Automated, Standardized, and Adaptive Abstraction of IoT Data into Process-Level Events

Edyta Brzychczy[1,*], Harry H. Beyel[2], Joscha Grüger[3,4], Krzysztof Kluza[1], Lukas Malburg[3,4], Giovanni Meroni[5,6] and Pedro Valderas[7]

[1]AGH University of Krakow, Poland
[2]Chair of Process and Data Science, RWTH Aachen University, Germany
[3]German Research Center for Artificial Intelligence (DFKI), Germany
[4]Business Information Systems II, Trier University, Trier, Germany
[5]Technical University of Denmark, Denmark
[6]University of Insubria, Italy
[7]Universitat Politècnica de València, Spain

Abstract

The increasing use of Internet of Things (IoT) technologies in industrial and cyber-physical environments creates new opportunities for process mining, but also reveals a fundamental data challenge. While IoT systems generate large volumes of heterogeneous, low-level sensor data, process mining requires structured event logs with cases, activities, timestamps, and attributes. This paper investigates the problem of converting raw IoT data into process-level event logs as a key research challenge at the intersection of business process management and IoT. It structures the problem into several dimensions, including automated event recognition, case identification, semantic modeling, IoT-enriched event log formats, adaptive abstraction mechanisms, and the role of large language models in abstraction pipelines. Based on our review of the state of the art, the paper identifies major research gaps, such as limited automation and adaptability, insufficient support for temporal dynamics and explainability, fragmented end-to-end solutions, and the lack of benchmark datasets and reproducible evaluation methods. Finally, it also outlines recommendations to guide future work toward more reliable and reusable IoT-driven process mining solutions.

Keywords

Process Mining in IoT, Event Abstraction, IoT Event Logs, Case Identification, Sensor Data Analytics

1. Introduction

Business Process Management (BPM) treats processes as main artifacts to be modeled, monitored, analyzed, and continuously improved [1]. More and more often, these processes are executed in environments enriched with Internet of Things (IoT) technologies: smart products, connected homes, factories. Also heavy industries generate massive amounts of time-stamped low-level sensor data that reflects, at a fine granularity, how a process is actually being carried out. This creates an opportunity for BPM to ground process models, conformance analysis, and improvement decisions in objective, sensor-based evidence rather than in self-reported or manually logged data alone.

Process Mining (PM) is one of the primary BPM disciplines positioned to exploit this opportunity, since it discovers, monitors, and enhances processes directly from event data [2]. However, PM algorithms require structured event logs (with case ID, activity, timestamp, and attributes), while raw IoT data is unstructured, high-frequency, and physical rather than process-level in nature. Without systematic preprocessing and abstraction, this data remains underused as a source of hidden knowledge about process execution in many companies across domains such as smart factories, manufacturing, healthcare, and heavy industries.

10th International Workshop on BP-Meet-IoT (affiliated with CBI/EDOC 2026) ver. 15.09.2026
*Corresponding author.
brzych3@agh.edu.pl (E. Brzychczy); beyel@pads.rwth-aachen.de (H. H. Beyel); grueger@uni-trier.de (J. Grüger); kluza@agh.edu.pl (K. Kluza); malburg@uni-trier.de (L. Malburg); giom@dtu.dk (G. Meroni); pvalderas@dsic.upv.es (P. Valderas)

Consider a manufacturing cell where a CNC machine is instrumented with vibration, temperature, and power-consumption sensors sampled at sub-second intervals (Figure 1). These sensors continuously report physical measurements, which by themselves carry no notion of a discrete process activity. A process analyst, however, is interested in high-level events such as "Tool change," "Drilling," or "Batch completed," each of which must be inferred from patterns across multiple correlated signals over time, and linked to the specific production order (case) to which it belongs. We use this example throughout the paper to illustrate the challenges involved in abstracting IoT data into process-level events.

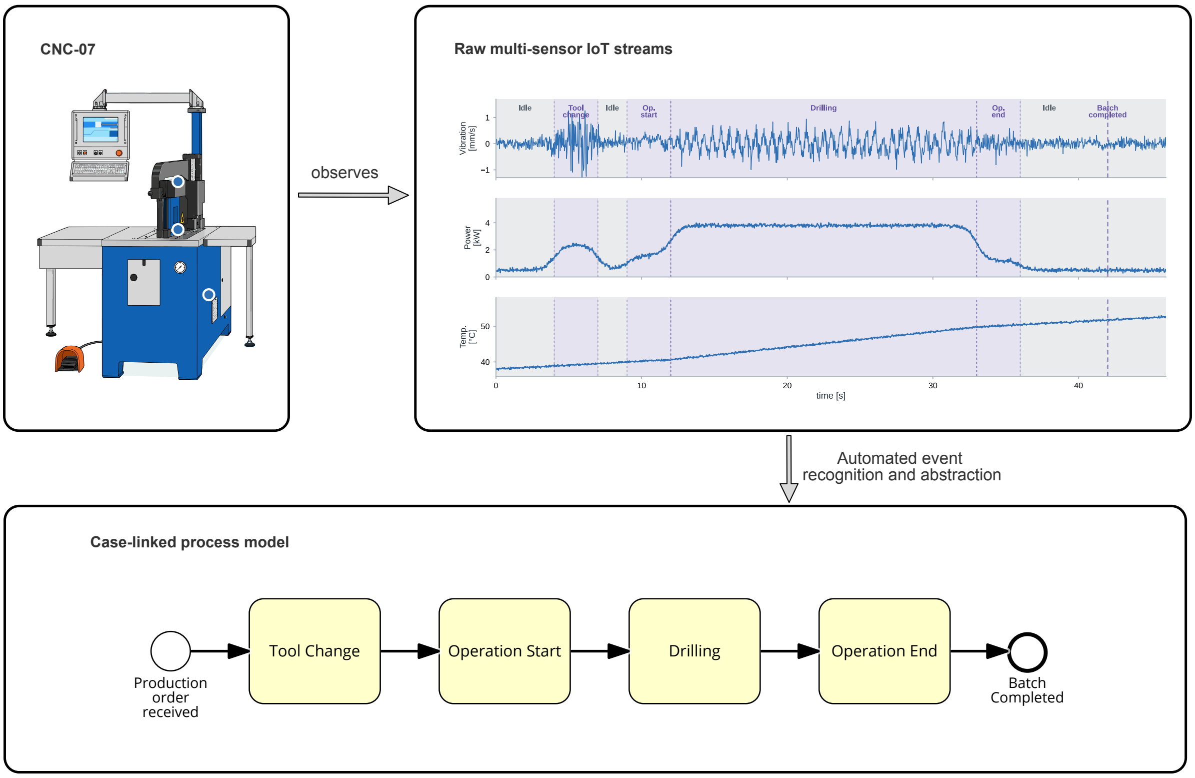

Figure 1: CNC machine process

The primary challenge is to transform heterogeneous, low-level IoT sensor data streams into high-level event logs that are directly usable for process mining as a basis for process modeling and analysis. It is one of the key challenges at the intersection of business process management (BPM) and the IoT domain [3]. Addressing this challenge requires automated event recognition (event abstraction), case identification (event correlation), multi-level abstraction, and semantic enrichment. The developed frameworks (concepts) should work across domains, leverage machine learning (ML) with domain knowledge and ontologies, and provide output event logs in a standardized, interoperable format.

The main contributions of this paper are:

- Identification of six problems (sub-challenges) that should be addressed in the scope of automated, standardized, and adaptive abstraction of IoT data into process-level events.
- Formulation of seven research gaps to be addressed by research and industrial communities.
- Definition of short-, medium-, and long-term recommendations for actions that should be taken to address the formulated gaps and develop the field of process mining based on IoT data.

