, G., de Carvalho, R.M., van der Aalst, W.M.P.: Automatic discovery of object-centric behavioral constraint models. In: Business Information Systems - 20th International Conference, BIS 2017, Poznan, Poland, June 28-30, 2017, Proceedings. Lecture Notes in Business Information Processing, vol. 288, pp. 43–58. Springer (2017)
17. Lu, X., Nagelkerke, M., van de Wiel, D., Fahland, D.: Discovering interacting artifacts from ERP systems. IEEE Trans. Serv. Comput. **8**(6), 861–873 (2015)
18. Nooijen, E.H.J., van Dongen, B.F., Fahland, D.: Automatic discovery of data-centric and artifact-centric processes. In: BPM 2012 International Workshops. Lecture Notes in Business Information Processing, vol. 132, pp. 316–327. Springer (2012)
19. Rahansyah, V.Z., Kusrini, E.: How to reduce overstock inventory: A case study. International Journal of Innovative Science and Research Techno (2023)
20. Schuh, G., Gutzlaff, A., Cremer, S., Schoppen, M.: Understanding process mining for data-driven optimization of order processing. Procedia Manufacturing **45**, 417–422 (2020)
21. Zakka, U.R., Khairunisa, A.: Integration of deterministic and probabilistic inventory methods to optimize the balance between overstock and stockout. IOP Conference Series: Materials Science and Engineering **722**(1) (2020)