We selected these six sub-challenges using two complementary criteria: coverage of the essential pipeline stages required to transform raw IoT sensor data into process-level events – abstraction, correlation, representation, and automation – and recurrence as an open problem across prior surveys and manifestos on IoT-driven process mining [3, 4, 5]. Each sub-challenge was subsequently elaborated by co-authors with direct research experience in the corresponding area, ensuring that the state-of-the-art discussion in Section 3 reflects both breadth of coverage and depth of expertise.

The paper is structured as follows. In Section 2, we identify six sub-challenges within automated, standardized, and adaptive abstraction of IoT data into process-level events. Section 3 presents the state of the art concerning each of the defined sub-challenges. In Section 4, we identify gaps that should be addressed in this scope, and in Section 5, we present the follow-up recommendations for concrete actions that should be taken. In section 6 we conclude the paper.

2. Sub-challenges and Problem Decomposition

We divided our challenge into six sub-challenges that cover the minimal and essential pipeline for IoT-to-process abstraction, addressing the most critical and widely recognized bottlenecks in the field. They are characterized in the following subsections.

2.1. Automated Event Recognition

Automated event recognition is a key research challenge in the automated, standardized, and adaptive abstraction of IoT data into process-level events. It refers to the ability to automatically and continuously identify events that are meaningful from a process perspective from heterogeneous streams of low-level IoT sensor data, which are typically high-frequency, noisy, and uncertain. In our running example, this means detecting the transition from "Tool change" to "Drilling" from a joint pattern in vibration and power signals, rather than from any single threshold crossing on either signal alone.

Addressing this challenge requires bridging the semantic gap between raw physical observations captured by IoT devices and the conceptual elements of process models by correlating multiple signals, reasoning over temporal and spatial patterns, and taking the operational context into account. Furthermore, automated event recognition must cope with the diversity of devices, data formats, and sampling rates, adapt to changes in the environment or the underlying process, and scale to (near) real-time operation. All this should be achieved while minimizing manual configuration and reliance on expert knowledge, so that the resulting events are reliable, standardized, and reusable for subsequent process monitoring, analysis, and automation tasks.

2.2. Case Identification

The presence of a case ID, which links events to a specific process instance, is a crucial requirement for an event log [2]. In raw IoT data, this core element of the event log is often not directly available. The main reason is a result of sensor orientation of data, which includes raw sensor streams attached to the sensor or machine name rather than to the process instance. Moreover, IoT streams frequently represent continuous signals rather than explicit discrete events, so process events (and their boundaries) must first be derived through event abstraction before they can be linked to a specific instance. In our running example, the challenge could be to determine which production order the detected "Tool change" event belongs to, when the sensor stream itself carries no order number and only reflects continuous machine state.

Addressing this challenge requires a systematic strategy for case identification that combines event abstraction with event correlation, using time-based rules, machine context, and (where available) identifiers from MES or ERP systems. In practice, it also requires validating alternative case notions (e.g., cycle vs. shift) against the analytical objective and domain knowledge to ensure the resulting event log is both interpretable and appropriate for process discovery and conformance checking.

2.3. Usage of Semantic Models and Ontologies

For automated abstraction of IoT data into process-level events, explicit semantic models and ontologies can provide a shared conceptual layer between signals and process concepts. Such semantic representations can describe (1) IoT sensors and observations (e.g., units, sampling rates, locations), (2) relevant entities (e.g., machines, products), (3) relations between sensors, the entities, and the process, and (4) process knowledge such as activities, required inputs/outputs, and pre-/post-conditions. As illustrated in Figure 2, this knowledge can be used to automatically derive events from IoT sensor data. In our running example, an ontology could formally state that a sustained vibration signature above a threshold, combined with tool rotation, constitutes the activity "Drilling" for this machine type.

However, the use of semantic models and ontologies is accompanied by the typically high knowledge acquisition and modeling efforts required for building such semantic representations by hand. In addition, it needs to be ensured that the semantic representation is formalized consistently and kept aligned with possibly evolving equipment, process variants, and changing sensor setups. Moreover, IoT data is noisy and incomplete, making purely symbolic mappings brittle without mechanisms to handle uncertainty, drift, and validation. Hence, a central open problem is how to reduce the knowledge acquisition and modeling efforts while preserving correctness, coverage, and portability of the abstraction process across application scenarios in IoT environments.

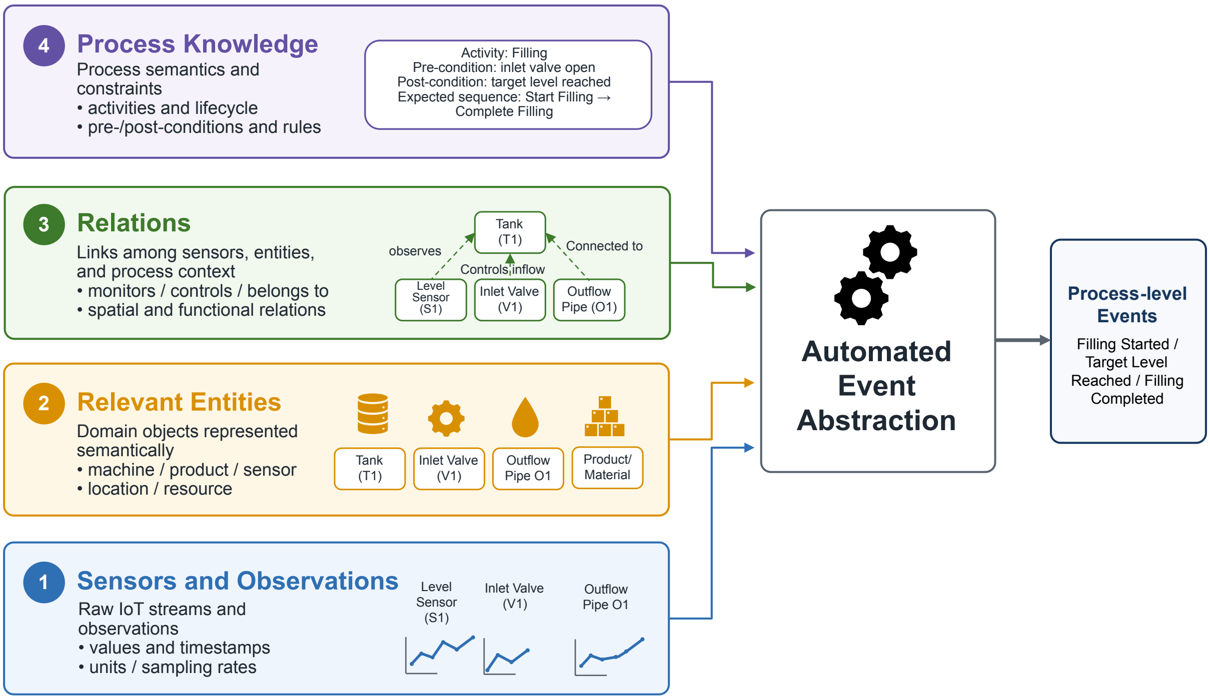

Figure 2: Four complementary semantic layers supporting the abstraction of IoT sensor data into process-level events.

2.4. Standardized, IoT-enriched Event Log Formats and Evaluation

XES is the de-facto standard for representing process event logs in a common and widely used format. In recent years, several initiatives have developed new event log formats, such as OCEL, overcoming problems by using the XES standard. However, all currently available process event log standards have in common that they are not directly tailored for use in IoT environments. In particular, XES and OCEL primarily represent discrete timestamped events, whereas high-frequency IoT data often encodes process-relevant information in continuous temporal patterns, durations, and gradual transitions that cannot be represented explicitly without prior abstraction. For example, they lack the ability to represent high-frequency streams, time-continuous observations, context, provenance, or multi-entity/objects.

In our running example, representing the vibration data stream that underlies a single "Tool change" event is exactly the kind of information current log formats struggle to retain. Sensor data is often omitted or stored outside the event log, without reference to the process level, which can lead to biased or incorrect process analyses and makes results challenging to reproduce. A pivotal challenge is to define a community-accepted representation for IoT-enriched event data that balances expressiveness and usability: the format should link events to underlying sensor observations (e.g., via time windows and referenced entities), support multi-entity relations, and remain interoperable with established process mining toolchains.

2.5. Domain-agnostic, Adaptive Abstraction Mechanisms and Level of Automation

IoT events can rarely be directly used to support business processes. Indeed, IoT events often represent one or more properties of the real-world environment (e.g., temperature and position), generated by a sensor either periodically or whenever one of these properties changes. Conversely, a business event usually represents the activation and termination of an activity, often represented by a text label. Therefore, frameworks that can aggregate, filter, and detect patterns in IoT events are required to interpret those events and produce a corresponding business event.

With reference to our CNC running example, such a framework should enable event abstraction independently of the specific machine type, manufacturer, or sensor configuration, so that the same abstraction mechanism can be applied across heterogeneous CNC systems with minimal reconfiguration.

However, existing abstraction mechanisms are often bound to a specific process, and in some cases to the hardware of the sensors being deployed. Thus, whenever a new process is introduced, or even when sensors are replaced with ones from a different manufacturer, a substantial engineering effort is typically required to reconfigure the whole infrastructure.

2.6. LLM Usage for Abstraction of IoT Data

This sub-challenge differs qualitatively from the mechanisms discussed so far: unlike CEP, ontologies, or model-driven engineering (MDE), which require the abstraction logic to be encoded upfront, Large Language Models (LLMs) as a supporting mechanism offer a way to generate plausible semantic interpretations of sensor patterns directly, reducing the manual knowledge-engineering effort that would otherwise be required. This makes them attractive where labels or domain rules are missing, but it also introduces a risk the other approaches lack: non-deterministic and occasionally hallucinated output. In this context, LLMs are not intended to replace established abstraction mechanisms, but rather to complement them by providing semantic interpretation and labeling capabilities.

In our running example, an LLM could be prompted to assign the label "Drilling" to a discovered cluster of vibration and power patterns based on summary statistics, without an analyst manually inspecting each cluster [6]. This idea is not specific to manufacturing: in elderly-care monitoring, LLMs have similarly been used to generate event records directly from motion, door, and appliance-use sensor readings and to merge them into a unified event log capturing activities such as meal preparation [7], illustrating that LLM-assisted abstraction generalizes across domains.

However, the key sub-challenge is integrating LLMs into IoT-to-event abstraction pipelines in a reliable, repeatable, and technically usable manner. Defining clear roles for LLMs within abstraction workflows requires constraining their outputs into structured, verifiable forms that can be consistently used by downstream process mining techniques. The non-deterministic and occasionally hallucinated behavior of LLMs directly threatens standardized and adaptive abstraction, undermining both the reproducibility required for standardization and the trust required for adaptive, low-supervision deployment.

3. State of the Art

In this section, we present the state of the art, structured around the defined sub-challenges. The literature reviewed in this section reflects the authors' domain expertise across the involved research communities (process mining, BPM, IoT, and semantic technologies), complemented by targeted literature searches for each sub-challenge and by snowballing from key surveys on event abstraction [4] and process mining on sensor data [5], which provided an initial map of the relevant literature landscape. While this approach does not constitute an exhaustive systematic literature review, it reflects a deliberate selection strategy tailored to the exploratory, multi-perspective nature of this position paper.

3.1. Automated Event Recognition

A substantial body of research has tackled the challenge of automated event recognition primarily under the umbrella of event abstraction for process mining [8]. Foundationally, [4] systematizes this space through a taxonomy and literature review of event abstraction techniques, clarifying how low-level observations can be lifted to higher-level events and highlighting recurring design dimensions (e.g., granularity, supervision, online vs. offline abstraction). Complementing this, a recent survey focusing specifically on process mining on sensor data [5] emphasizes that abstraction and contextualization are the central obstacles when moving from raw sensor signals to event logs suitable for downstream discovery and conformance checking. [9] proposes a framework for process discovery from raw sensor data that explicitly addresses the multiple aggregation steps necessary to move from distributed, noisy sensor readings to discrete activity instances and ultimately to process models. Their modular pipeline includes case/event correlation, activity discovery, event abstraction, and process discovery, emphasizing flexible application across datasets while highlighting the core challenge of recognizing meaningful activities from continuous raw events.

A prominent line of work uses Complex Event Processing (CEP) [10]. CEP is a paradigm that, through aggregation and composition, supports iterative abstraction of complex events. It builds on the underlying technology and brings events closer to an application's semantics. In our setting, it is used to recognize activity executions by specifying (or generating) temporal patterns over sensor streams. Early and influential frameworks explicitly position CEP as the core mechanism to correlate low-level IoT signals into process events in (near) real time, often via multi-stage detection/refinement pipelines [11]. More recent contributions operationalize this idea by proposing architectures and tool-supported frameworks that generate event-processing services for detecting arbitrary process activities at runtime from IoT streams, aiming to reduce the manual effort traditionally required to implement abstraction logic [12].

Complementary to CEP-based solutions, a rapidly growing body of work investigates data-driven and learning-based event abstraction, aiming to infer event boundaries and semantics directly from sensor data. IoT Miner [6] exemplifies this direction by combining automated preprocessing, pattern discovery, and LLM-based labeling to transform raw sensor streams into high-level event logs. The approach reduces manual effort by using unsupervised techniques to identify recurring patterns and leveraging LLMs to assign interpretable activity labels, thereby improving portability across domains while maintaining semantic interpretability. Similarly, [7] proposes an LLM-based abstraction and integration pipeline that generates event records from raw IoT readings and merges multiple data sources into a unified event log. Their results demonstrate that LLMs can effectively support semantic interpretation and integration in scenarios where explicit activity labels or structured logs are missing. However, these approaches also introduce new challenges related to non-determinism, reproducibility, and robustness under noisy and evolving sensor conditions.

3.2. Case Identification

Methods related to the case identification task can be divided into four categories [13]: (1) without any additional input, using only event names, (2) based on a process model, also using timestamps of events, (3) based on event similarity or case identifier in the log, and (4) relying on correlation conditions, using data attributes. Most of the existing approaches assume a classic event-log structure (explicit activity labels, discrete events, or structured attributes), which raw sensor data typically does not provide (especially from the industrial domain). In the literature, only a few solutions for case ID identification without process models or activity labels exist [14].

Current solutions for case identification include:

- heuristic approaches, including a heuristic for discrete events based on expert knowledge as input to cluster events into cases [15], a framework proposing to derive case boundaries from empirically chosen machine data (e.g., status changes) [16] and heuristic using unique temporal patterns observed in event logs to distinguish case ID attributes from other attributes [17].

- rule-based approaches to distinguish entity detection in smart spaces scenarios based on the weighted average distance method [18] or to infer missing identifiers with the usage of Event Knowledge Graphs [19],
- time-series approach for machine cycle detection based on continuous variable in industrial setting [20],
- segmentation approaches, including segmenting a smart home log into habits with Chi-merge discretization method based on the structuredness and simplicity measures of Petri nets [21] and segmenting semi-structured event streams using a combination of syntactic similarity and model-based conformance checking [22].

Some approaches presented in literature apply to the low-level IoT data of a discrete nature [16, 18] or a continuous one [20, 23]. They are strongly related to process specificity, e.g., cyclic manufacturing process [16], cyclic mining process [23], or smart home scenarios with semi-cyclic character [18]. The attempt to create a domain-agnostic approach for case identification was undertaken in [20].

3.3. Usage of Semantic Models and Ontologies

In recent work, ontologies and semantic models have been proposed for IoT. The focus of these ontologies has changed from the creation of ontologies that are as comprehensive as possible (e.g., the Semantic Sensor Network (SSN) ontology [1]) to the development of ontologies that are lighter-weight and, thus, easier to adopt in practice (e.g., IoTStream [24]). Semantic models and ontologies provide a shared and machine-readable format to describe IoT data for integration and interoperability as well as the IoT environments, such as sensors and actuators.

Two widely used ontologies for representing IoT data are the Sensor, Observation, Sample and Actuator (SOSA) ontology [25], and IoTStream [24]. SOSA is an ontology to formally define sensor networks and their observations [25]. SOSA is a lightweight core ontology that builds the basis for the SSN ontology. This core of SOSA is composed of (1) sensors (i.e., physical or virtual) capable of observing properties of features of interest, (2) observations that define the relationship between different elements (i.e., the sensor, observed property, feature of interest, observation value), and (3) actuators performing actuations that cause a state change in a feature of interest. IoTStream is more specific and focuses on the treatment of streaming data [24]. The main purpose is to utilize semantic technologies to bridge interoperability and heterogeneity issues. For this purpose, metadata descriptions can be provided in IoTStream to help search for needed data by representing their properties, such as attribute values and types, finally making data analysis more accessible.

In contrast, there are ontologies that focus more on the representation of the IoT environment itself, for example, by representing the capabilities of IoT devices. One such upper ontology is *MASON* [26], which has been developed to conceptualize manufacturing domains. Similar to this is the OWL-DL *Manufacturing Service Description Language (MSDL)* ontology [27] that focuses on the manufacturing process as the core class of manufacturing services. The FTOnto domain ontology [28] represents the capabilities and IoT devices of a physical smart factory from Fischertechnik. Finally, there have recently also been initiatives at the European level to create a suite of shared ontologies as a common standard for semantic modeling to increase interoperability, such as in the context of *The Smart Applications REFerence Ontology (SAREF)*[2].

3.4. Standardized, IoT-enriched Event Log Formats and Evaluation

The search for a standard format for event logs that natively supports IoT data has become a key issue recently. Researchers have identified unique challenges in integrating IoT sensor streams with traditional process event data, leading to formal requirements for an IoT-enhanced event log model. For example, one study distilled ten key requirements (e.g., handling high-frequency sensor readings,

[1]https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/
[2]https://saref.etsi.org/

linking data to multiple context levels, and coping with noise) and showed that neither the XES nor OCEL standards fully meet them [29]. This realization has spurred several proposals of IoT-enriched log formats designed to bridge the gap. The DataStream XES extension [30] is one such effort: it extends the XES standard by introducing new constructs to connect time-series sensor data with process events. DataStream preserves rich context (e.g., sensor attributes, configurations, and units) alongside event logs, ensuring that IoT observations can be related to events, traces, or even across multiple cases as needed. By evaluating DataStream on real-world scenarios in manufacturing and transportation, the authors demonstrate that the extension could uniformly capture IoT readings without losing context. In a different paradigm, the Native IoT-Centric Event (NICE) log model adopts an object-centric approach to log design [31]. The NICE format has been defined with traceability and flexibility in mind, for instance, allowing events to be linked with IoT object states or environmental conditions, thereby minimizing information loss when merging sensor data with business events. Another recent proposal, CAIRO (focusing on knowledge-intensive, manual processes), introduced its own metamodel for IoT event logging, underscoring that each solution targets specific use cases and domains [32]. To counter this, a unified core metamodel for IoT-enhanced event logs has been put forward as an emerging standard [33]. This CORE model synthesizes the common concepts from prior proposals into a single schema (implemented as an OCEL 2.0 extension) intended to support a wide range of IoT-enhanced processes [33]. A prototypical implementation has shown that existing IoT logs (in formats like DataStream or NICE) can be translated into the CORE model, and multiple real-life-inspired use cases are used to validate that the model indeed covers the required features across domains.

Evaluation and benchmarking of IoT-driven event abstraction techniques remain challenging. Unlike traditional process logs, IoT-enhanced logs often lack widely available, ground-truth datasets for objective comparison [34]. Early works therefore rely on case studies and synthetic data to assess the effectiveness of event abstraction. For instance, researchers have generated artificial sensor event logs to simulate various process behaviors and then measured how well abstraction methods can recover high-level events [35]. Common evaluation metrics include conformance-based measures (fitness and precision of discovered models) and complexity or comprehensibility of the abstracted log [35]. There is no widely accepted open benchmarking for event abstraction.

3.5. Domain-agnostic, Adaptive Abstraction Mechanisms and Level of Automation

The problem of automatic and domain-agnostic event abstraction has been primarily tackled by designing architectures relying on CEP components. As already mentioned in Section 3.1, CEP makes possible to define arbitrary rules that aggregate and transform sensor events into business events. By externalizing this logic into rules, CEP components centralize complex temporal and aggregation behavior, allowing the rest of the system to remain simpler, loosely coupled, and event-driven.

To this aim, [36] presents a framework that relies on CEP to aggregate IoT sensor data into events that are then mapped to the process either as part of the activity lifecycle or as external events. [12] proposes a framework to automatically generate services from IoT event logs to detect when activities are executed. Such services are internally implemented as CEP rules. [37] presents a microservice-based architecture to make the integration between IoT devices, CEP, and BPMS implementation-agnostic. To do so, the architecture relies on ontologies to describe sensor data and process events. [38] proposes a framework to semi-automatically perform a set of structured steps to convert low-level IoT sensor data into higher-level process events. Steps are kept abstract, so that specific analysis techniques aimed at event extraction, abstraction, and correlation can be plugged-in.

To dynamically select IoT sensors and actuators that best fit the needs for a specific process, frameworks that rely on ontologies have been proposed. As mentioned in Section 3.3, ontologies can capture the capabilities of IoT devices, and their support for automatic querying and reasoning can be used to automate the identification and configuration of IoT devices.

To this aim, [39] introduces an ontology for process-aware scenarios in the Industrial Internet of Things, where raw sensor data are augmented with details about process activities and the physical production environment. This makes it possible to turn event logs into process graphs and to explore the connections between sensor data and process activities, and to run analytical queries over it. [40] makes use of an ontology to represent sensors and their capabilities, which is used to determine to what extent the IoT devices involved in a process can monitor the process. The ontology can also suggest which sensors should be included/replaced in the process to better monitor its execution. Similarly, [41] relies on an ontology to represent the capabilities of sensors, which can be referenced in the process to be executed through custom BPMN extensions.

CEP and ontologies introduce some level of automation in the configuration and enactment of an IoT-based BPM solution. However, manual intervention is still required to define the aggregation mechanisms through code, be it a query over an ontology or a CEP rule. To overcome this limitation, solutions exploiting the MDE have been proposed. MDE is a software development approach where models are the primary artifacts of development, and automated transformations are used to generate code, configurations, or other models from them. In MDE, developers focus on high-level, domain-specific models, and tools handle much of the implementation detail through model transformations and code generation.

To this aim, [42] proposes a framework that exploits SysML to represent sensors and their capabilities, and uses MDE to automatically extend process models with information on the sensors required for automating the process. [43] extends the BPMN 2.0 metamodel with IoT-specific modeling elements that represent sensors and actuators, making it possible for the process to react to sensor events and to trigger actuator actions. Conversely, [44] reuses the existing elements in the BPMN 2.0 metamodel to represent IoT elements, and makes use of the Node-red low-code platform[3] to implement the logic to generate business events from IoT events. In [45], the behavior of IoT sensors to handle sensor data is modeled as a business process (micro-process), which at runtime is then mapped to the global process to execute. [46] introduces a Domain-specific Language (DSL) to represent the conditions on sensor data that determine the occurrence of a business event. From that DSL, CEP rules can be automatically generated.

3.6. LLM Usage for Abstraction of IoT Data

LLMs are used with IoT data to convert raw sensor streams into meaningful insights by enabling natural-language interaction, fusing multimodal sensor inputs with contextual and domain knowledge, applying structured reasoning for decision-making, supporting synthetic data generation, automating preprocessing and feature extraction, and orchestrating IoT workflows [47, 48]. In process mining scenarios, especially when event logs and activity semantics are not directly available, LLMs are rarely used as standalone abstraction mechanisms. Instead, they function as a semantic layer within broader pipelines that transform raw sensor data into process-level evidence, typically combining signal preprocessing and data-driven structure discovery with LLM-assisted interpretation and labeling. Existing research on LLM usage in IoT-driven event abstraction for process mining mainly follows two complementary directions:

- Direct abstraction and integration from IoT sources are achieved by generating event records directly from raw IoT sensor readings and by merging data from multiple IoT sources into a unified event log suitable for process mining [7]. The contribution focuses on LLM-supported transformation from sensor observations to event records, as well as multi-source log integration.
- Unsupervised abstraction with LLM-based interpretable rules and labeling. In [49], LLMs are used to generate interpretable labeling rules as human-readable alternatives to conventional classifiers (e.g., decision trees). IoT Miner [6] uses LLMs to generate activity labels from cluster statistics guided by domain-specific prompts. LLMs are used in unsupervised abstraction pipelines, where clustering or segmentation structures sensor data and LLMs assign activity semantics. The approach also addresses lexical variability in labels by measuring similarity between generated labels and ground truth labels when available.

[3]See http://nodered.org

Beyond direct event abstraction, several works investigate the use of LLMs as general semantic reasoning layers for IoT data analysis. IoT-LLM [50] proposes a unified framework that enables LLMs to reason over real-world IoT sensor data by transforming raw time-series into interpretable descriptions, retrieving task-relevant domain knowledge, and guiding inference via structured chain-of-thought prompting. Similarly, Generative IoT (GIoT) systems [51] employ LLMs as a flexible reasoning layer over semi-structured IoT data, often deployed on edge servers for privacy and latency reasons. While these approaches are not explicitly designed for process mining, they demonstrate how LLMs can interpret heterogeneous IoT data, integrate domain knowledge, and perform structured reasoning capabilities that are increasingly reused in IoT-to-event abstraction pipelines. A similar pattern can be observed in LLM-assisted data-to-log extraction for process mining, where LLMs act as a reasoning layer over structured enterprise data [52]. In a complementary direction, conversational AI has been used to support natural-language interaction with data and process mining results, enabling non-expert exploration and explanation of manufacturing processes without altering the underlying abstraction pipeline [53].

4. Research Gaps and Opportunities

Based on the reviewed works, we identified the following research gaps, which should be addressed within this challenge:

Gap 1 – Limited automation and adaptability of IoT-to-process event abstraction

Current approaches for abstracting low-level IoT sensor data into process-level events still suffer from limited automation and adaptability. Many solutions depend on manually defined rules, handcrafted activity signatures, or process-specific configurations that require significant expert involvement during setup and maintenance. As a result, abstraction mechanisms are tightly coupled to particular processes, environments, or sensor configurations, and they do not generalize well when processes evolve, new variants are introduced, or sensors are replaced. Although data-driven and learning-based techniques, including recent LLM-assisted approaches, aim to reduce manual effort, they often require extensive tuning, retraining, or contextual reconfiguration to remain effective in dynamic IoT environments. Moreover, existing methods typically lack mechanisms for continuous adaptation to concept drift, changing operational conditions, or evolving process behavior, leading to degradation in abstraction quality over time. Consequently, there is still no widely applicable abstraction approach that can autonomously and continuously adapt to heterogeneous and evolving IoT settings while producing stable and reliable process-level events.

Gap 2 – Lack of generic and reusable methods for case ID identification

There is a lack of mature, generalizable methods that enable case ID identification directly from raw, mixed-type industrial IoT data in a manner that is domain-agnostic and independent of activity labels or process models. While some recent approaches operate without explicit activity labels they typically target specific process types, most often in cyclic or semi-cyclic settings, and rely on domain-specific heuristics or process-specific feature engineering. As a result, transferability across heterogeneous industrial processes remains unproven, which limits scalability and reusability.

Gap 3 – Unreliable evaluation and reproducibility under fuzzy abstraction targets

Evaluation and reproducibility are major challenges in IoT-to-process event abstraction, especially when LLMs are used. In IoT environments, activity boundaries and transitions are often unclear and depend on context, and reliable ground-truth event logs are rarely available or are created using heuristic rules. However, most evaluation methods still assume precise labels, exact timing, and a single correct abstraction, which does not reflect the inherently fuzzy nature of sensor-based data. These challenges are further increased by the non-deterministic behavior of LLMs and their sensitivity to prompt design, model versions, and generation settings, which can lead to different abstraction results for the same input. Existing studies typically evaluate abstraction results in isolation, such as label accuracy or trace similarity, and lack standardized methods to assess stability, semantic correctness, and usefulness for downstream process mining together. As a result, there is still no reproducible and comparable evaluation framework that explicitly considers ambiguity, multiple valid abstractions, and non-determinism in LLM-based IoT event abstraction.

Gap 4 – Insufficient modeling of temporal dynamics and transition states

A key limitation of current IoT-to-process event abstraction methods is that they do not explicitly model temporal dynamics or transitions. IoT sensor data often captures continuous, overlapping, and gradually changing behaviors, while process mining relies on discrete events with clear start and end times. Most abstraction pipelines, including those using LLMs, force continuous sensor data into fixed activity segments without representing transition phases, concurrent activities, or uncertainty at activity boundaries. This mismatch can be understood as a conflict between the generally fuzzy or unclear boundaries of physical processes, where a machine's transition from idle to active is gradual and depends on the context, and the discrete event semantics used in classical process mining, where events are assumed to occur at a specific moment in time [4]. As a result, this may lead to distorted event logs, unnecessary fragmentation, or incorrect labeling of ambiguous periods. Although some methods use heuristics or time-based rules, there is still no widely accepted approach that treats transitions, gradual change, or temporal uncertainty as explicit elements in process-level event construction.

Gap 5 – Fragmented integration of abstraction, correlation, and semantics

Current research typically addresses event abstraction, case identification, semantic modeling, and log standardization in isolation. Few approaches provide end-to-end pipelines where these concerns are jointly optimized and mutually informed. This fragmentation leads to brittle solutions where errors or uncertainty introduced at early stages (e.g., abstraction) propagate to later stages (e.g., process discovery) without feedback or correction mechanisms.

Gap 6 – Limited support for explainability and human-in-the-loop validation

As abstraction pipelines become increasingly automated and learning-based, especially with the use of LLMs, there is limited support for explainability and interactive validation. Domain experts often lack transparency into why specific sensor patterns are mapped to particular events or cases, which hinders trust, adoption, and iterative refinement in real-world deployments.

Gap 7 – Lack of publicly available, real-life labeled IoT benchmark datasets for method testing and evaluation

Progress in IoT-to-process mining research is constrained by the limited availability of publicly accessible real-world industrial IoT datasets for testing and validation of developed methods. Most datasets used in existing studies are synthetic, simulated, or collected in controlled environments such as smart homes or laboratory settings, and therefore do not sufficiently reflect the complexity and irregular behavior observed in real-world IoT-enriched processes. Moreover, publicly available datasets are stored in a fragmented manner at various websites/servers, making the search very difficult, so a curated repository of diverse, low-level industrial IoT datasets would be beneficial for the PM community.

5. Proposed Actions and Recommendations

Based on the identified gaps, we recommend the following actions that should be taken from short-term to long-term actions.

5.1. Short-Term Actions

We recommend the following actions in the nearest time:

- *Recommendation 1*: Research review on publicly available datasets appropriate for testing and benchmarking the approaches and methods from sensor data level to process level events, with evaluation of their structure and usefulness. Curation of the dataset repository or creation of the database of dataset links with detailed information.
- *Recommendation 2*: Research on modeling the temporal dynamics and transition states in raw data and their impact on the event abstraction process.
- *Recommendation 3*: Research on evaluation and reproducibility under fuzzy abstraction targets. Since many labels in the dataset are obtained from automation systems and are very often general, they can not be directly used as ground truth for abstraction evaluation.
- *Recommendation 4*: Develop a simple schema to document sensor data quality: sampling rates, missing-value patterns, sensor drifts, and preprocessing applied. If new datasets are shared, the inclusion of such metadata would benchmarking fairer, and the interpretation of why certain methods work better than others becomes more accessible.
- *Recommendation 5*: Develop a checklist for LLM-based experiments, including model name/version, used parameters, prompt templates, etc., to mitigate reproducibility problems. Also, standard ways of representing LLM outputs, such as JSON schemas, might improve reproducibility.

5.2. Medium-Term Directions

We recommend the following medium-term actions, also in cooperation with industrial partners:

- *Recommendation 1*: Creation of high-quality new datasets in cooperation with academic laboratories and industrial partners, ensuring their usefulness for the BPM and process mining community.
- *Recommendation 2*: Elaboration of more consistent frameworks for IoT-to-process event abstraction based on already developed concepts and methods (including event abstraction and case ID identification). This step requires a review of existing concepts and approaches, and creation of a taxonomy.
- *Recommendation 3*: Creation of automated and adaptable solutions for heterogeneous IoT settings, taking into consideration the usage of semantic models and ontologies.
- *Recommendation 4*: Develop baseline benchmarking pipelines for IoT-to-process event abstraction that cover the main stages from raw sensor streams to process-level event logs.
- *Recommendation 5*: Develop more generic and reusable methods for case identification directly from raw IoT data, without assuming the existence of explicit activity labels, process models, or predefined case notions.
- *Recommendation 6*: Create reusable ontology patterns, e.g., for machine states or product lifecycle stages, so that less effort is needed to create domain ontologies.

5.3. Long-Term Vision

We recommend the following long-term actions, requiring substantial efforts from academic and from industrial partners:

- *Recommendation 1*: Creation of an expert system for acquiring human expert knowledge useful for the abstraction process of IoT data into process-level events, ensuring explainability of inference and abstraction results.
- *Recommendation 2*: Creation of a dedicated AI-based solution (LLM-based and/or JEPA-based) for automated support of standardized and adaptive abstraction of IoT data into process-level events in a real-life setting.

Table 1
Overview of recommendations for advancing IoT-driven process mining, with target stakeholders.

Recommendation	Term	Target Stakeholder
Curate/review public datasets	Short	Academic
Schema for sensor data quality documentation	Short	Academic & Industrial
Evaluation & reproducibility under fuzzy targets	Short	Academic
Model temporal dynamics/transitions	Short	Academic
Checklist for LLM-based experiments	Short	Academic & Industrial
Create high-quality new datasets	Medium	Academic & Industrial
Generic/reusable case identification methods	Medium	Academic
Consistent frameworks and taxonomy	Medium	Academic
Baseline benchmarking pipelines	Medium	Academic & Industrial
Automated, adaptable solutions for heterogeneous settings	Medium	Academic & Industrial
Reusable ontology patterns	Medium	Academic
Expert system for knowledge acquisition	Long	Academic & Industrial
Community-accepted standard for IoT-enriched event logs	Long	Academic & Industrial
Dedicated AI-based solution for real-life deployment	Long	Academic & Industrial

- *Recommendation 3*: Work toward a community-accepted standard for IoT-enriched event logs that can represent process events together with linked sensor observations, context, provenance, uncertainty, and multi-entity relations, while remaining interoperable with established process mining tools and standards.

Table 1 summarizes these recommendations across time horizons and stakeholders.

6. Conclusions

This paper highlights the central challenge of transforming heterogeneous, low-level IoT sensor streams into structured process-level event logs that can be used for process mining. Although IoT environments generate large volumes of potentially valuable data, the lack of automated, scalable, and reusable abstraction mechanisms prevents organizations from fully exploiting this data for process monitoring and improvement. Addressing this challenge requires bridging the semantic gap between raw sensor observations and business process concepts through automated event recognition, case identification, semantic enrichment, and standardized event log representations.

The analysis of existing research reveals several major limitations. Current approaches often rely on manually defined rules, process-specific configurations, or domain expertise, thereby reducing portability and adaptability across different IoT environments. Furthermore, robust methods for directly identifying case identifiers from raw IoT data remain limited, evaluation methods are hindered by the lack of reliable ground-truth datasets, and existing abstraction pipelines struggle to represent the temporal dynamics and transitions inherent in continuous sensor data. Research efforts are also fragmented, typically addressing abstraction, correlation, semantics, and standardization separately rather than as integrated pipelines.

To address these issues, this paper emphasizes several research priorities. In the short term, efforts should focus on improving evaluation methodologies, documenting dataset characteristics, and investigating temporal dynamics in sensor data. Medium-term priorities include the development of reusable frameworks for abstracting events from the IoT systems into processes, more general case-identification techniques, standardized benchmarking pipelines, and reusable ontology patterns. In the long term, the community should work toward AI-assisted abstraction systems and community-accepted standards for IoT-enriched event logs that integrate sensor observations, context, and process-level semantics.

In conclusion, advancing IoT-driven process mining will require interdisciplinary solutions that combine machine learning, semantic technologies, process modeling, and standardized data representations to enable reliable, adaptive, and explainable abstraction of IoT data into meaningful process events.

Acknowledgments

The research was partly financed by a subsidy and the program „Excellence initiative – research university" granted to the AGH University of Krakow.

References

[1] M. Dumas, M. Rosa, J. Mendling, H. Reijers, Fundamentals of Business Process Management, Springer Berlin Heidelberg, 2019.

[2] J. D. Weerdt, M. T. Wynn, Foundations of process event data, in: W. M. P. van der Aalst, J. Carmona (Eds.), Process Mining Handbook, volume 448 of *Lecture Notes in Business Information Processing*, Springer, 2022, pp. 193–211. doi:10.1007/978-3-031-08848-3_6.

[3] C. Janiesch, A. Koschmider, M. Mecella, B. Weber, A. Burattin, C. Di Ciccio, G. Fortino, A. Gal, U. Kannengiesser, F. Leotta, F. Mannhardt, A. Marrella, J. Mendling, A. Oberweis, M. Reichert, S. Rinderle-Ma, E. Serral, W. Song, J. Su, V. Torres, M. Weidlich, M. Weske, L. Zhang, The internet of things meets business process management: A manifesto, IEEE Systems, Man, and Cybernetics Magazine 6 (2020) 34–44. doi:10.1109/MSMC.2020.3003135.

[4] S. J. van Zelst, F. Mannhardt, M. de Leoni, A. Koschmider, Event abstraction in process mining: literature review and taxonomy, Granular Computing 6 (2021) 719–736.

[5] E. Brzychczy, M. Aleknonytė-Resch, D. Janssen, A. Koschmider, Process mining on sensor data: a review of related works, Knowledge and Information Systems (2025) 1–34.

[6] E. Brzychczy, U. Jessen, K. Kluza, S. Sriram, M. V. Nettelnstroth, Iot miner: Intelligent extraction of event logs from sensor data for process mining, in: Business Process Management Workshops: BPM 2025 International Workshops, Seville, Spain, August 31–September 5, 2025, Revised Selected Papers, Springer, 2026, p. 232–246.

[7] M. Shirali, M. F. Sani, Z. Ahmadi, E. Serral, Llm-based event abstraction and integration for iot-sourced logs, in: K. Gdowska, M. T. Gómez-López, J. Rehse (Eds.), Business Process Management Workshops - BPM 2024 International Workshops, Krakow, Poland, September 1-6, 2024, Revised Selected Papers, volume 534 of *Lecture Notes in Business Information Processing*, Springer, 2024, pp. 138–149. doi:10.1007/978-3-031-78666-2_11.

[8] K. Diba, K. Batoulis, M. Weidlich, M. Weske, Extraction, correlation, and abstraction of event data for process mining, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 10 (2020) e1346.

[9] A. Koschmider, D. Janssen, F. Mannhardt, Framework for process discovery from sensor data, in: 10th International Workshop on Enterprise Modeling and Information Systems Architectures, EMISA 2020, CEUR-WS. org, 2020, pp. 32–38.

[10] A. Buchmann, B. Koldehofe, Complex event processing, it - Information Technology (2009).

[11] R. Seiger, F. Zerbato, A. Burattin, L. García-Bañuelos, B. Weber, Towards iot-driven process event log generation for conformance checking in smart factories, in: 2020 IEEE 24th international enterprise distributed object computing workshop (edocw), IEEE, 2020, pp. 20–26.

[12] R. Seiger, A. Kurz, M. Franceschetti, Online detection of process activity executions from iot sensors using generated event processing services, Future Gener. Comput. Syst. 174 (2026) 107987. doi:10.1016/J.FUTURE.2025.107987.

[13] D. Bayomie, C. Di Ciccio, J. Mendling, Event-case correlation for process mining using probabilistic optimization, Inf. Syst. 114 (2023). doi:10.1016/j.is.2023.102167.

[14] Y. Bertrand, B. Van den Abbeele, S. Veneruso, F. Leotta, M. Mecella, E. Serral, A survey on the application of process discovery techniques to smart spaces data, Engineering Applications of Artificial Intelligence 126 (2023) 106748. doi:10.1016/j.engappai.2023.106748.

[15] R. De Fazio, A. Balzanella, S. Marrone, F. Marulli, L. Verde, V. Reccia, P. Valletta, Caseid detection for process mining: A heuristic-based methodology, in: J. De Smedt, P. Soffer (Eds.), Process Mining Workshops, Springer Nature Switzerland, Cham, 2024, pp. 45–57.

[16] J. Brock, N. Rempe, S. von Enzberg, A. Kühn, R. Dumitrescu, A framework for the domain-driven utilization of manufacturing sensor data in process mining: An action design approach, Proceedings of the Conference on Production Systems and Logistics: CPSL 2023 - 2 (2023) 771–781. doi:10.15488/15267.
[17] A. Vicente, C. Grilo, R. Rijo, R. Martinho, A simple heuristic for the identification of the case id attribute in unlabelled process mining event logs, Array 30 (2026) 100762. doi:https://doi.org/10.1016/j.array.2026.100762.
[18] D. Janssen, F. Mannhardt, A. Koschmider, S. J. van Zelst, Process model discovery from sensor event data, in: S. Leemans, H. Leopold (Eds.), Process Mining Workshops, Springer International Publishing, Cham, 2021, pp. 69–81.
[19] A. Swevels, R. Dijkman, D. Fahland, Inferring missing entity identifiers from context using event knowledge graphs, in: C. Di Francescomarino, A. Burattin, C. Janiesch, S. Sadiq (Eds.), Business Process Management, Springer Nature Switzerland, Cham, 2023, pp. 180–197.
[20] E. Brzychczy, T. Pełech-Pilichowski, Z. Dworakowski, Case id detection based on time series data – the mining use case, 2024. arXiv:2410.23846.
[21] L. Esposito, S. Veneruso, F. Leotta, F. Monti, J. G. Mathew, M. Mecella, Unsupervised segmentation of human habits in smart home logs through process discovery, in: A. Marrella, D. Theseider Dupré (Eds.), Proceedings of the 1st Italian Forum on Business Process Management (ITBPM 2021), co-located with the 19th International Conference on Business Process Management (BPM 2021), volume 2952 of *CEUR Workshop Proceedings*, CEUR-WS.org, Rome, Italy, 2021, pp. 56–61. URL: https://ceur-ws.org/Vol-2952/paper_300a.pdf.
[22] P. Przymus, K. Rykaczewski, J. Zieliński, Ł. Mikulski, Segmentation and process assignment of semi-structured event logs, in: M. Bolanowski, M. Ganzha, L. A. Maciaszek, M. Paprzycki, D. Ślęzak (Eds.), Proceedings of the 20th Conference on Computer Science and Intelligence Systems (FedCSIS 2025), Kraków, Poland, September 14–17, 2025, volume 43 of *Annals of Computer Science and Information Systems*, 2025, pp. 231–241. doi:10.15439/2025F4755.
[23] E. Brzychczy, A. Trzcionkowska, Creation of an event log from a low-level machinery monitoring system for process mining purposes, in: H. Yin, D. Camacho, P. Novais, A. J. Tallón-Ballesteros (Eds.), Intelligent Data Engineering and Automated Learning - IDEAL 2018 - 19th International Conference, Madrid, Spain, November 21-23, 2018, Proceedings, Part II, volume 11315 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 54–63. doi:10.1007/978-3-030-03496-2_7.
[24] T. Elsaleh, S. Enshaeifar, R. Rezvani, S. T. Acton, V. Janeiko, M. Bermudez-Edo, IoT-Stream: A lightweight ontology for internet of things data streams and its use with data analytics and event detection services, Sensors 20 (2020) 953.
[25] K. Janowicz, A. Haller, S. J. Cox, D. Le Phuoc, M. Lefrançois, SOSA: A lightweight ontology for sensors, observations, samples, and actuators, Journal of Web Semantics 56 (2019) 1–10.
[26] S. Lemaignan, A. Siadat, J.-Y. Dantan, A. Semenenko, MASON: A Proposal For An Ontology Of Manufacturing Domain, in: IEEE Workshop on Distributed Intell. Syst.: Collective Intelligence and Its Applications, IEEE, 2006, pp. 195–200.
[27] F. Ameri, D. Dutta, An Upper Ontology for Manufacturing Service Description, in: Volume 3: 26th Computers and Information in Engineering Conf., ASME, 2006, pp. 651–661.
[28] P. Klein, L. Malburg, R. Bergmann, FTOnto: A Domain Ontology for a Fischertechnik Simulation Production Factory by Reusing Existing Ontologies, in: 21st LWDA, volume 2454, CEUR-WS.org, 2019, pp. 253–264.
[29] Y. Bertrand, J. De Weerdt, E. Serral, Assessing the Suitability of Traditional Event Log Standards for IoT-Enhanced Event Logs, in: Business Process Management Workshops, Springer, Cham, Switzerland, 2023, pp. 63–75. doi:10.1007/978-3-031-25383-6_6.
[30] J. Mangler, J. Grüger, L. Malburg, M. Ehrendorfer, Y. Bertrand, J.-V. Benzin, S. Rinderle-Ma, E. Serral Asensio, R. Bergmann, DataStream XES Extension: Embedding IoT Sensor Data into Extensible Event Stream Logs, Future Internet 15 (2023) 109. doi:10.3390/fi15030109.
[31] Y. Bertrand, S. Veneruso, F. Leotta, M. Mecella, E. Serral, NICE: The Native IoT-Centric Event Log Format for Process Mining, Springer Nature Switzerland, Cham, 2026, pp. 219–249. URL:

https://doi.org/10.1007/978-3-031-90746-3_10. doi:10.1007/978-3-031-90746-3_10.
[32] M. Franceschetti, R. Seiger, B. Weber, An event-centric metamodel for iot-driven process monitoring and conformance checking, in: International Conference on Business Process Management, Springer, 2024, pp. 131–143. doi:10.1007/978-3-031-50974-2_11.
[33] Y. Bertrand, C. Imenkamp, L. Malburg, M. Ehrendorfer, M. Franceschetti, J. Grüger, F. Leotta, J. Mangler, R. Seiger, A. Koschmider, et al., An object-centric core metamodel for iot-enhanced event logs, arXiv preprint arXiv:2506.21300 (2025).
[34] M. Ehrendorfer, Y. Bertrand, L. Malburg, J. Mangler, J. Grüger, S. Rinderle-Ma, R. Bergmann, E. Serral Asensio, Internet of processes and things: A repository for iot-enriched event logs in smart environments, in: Proceedings of the Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Forum at BPM 2023 (BPM-D 2023), volume 3469 of *CEUR Workshop Proceedings*, Utrecht, The Netherlands, 2023, pp. 92–96. URL: https://ceur-ws.org/Vol-3469/paper-16.pdf.
[35] G. Van Houdt, M. de Leoni, N. Martin, B. Depaire, An empirical evaluation of unsupervised event log abstraction techniques in process mining, Information Systems 121 (2024) 102320. doi:10.1016/j.is.2023.102320.
[36] S. Mandal, M. Hewelt, M. Weske, A framework for integrating real-world events and business processes in an iot environment, in: H. Panetto, C. Debruyne, W. Gaaloul, M. P. Papazoglou, A. Paschke, C. A. Ardagna, R. Meersman (Eds.), On the Move to Meaningful Internet Systems. OTM 2017 Conferences - Confederated International Conferences: CoopIS, C&TC, and ODBASE 2017, Rhodes, Greece, October 23-27, 2017, Proceedings, Part I, volume 10573 of *Lecture Notes in Computer Science*, Springer, 2017, pp. 194–212. doi:10.1007/978-3-319-69462-7_13.
[37] P. Valderas, V. Torres, Towards a semantic interoperability in iotenhanced business processes. an event-driven solution based on microservices, in: 19th International Conference on Intelligent Environments, IE 2023, Uniciti, Mauritius, June 29-30, 2023, IEEE, 2023, pp. 1–8. doi:10.1109/IE57519.2023.10179096.
[38] J. Mangler, R. Seiger, J. Benzin, J. Grüger, Y. Kirikkayis, F. Gallik, L. Malburg, M. Ehrendorfer, Y. Bertrand, M. Franceschetti, B. Weber, S. Rinderle-Ma, R. Bergmann, E. S. Asensio, M. Reichert, From internet of things data to business processes: Challenges and a framework, CoRR abs/2405.08528 (2024). URL: https://doi.org/10.48550/arXiv.2405.08528. doi:10.48550/ARXIV.2405.08528. arXiv:2405.08528.
[39] C. Diamantini, A. Mircoli, D. Potena, E. Storti, Process-aware iiot knowledge graph: A semantic model for industrial iot integration and analytics, Future Gener. Comput. Syst. 139 (2023) 224–238. doi:10.1016/J.FUTURE.2022.10.003.
[40] G. Meroni, P. Plebani, Artifact-driven monitoring for human-centric business processes with smart devices: Assessment and improvement, in: J. Carmona, G. Engels, A. Kumar (Eds.), Business Process Management Forum - BPM Forum 2017, Barcelona, Spain, September 10-15, 2017, Proceedings, volume 297 of *Lecture Notes in Business Information Processing*, Springer, 2017, pp. 160–176. doi:10.1007/978-3-319-65015-9_10.
[41] T. Panetti, A. D'Ambrogio, P. Bocciarelli, Process over things (pot): an ontology based approach for iot-aware business processes, in: 30th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2021, Bayonne, France, October 27-29, 2021, IEEE, 2021, pp. 95–100. doi:10.1109/WETICE53228.2021.00029.
[42] P. Bocciarelli, A. D'Ambrogio, T. Panetti, A model based framework for iot-aware business process management, Future Internet 15 (2023) 50. doi:10.3390/FI15020050.
[43] Y. Kirikkayis, F. Gallik, M. Winter, M. Reichert, Bpmne4iot: A framework for modeling, executing and monitoring iot-driven processes, Future Internet 15 (2023) 90. doi:10.3390/FI15030090.
[44] A. Fedeli, F. Fornari, A. Polini, B. Re, V. Torres, P. Valderas, Flobp: a model-driven approach for developing and executing iot-enhanced business processes, Softw. Syst. Model. 23 (2024) 1217–1246. doi:10.1007/S10270-024-01150-8.
[45] V. Gruhn, Y. Han, M. Hesenius, M. Reichert, G. Wang, J. Yu, L. Zhang, BRIBOT: towards a service-based methodology for bridging business processes and iot big data, in: H. Hacid, O. Kao, M. Mecella, N. Moha, H. Paik (Eds.), Service-Oriented Computing - 19th International Conference,

ICSOC 2021, Virtual Event, November 22-25, 2021, Proceedings, volume 13121 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 597–611. doi:10.1007/978-3-030-91431-8_37.

[46] R. Seiger, D. Locher, M. Kaufmann, A. F. Kurz, A domain-specific language and architecture for detecting process activities from sensor streams in iot, Internet Things 36 (2026) 101870. doi:10.1016/J.IOT.2025.101870.

[47] İ. Kök, O. Demirci, S. Özdemir, When iot meet llms: Applications and challenges, in: 2024 IEEE International Conference on Big Data (BigData), IEEE, 2024, pp. 7075–7084.

[48] F. Sarhaddi, N. T. Nguyen, A. Zuniga, P. Hui, S. Tarkoma, H. Flores, P. Nurmi, Llms and iot: A comprehensive survey on large language models and the internet of things, Authorea Preprints (2025).

[49] E. Brzychczy, K. Kluza, L. Szała, Enhancement of low-level event abstraction with large language models (llms), in: K. Gdowska, M. T. Gómez-López, J.-R. Rehse (Eds.), Business Process Management Workshops, Springer Nature Switzerland, Cham, 2025, pp. 209–220.

[50] T. An, Y. Zhou, H. Zou, J. Yang, Iot-llm: Enhancing real-world iot task reasoning with large language models, arXiv preprint arXiv:2410.02429 (2024).

[51] B. Xiao, B. Kantarci, J. Kang, D. Niyato, M. Guizani, Efficient prompting for LLM-based generative Internet of Things, IEEE Internet of Things Journal (2024).

[52] V. Stein Dani, M. Dees, H. Leopold, K. Busch, I. Beerepoot, J. M. E. van der Werf, H. A. Reijers, Event log extraction for process mining using large language models, in: International Conference on Cooperative Information Systems, Springer, 2024, pp. 56–72.

[53] U. Jessen, L. Schroth, M. Mühllechner, From data to actionable insights: Utilizing ai and process mining in manufacturing processes, in: International Conference on Business Process Management, Springer, 2024, pp. 462–471.